Package 'civis'

May 8, 2026

Title R Client for the 'Civis Platform API'

Version 3.1.4

Description A convenient interface for making
requests directly to the 'Civis Platform API' <https://www.civisanalytics.com/platform>.
Full documentation available 'here' <https://civisanalytics.github.io/civis-r/>.

Depends R (>= 3.2.0)

License BSD_3_clause + file LICENSE

URL https://github.com/civisanalytics/civis-r

BugReports https://github.com/civisanalytics/civis-r/issues

Imports future (>= 1.8.0), httr, jsonlite, methods, memoise, stats,
utils

Suggests feather, ggplot2, knitr, lifecycle, rmarkdown, roxygen2,
mockery, R.utils, rstudioapi, testthat, yaml

RoxygenNote 7.3.3

VignetteBuilder knitr

Collate 'await.R' 'civis_future.R' 'civis_ml.R' 'civis_ml_plot.R'
'civis_ml_utils.R' 'civis_ml_workflows.R' 'client_base.R'
'generate_client.R' 'generated_client.R' 'io.R' 'logs.R'
'pagination_helpers.R' 'reports.R' 'rstudio.R' 'scripts.R'
'tables.R' 'utils.R' 'zzz.R'

Encoding UTF-8

NeedsCompilation no

Author Laura Cleary [cre],
Dustin Leatherman [ctb],
Peter Cooman [ctb],
Patrick Miller [aut],
Keith Ingersoll [aut],
Bill Lattner [ctb],
Anh Le [ctb],
Michelangelo D'Agostino [ctb],
Sam Weiss [ctb],

Stephen Hoover [ctb],
Danning Chen [ctb],
Elizabeth Sander [ctb],
Madison Hobbs [ctb],
Anna Bladey [ctb],
Sahil Shah [ctb],
Bryan Baird [ctb]

Maintainer Laura Cleary <lcleary@civisanalytics.com>

Repository CRAN

Date/Publication 2026-03-27 21:20:02 UTC

Contents

admin_list_organizations . . . 22
aliases_delete . . . 23
aliases_delete_shares_groups . . . 24
aliases_delete_shares_users . . . 24
aliases_get . . . 25
aliases_get_object_type . . . 25
aliases_list . . . 26
aliases_list_dependencies . . . 27
aliases_list_shares . . . 28
aliases_patch . . . 29
aliases_post . . . 30
aliases_put . . . 31
aliases_put_shares_groups . . . 32
aliases_put_shares_users . . . 33
aliases_put_transfer . . . 34
announcements_list . . . 35
await . . . 36
cancel . . . 38
CivisFuture . . . 38
civis_file . . . 40
civis_file_manifest . . . 41
civis_ml . . . 41
CIVIS_ML_CLASSIFIERS . . . 47
civis_ml_extra_trees_classifier . . . 48
civis_ml_extra_trees_regressor . . . 52
civis_ml_gradient_boosting_classifier . . . 56
civis_ml_gradient_boosting_regressor . . . 61
civis_ml_random_forest_classifier . . . 65
civis_ml_random_forest_regressor . . . 70
CIVIS_ML_REGRESSORS . . . 74
civis_ml_sparse_linear_regressor . . . 74
civis_ml_sparse_logistic . . . 77
civis_ml_sparse_ridge_regressor . . . 82
civis_platform . . . 85

civis_script . . . 86
civis_table . . . 87
civis_to_multifile_csv . . . 88
clusters_delete_kubernetes_partitions . . . 89
clusters_get_kubernetes . . . 90
clusters_get_kubernetes_instance_configs . . . 91
clusters_get_kubernetes_partitions . . . 92
clusters_list_kubernetes . . . 93
clusters_list_kubernetes_compute_hours . . . 95
clusters_list_kubernetes_compute_metrics . . . 95
clusters_list_kubernetes_deployments . . . 96
clusters_list_kubernetes_deployment_stats . . . 98
clusters_list_kubernetes_instance_configs_active_workloads . . . 98
clusters_list_kubernetes_instance_configs_historical_graphs . . . 99
clusters_list_kubernetes_instance_configs_historical_metrics . . . 100
clusters_list_kubernetes_instance_configs_user_statistics . . . 101
clusters_list_kubernetes_partitions . . . 102
clusters_patch_kubernetes_partitions . . . 103
clusters_post_kubernetes_partitions . . . 105
coef.civis_ml . . . 106
credentials_delete . . . 107
credentials_delete_shares_groups . . . 107
credentials_delete_shares_users . . . 108
credentials_get . . . 108
credentials_list . . . 109
credentials_list_dependencies . . . 111
credentials_list_shares . . . 112
credentials_list_types . . . 113
credentials_patch . . . 113
credentials_post . . . 115
credentials_post_authenticate . . . 117
credentials_post_temporary . . . 118
credentials_put . . . 119
credentials_put_shares_groups . . . 120
credentials_put_shares_users . . . 121
credentials_put_transfer . . . 122
databases_delete_schemas_tables_projects . . . 123
databases_get . . . 124
databases_get_schemas_tables . . . 125
databases_get_schema_privileges . . . 128
databases_get_status_graphs_timeframe . . . 128
databases_get_table_privilegesschema_name . . . 129
databases_get_whitelist_ips . . . 130
databases_list . . . 130
databases_list_advanced_settings . . . 131
databases_list_groups . . . 132
databases_list_schemas . . . 132
databases_list_schemas_tables . . . 133

databases_list_schemas_tables_projects . . . 133
databases_list_tables . . . 134
databases_list_tables_search . . . 136
databases_list_users . . . 137
databases_list_whitelist_ips . . . 138
databases_patch_advanced_settings . . . 138
databases_patch_schemas_tables . . . 139
databases_post_schemas_scan . . . 142
databases_put_advanced_settings . . . 143
databases_put_schemas_tables_projects . . . 143
default_credential . . . 144
download_civis . . . 144
endpoints_list . . . 146
enhancements_delete_cass_ncoa_projects . . . 147
enhancements_delete_cass_ncoa_runs . . . 147
enhancements_delete_cass_ncoa_shares_groups . . . 148
enhancements_delete_cass_ncoa_shares_users . . . 148
enhancements_delete_civis_data_match_projects . . . 149
enhancements_delete_civis_data_match_runs . . . 149
enhancements_delete_civis_data_match_shares_groups . . . 150
enhancements_delete_civis_data_match_shares_users . . . 150
enhancements_delete_geocode_projects . . . 151
enhancements_delete_geocode_runs . . . 151
enhancements_delete_geocode_shares_groups . . . 152
enhancements_delete_geocode_shares_users . . . 152
enhancements_delete_identity_resolution_projects . . . 153
enhancements_delete_identity_resolution_runs . . . 153
enhancements_delete_identity_resolution_shares_groups . . . 154
enhancements_delete_identity_resolution_shares_users . . . 154
enhancements_get_cass_ncoa . . . 155
enhancements_get_cass_ncoa_runs . . . 157
enhancements_get_civis_data_match . . . 158
enhancements_get_civis_data_match_runs . . . 160
enhancements_get_geocode . . . 161
enhancements_get_geocode_runs . . . 163
enhancements_get_identity_resolution . . . 164
enhancements_get_identity_resolution_runs . . . 167
enhancements_list . . . 168
enhancements_list_cass_ncoa_dependencies . . . 169
enhancements_list_cass_ncoa_projects . . . 170
enhancements_list_cass_ncoa_runs . . . 171
enhancements_list_cass_ncoa_runs_logs . . . 172
enhancements_list_cass_ncoa_runs_outputs . . . 173
enhancements_list_cass_ncoa_shares . . . 174
enhancements_list_civis_data_match_dependencies . . . 175
enhancements_list_civis_data_match_projects . . . 175
enhancements_list_civis_data_match_runs . . . 176
enhancements_list_civis_data_match_runs_logs . . . 177

enhancements_list_civis_data_match_runs_outputs . . . 178
enhancements_list_civis_data_match_shares . . . 179
enhancements_list_field_mapping . . . 180
enhancements_list_geocode_dependencies . . . 180
enhancements_list_geocode_projects . . . 181
enhancements_list_geocode_runs . . . 182
enhancements_list_geocode_runs_logs . . . 183
enhancements_list_geocode_runs_outputs . . . 184
enhancements_list_geocode_shares . . . 185
enhancements_list_identity_resolution . . . 186
enhancements_list_identity_resolution_dependencies . . . 187
enhancements_list_identity_resolution_projects . . . 188
enhancements_list_identity_resolution_runs . . . 189
enhancements_list_identity_resolution_runs_logs . . . 191
enhancements_list_identity_resolution_shares . . . 192
enhancements_list_types . . . 193
enhancements_patch_cass_ncoa . . . 193
enhancements_patch_civis_data_match . . . 197
enhancements_patch_geocode . . . 201
enhancements_patch_identity_resolution . . . 205
enhancements_post_cass_ncoa . . . 210
enhancements_post_cass_ncoa_cancel . . . 214
enhancements_post_cass_ncoa_runs . . . 215
enhancements_post_civis_data_match . . . 215
enhancements_post_civis_data_match_cancel . . . 219
enhancements_post_civis_data_match_clone . . . 220
enhancements_post_civis_data_match_runs . . . 222
enhancements_post_geocode . . . 223
enhancements_post_geocode_cancel . . . 226
enhancements_post_geocode_runs . . . 227
enhancements_post_identity_resolution . . . 227
enhancements_post_identity_resolution_cancel . . . 232
enhancements_post_identity_resolution_clone . . . 233
enhancements_post_identity_resolution_runs . . . 236
enhancements_put_cass_ncoa . . . 238
enhancements_put_cass_ncoa_archive . . . 242
enhancements_put_cass_ncoa_projects . . . 245
enhancements_put_cass_ncoa_shares_groups . . . 245
enhancements_put_cass_ncoa_shares_users . . . 246
enhancements_put_cass_ncoa_transfer . . . 247
enhancements_put_civis_data_match . . . 248
enhancements_put_civis_data_match_archive . . . 252
enhancements_put_civis_data_match_projects . . . 254
enhancements_put_civis_data_match_shares_groups . . . 255
enhancements_put_civis_data_match_shares_users . . . 256
enhancements_put_civis_data_match_transfer . . . 257
enhancements_put_geocode . . . 258
enhancements_put_geocode_archive . . . 261

enhancements_put_geocode_projects . . . 263
enhancements_put_geocode_shares_groups . . . 264
enhancements_put_geocode_shares_users . . . 265
enhancements_put_geocode_transfer . . . 266
enhancements_put_identity_resolution . . . 267
enhancements_put_identity_resolution_archive . . . 272
enhancements_put_identity_resolution_projects . . . 275
enhancements_put_identity_resolution_shares_groups . . . 276
enhancements_put_identity_resolution_shares_users . . . 277
enhancements_put_identity_resolution_transfer . . . 278
exports_delete_files_csv_runs . . . 279
exports_get_files_csv . . . 279
exports_get_files_csv_runs . . . 281
exports_list . . . 281
exports_list_files_csv_runs . . . 283
exports_list_files_csv_runs_logs . . . 284
exports_list_files_csv_runs_outputs . . . 285
exports_patch_files_csv . . . 286
exports_post_files_csv . . . 288
exports_post_files_csv_runs . . . 290
exports_put_files_csv . . . 291
exports_put_files_csv_archive . . . 293
fetch_all . . . 295
fetch_and_generate_client . . . 295
fetch_logs . . . 296
fetch_oos_scores . . . 297
fetch_output_file_ids . . . 297
fetch_predictions . . . 298
fetch_until . . . 299
files_delete_preprocess_csv . . . 300
files_delete_projects . . . 300
files_delete_shares_groups . . . 301
files_delete_shares_users . . . 301
files_get . . . 302
files_get_preprocess_csv . . . 303
files_list_dependencies . . . 304
files_list_projects . . . 304
files_list_shares . . . 305
files_patch . . . 306
files_patch_preprocess_csv . . . 307
files_post . . . 309
files_post_multipart . . . 310
files_post_multipart_complete . . . 311
files_post_preprocess_csv . . . 311
files_put . . . 313
files_put_preprocess_csv . . . 314
files_put_preprocess_csv_archive . . . 315
files_put_projects . . . 316

files_put_shares_groups . . . 317
files_put_shares_users . . . 318
files_put_transfer . . . 319
generate_client . . . 320
get_database_id . . . 320
get_error . . . 321
get_feature_importance . . . 322
get_job_type_version . . . 322
get_metric . . . 323
get_script_fun . . . 323
get_status . . . 324
get_table_id . . . 325
get_template_ids_all_versions . . . 325
get_train_template_id . . . 326
git_repos_delete . . . 326
git_repos_get . . . 327
git_repos_list . . . 327
git_repos_list_refs . . . 328
git_repos_post . . . 328
groups_delete . . . 329
groups_delete_members . . . 329
groups_delete_shares_groups . . . 330
groups_delete_shares_users . . . 330
groups_get . . . 331
groups_list . . . 332
groups_list_child_groups . . . 333
groups_list_shares . . . 334
groups_patch . . . 335
groups_post . . . 337
groups_put . . . 339
groups_put_members . . . 341
groups_put_shares_groups . . . 342
groups_put_shares_users . . . 343
hist.civis_ml . . . 344
imports_delete_files_csv . . . 345
imports_delete_files_csv_runs . . . 345
imports_delete_files_runs . . . 346
imports_delete_projects . . . 346
imports_delete_shares_groups . . . 347
imports_delete_shares_users . . . 347
imports_delete_syncs . . . 348
imports_get . . . 348
imports_get_batches . . . 351
imports_get_files_csv . . . 351
imports_get_files_csv_runs . . . 353
imports_get_files_runs . . . 354
imports_list . . . 355
imports_list_batches . . . 357

imports_list_dependencies . . . 358
imports_list_files_csv_runs . . . 359
imports_list_files_csv_runs_logs . . . 360
imports_list_files_runs . . . 361
imports_list_files_runs_logs . . . 362
imports_list_projects . . . 362
imports_list_runs . . . 363
imports_list_runs_logs . . . 364
imports_list_shares . . . 365
imports_patch_files_csv . . . 366
imports_post . . . 369
imports_post_batches . . . 373
imports_post_cancel . . . 374
imports_post_files . . . 375
imports_post_files_csv . . . 376
imports_post_files_csv_runs . . . 380
imports_post_files_runs . . . 381
imports_post_runs . . . 381
imports_post_syncs . . . 382
imports_put . . . 387
imports_put_archive . . . 391
imports_put_files_csv . . . 393
imports_put_files_csv_archive . . . 397
imports_put_projects . . . 399
imports_put_shares_groups . . . 399
imports_put_shares_users . . . 401
imports_put_syncs . . . 402
imports_put_syncs_archive . . . 407
imports_put_transfer . . . 410
jobs_delete_projects . . . 411
jobs_delete_runs . . . 411
jobs_delete_shares_groups . . . 412
jobs_delete_shares_users . . . 412
jobs_get . . . 413
jobs_get_runs . . . 414
jobs_list . . . 415
jobs_list_children . . . 417
jobs_list_dependencies . . . 418
jobs_list_parents . . . 419
jobs_list_projects . . . 420
jobs_list_runs . . . 421
jobs_list_runs_logs . . . 422
jobs_list_runs_outputs . . . 423
jobs_list_shares . . . 424
jobs_list_workflows . . . 425
jobs_post_runs . . . 426
jobs_post_trigger_email . . . 427
jobs_put_archive . . . 427

jobs_put_projects . . . 429
jobs_put_shares_groups . . . 429
jobs_put_shares_users . . . 430
jobs_put_transfer . . . 431
json_values_delete_shares_groups . . . 432
json_values_delete_shares_users . . . 433
json_values_get . . . 433
json_values_list_dependencies . . . 434
json_values_list_shares . . . 434
json_values_patch . . . 435
json_values_post . . . 436
json_values_put_shares_groups . . . 436
json_values_put_shares_users . . . 438
json_values_put_transfer . . . 439
match_targets_delete_shares_groups . . . 440
match_targets_delete_shares_users . . . 440
match_targets_get . . . 441
match_targets_list . . . 441
match_targets_list_shares . . . 442
match_targets_patch . . . 443
match_targets_post . . . 443
match_targets_put_archive . . . 444
match_targets_put_shares_groups . . . 445
match_targets_put_shares_users . . . 446
media_delete_optimizations_runs . . . 447
media_delete_optimizations_shares_groups . . . 447
media_delete_optimizations_shares_users . . . 448
media_delete_ratecards_shares_groups . . . 448
media_delete_ratecards_shares_users . . . 449
media_delete_spot_orders_shares_groups . . . 449
media_delete_spot_orders_shares_users . . . 450
media_get_optimizations . . . 450
media_get_optimizations_runs . . . 452
media_get_ratecards . . . 452
media_get_spot_orders . . . 453
media_list_dmas . . . 454
media_list_optimizations . . . 454
media_list_optimizations_runs . . . 455
media_list_optimizations_runs_logs . . . 456
media_list_optimizations_shares . . . 457
media_list_ratecards . . . 458
media_list_ratecards_shares . . . 459
media_list_spot_orders . . . 460
media_list_spot_orders_shares . . . 460
media_list_targets . . . 461
media_patch_optimizations . . . 462
media_patch_ratecards . . . 464
media_post_optimizations . . . 465

media_post_optimizations_clone . . . 467
media_post_optimizations_runs . . . 469
media_post_ratecards . . . 470
media_post_spot_orders . . . 470
media_put_optimizations_archive . . . 471
media_put_optimizations_shares_groups . . . 473
media_put_optimizations_shares_users . . . 474
media_put_ratecards . . . 475
media_put_ratecards_archive . . . 476
media_put_ratecards_shares_groups . . . 476
media_put_ratecards_shares_users . . . 477
media_put_spot_orders . . . 478
media_put_spot_orders_archive . . . 479
media_put_spot_orders_shares_groups . . . 480
media_put_spot_orders_shares_users . . . 481
models_delete_builds . . . 482
models_delete_projects . . . 482
models_delete_shares_groups . . . 483
models_delete_shares_users . . . 483
models_get . . . 484
models_get_builds . . . 487
models_list . . . 488
models_list_builds . . . 491
models_list_builds_logs . . . 492
models_list_dependencies . . . 493
models_list_projects . . . 494
models_list_schedules . . . 495
models_list_shares . . . 495
models_list_types . . . 496
models_put_archive . . . 497
models_put_projects . . . 500
models_put_shares_groups . . . 500
models_put_shares_users . . . 501
models_put_transfer . . . 502
multipart_upload . . . 503
notebooks_delete . . . 504
notebooks_delete_deployments . . . 504
notebooks_delete_projects . . . 505
notebooks_delete_shares_groups . . . 505
notebooks_delete_shares_users . . . 506
notebooks_get . . . 506
notebooks_get_deployments . . . 508
notebooks_get_git_commits . . . 509
notebooks_list . . . 510
notebooks_list_dependencies . . . 511
notebooks_list_deployments . . . 512
notebooks_list_deployments_logs . . . 514
notebooks_list_git . . . 515

notebooks_list_git_commits . . . 515
notebooks_list_projects . . . 516
notebooks_list_shares . . . 517
notebooks_list_update_links . . . 518
notebooks_patch . . . 518
notebooks_patch_git . . . 521
notebooks_post . . . 522
notebooks_post_clone . . . 525
notebooks_post_deployments . . . 527
notebooks_post_git_commits . . . 528
notebooks_put . . . 529
notebooks_put_archive . . . 532
notebooks_put_git . . . 534
notebooks_put_projects . . . 535
notebooks_put_shares_groups . . . 536
notebooks_put_shares_users . . . 537
notebooks_put_transfer . . . 538
notifications_list . . . 539
ontology_list . . . 539
organizations_delete_favorites . . . 540
organizations_list_favorites . . . 540
organizations_post_favorites . . . 542
permission_sets_delete_resources . . . 543
permission_sets_delete_resources_shares_groups . . . 543
permission_sets_delete_resources_shares_users . . . 544
permission_sets_delete_shares_groups . . . 544
permission_sets_delete_shares_users . . . 545
permission_sets_get . . . 545
permission_sets_get_resources . . . 546
permission_sets_list . . . 547
permission_sets_list_dependencies . . . 548
permission_sets_list_resources . . . 549
permission_sets_list_resources_shares . . . 550
permission_sets_list_shares . . . 551
permission_sets_list_users_permissions . . . 552
permission_sets_patch . . . 552
permission_sets_patch_resources . . . 553
permission_sets_post . . . 554
permission_sets_post_resources . . . 555
permission_sets_put . . . 555
permission_sets_put_archive . . . 556
permission_sets_put_resources_shares_groups . . . 557
permission_sets_put_resources_shares_users . . . 558
permission_sets_put_shares_groups . . . 559
permission_sets_put_shares_users . . . 560
permission_sets_put_transfer . . . 561
plot.civis_ml_classifier . . . 562
plot.civis_ml_regressor . . . 563

predictions_get . . . 563
predictions_list . . . 565
predictions_list_schedules . . . 566
print.civis_api . . . 566
projects_delete . . . 567
projects_delete_parent_projects . . . 568
projects_delete_shares_groups . . . 568
projects_delete_shares_users . . . 569
projects_get . . . 569
projects_list . . . 573
projects_list_dependencies . . . 575
projects_list_parent_projects . . . 575
projects_list_shares . . . 576
projects_post . . . 577
projects_post_clone . . . 581
projects_put . . . 585
projects_put_archive . . . 589
projects_put_auto_share . . . 593
projects_put_parent_projects . . . 597
projects_put_shares_groups . . . 598
projects_put_shares_users . . . 599
projects_put_transfer . . . 600
publish_addin . . . 601
publish_html . . . 601
publish_rmd . . . 602
queries_delete . . . 604
queries_delete_runs . . . 605
queries_get . . . 605
queries_get_runs . . . 606
queries_list . . . 607
queries_list_runs . . . 609
queries_list_runs_logs . . . 610
queries_post . . . 611
queries_post_runs . . . 613
queries_put_scripts . . . 613
query_civis . . . 614
query_civis_file . . . 616
read_civis . . . 617
refresh_table . . . 619
remote_hosts_delete_shares_groups . . . 620
remote_hosts_delete_shares_users . . . 621
remote_hosts_get . . . 621
remote_hosts_list . . . 622
remote_hosts_list_data_sets . . . 623
remote_hosts_list_shares . . . 624
remote_hosts_patch . . . 625
remote_hosts_post . . . 626
remote_hosts_post_authenticate . . . 627

remote_hosts_put . . . 628
remote_hosts_put_shares_groups . . . 629
remote_hosts_put_shares_users . . . 630
reports_delete_grants . . . 631
reports_delete_projects . . . 631
reports_delete_services_projects . . . 632
reports_delete_services_shares_groups . . . 632
reports_delete_services_shares_users . . . 633
reports_delete_shares_groups . . . 633
reports_delete_shares_users . . . 634
reports_delete_sql_projects . . . 634
reports_delete_sql_shares_groups . . . 635
reports_delete_sql_shares_users . . . 635
reports_get . . . 636
reports_get_git_commits . . . 637
reports_get_services . . . 638
reports_get_sql . . . 639
reports_list . . . 640
reports_list_dependencies . . . 642
reports_list_git . . . 643
reports_list_git_commits . . . 644
reports_list_projects . . . 644
reports_list_services_dependencies . . . 645
reports_list_services_projects . . . 646
reports_list_services_shares . . . 647
reports_list_shares . . . 648
reports_list_sql_dependencies . . . 649
reports_list_sql_projects . . . 649
reports_list_sql_shares . . . 650
reports_patch . . . 651
reports_patch_git . . . 653
reports_patch_services . . . 655
reports_patch_sql . . . 656
reports_post . . . 657
reports_post_git_commits . . . 659
reports_post_grants . . . 660
reports_post_refresh . . . 662
reports_post_services . . . 662
reports_post_sql . . . 663
reports_post_sql_refresh . . . 665
reports_put_archive . . . 666
reports_put_git . . . 668
reports_put_projects . . . 669
reports_put_services_archive . . . 669
reports_put_services_projects . . . 670
reports_put_services_shares_groups . . . 671
reports_put_services_shares_users . . . 672
reports_put_services_transfer . . . 673

reports_put_shares_groups . . . 674
reports_put_shares_users . . . 675
reports_put_sql_archive . . . 676
reports_put_sql_projects . . . 677
reports_put_sql_shares_groups . . . 678
reports_put_sql_shares_users . . . 679
reports_put_sql_transfer . . . 680
reports_put_transfer . . . 681
resolved . . . 682
roles_list . . . 682
run_civis . . . 683
run_template . . . 684
scripts_delete . . . 685
scripts_delete_containers . . . 685
scripts_delete_containers_projects . . . 686
scripts_delete_containers_runs . . . 686
scripts_delete_containers_shares_groups . . . 687
scripts_delete_containers_shares_users . . . 687
scripts_delete_custom . . . 688
scripts_delete_custom_projects . . . 688
scripts_delete_custom_runs . . . 689
scripts_delete_custom_shares_groups . . . 689
scripts_delete_custom_shares_users . . . 690
scripts_delete_dbt . . . 690
scripts_delete_dbt_projects . . . 691
scripts_delete_dbt_runs . . . 691
scripts_delete_dbt_shares_groups . . . 692
scripts_delete_dbt_shares_users . . . 692
scripts_delete_javascript . . . 693
scripts_delete_javascript_projects . . . 693
scripts_delete_javascript_runs . . . 694
scripts_delete_javascript_shares_groups . . . 694
scripts_delete_javascript_shares_users . . . 695
scripts_delete_python3 . . . 695
scripts_delete_python3_projects . . . 696
scripts_delete_python3_runs . . . 696
scripts_delete_python3_shares_groups . . . 697
scripts_delete_python3_shares_users . . . 697
scripts_delete_r . . . 698
scripts_delete_r_projects . . . 698
scripts_delete_r_runs . . . 699
scripts_delete_r_shares_groups . . . 699
scripts_delete_r_shares_users . . . 700
scripts_delete_sql . . . 700
scripts_delete_sql_projects . . . 701
scripts_delete_sql_runs . . . 701
scripts_delete_sql_shares_groups . . . 702
scripts_delete_sql_shares_users . . . 702

scripts_get . . . 703
scripts_get_containers . . . 706
scripts_get_containers_runs . . . 709
scripts_get_custom . . . 710
scripts_get_custom_runs . . . 714
scripts_get_dbt . . . 715
scripts_get_dbt_runs . . . 719
scripts_get_javascript . . . 720
scripts_get_javascript_git_commits . . . 723
scripts_get_javascript_runs . . . 723
scripts_get_python3 . . . 724
scripts_get_python3_git_commits . . . 727
scripts_get_python3_runs . . . 728
scripts_get_r . . . 729
scripts_get_r_git_commits . . . 732
scripts_get_r_runs . . . 733
scripts_get_sql . . . 734
scripts_get_sql_git_commits . . . 737
scripts_get_sql_runs . . . 738
scripts_list . . . 739
scripts_list_containers_dependencies . . . 741
scripts_list_containers_projects . . . 741
scripts_list_containers_runs . . . 742
scripts_list_containers_runs_logs . . . 743
scripts_list_containers_runs_outputs . . . 744
scripts_list_containers_shares . . . 745
scripts_list_custom . . . 746
scripts_list_custom_dependencies . . . 748
scripts_list_custom_projects . . . 748
scripts_list_custom_runs . . . 749
scripts_list_custom_runs_logs . . . 750
scripts_list_custom_runs_outputs . . . 751
scripts_list_custom_shares . . . 752
scripts_list_dbt_dependencies . . . 753
scripts_list_dbt_projects . . . 754
scripts_list_dbt_runs . . . 755
scripts_list_dbt_runs_logs . . . 756
scripts_list_dbt_runs_outputs . . . 757
scripts_list_dbt_shares . . . 758
scripts_list_history . . . 759
scripts_list_javascript_dependencies . . . 759
scripts_list_javascript_git . . . 760
scripts_list_javascript_git_commits . . . 761
scripts_list_javascript_projects . . . 762
scripts_list_javascript_runs . . . 763
scripts_list_javascript_runs_logs . . . 764
scripts_list_javascript_runs_outputs . . . 764
scripts_list_javascript_shares . . . 765

scripts_list_python3_dependencies . . . 766
scripts_list_python3_git . . . 767
scripts_list_python3_git_commits . . . 768
scripts_list_python3_projects . . . 768
scripts_list_python3_runs . . . 769
scripts_list_python3_runs_logs . . . 770
scripts_list_python3_runs_outputs . . . 771
scripts_list_python3_shares . . . 772
scripts_list_r_dependencies . . . 773
scripts_list_r_git . . . 774
scripts_list_r_git_commits . . . 774
scripts_list_r_projects . . . 775
scripts_list_r_runs . . . 776
scripts_list_r_runs_logs . . . 777
scripts_list_r_runs_outputs . . . 778
scripts_list_r_shares . . . 779
scripts_list_sql_dependencies . . . 780
scripts_list_sql_git . . . 780
scripts_list_sql_git_commits . . . 781
scripts_list_sql_projects . . . 782
scripts_list_sql_runs . . . 783
scripts_list_sql_runs_logs . . . 784
scripts_list_sql_runs_outputs . . . 785
scripts_list_sql_shares . . . 786
scripts_list_types . . . 787
scripts_patch_containers . . . 787
scripts_patch_container_runs . . . 793
scripts_patch_custom . . . 794
scripts_patch_dbt . . . 799
scripts_patch_dbt_runs . . . 805
scripts_patch_javascript . . . 806
scripts_patch_javascript_git . . . 811
scripts_patch_javascript_runs . . . 812
scripts_patch_python3 . . . 812
scripts_patch_python3_git . . . 818
scripts_patch_python3_runs . . . 819
scripts_patch_r . . . 819
scripts_patch_r_git . . . 825
scripts_patch_r_runs . . . 826
scripts_patch_sql . . . 826
scripts_patch_sql_git . . . 832
scripts_patch_sql_runs . . . 833
scripts_post . . . 833
scripts_post_cancel . . . 837
scripts_post_containers . . . 838
scripts_post_containers_clone . . . 844
scripts_post_containers_runs . . . 848
scripts_post_containers_runs_logs . . . 849

scripts_post_containers_runs_outputs . . . 850
scripts_post_custom . . . 850
scripts_post_custom_clone . . . 855
scripts_post_custom_runs . . . 859
scripts_post_custom_runs_outputs . . . 860
scripts_post_dbt . . . 861
scripts_post_dbt_clone . . . 868
scripts_post_dbt_runs . . . 872
scripts_post_dbt_runs_outputs . . . 873
scripts_post_javascript . . . 873
scripts_post_javascript_clone . . . 878
scripts_post_javascript_git_checkout . . . 881
scripts_post_javascript_git_checkout_latest . . . 882
scripts_post_javascript_git_commits . . . 883
scripts_post_javascript_runs . . . 883
scripts_post_javascript_runs_outputs . . . 884
scripts_post_python3 . . . 885
scripts_post_python3_clone . . . 890
scripts_post_python3_git_checkout . . . 894
scripts_post_python3_git_checkout_latest . . . 894
scripts_post_python3_git_commits . . . 895
scripts_post_python3_runs . . . 896
scripts_post_python3_runs_outputs . . . 897
scripts_post_r . . . 897
scripts_post_run . . . 903
scripts_post_r_clone . . . 903
scripts_post_r_git_checkout . . . 907
scripts_post_r_git_checkout_latest . . . 908
scripts_post_r_git_commits . . . 908
scripts_post_r_runs . . . 909
scripts_post_r_runs_outputs . . . 910
scripts_post_sql . . . 910
scripts_post_sql_clone . . . 916
scripts_post_sql_git_checkout . . . 919
scripts_post_sql_git_checkout_latest . . . 920
scripts_post_sql_git_commits . . . 921
scripts_post_sql_runs . . . 921
scripts_put_containers . . . 922
scripts_put_containers_archive . . . 928
scripts_put_containers_projects . . . 932
scripts_put_containers_shares_groups . . . 933
scripts_put_containers_shares_users . . . 934
scripts_put_containers_transfer . . . 935
scripts_put_custom . . . 936
scripts_put_custom_archive . . . 941
scripts_put_custom_projects . . . 944
scripts_put_custom_shares_groups . . . 945
scripts_put_custom_shares_users . . . 946

scripts_put_custom_transfer . . . 947
scripts_put_dbt . . . 948
scripts_put_dbt_archive . . . 955
scripts_put_dbt_projects . . . 959
scripts_put_dbt_shares_groups . . . 959
scripts_put_dbt_shares_users . . . 960
scripts_put_dbt_transfer . . . 961
scripts_put_javascript . . . 962
scripts_put_javascript_archive . . . 967
scripts_put_javascript_git . . . 970
scripts_put_javascript_projects . . . 971
scripts_put_javascript_shares_groups . . . 972
scripts_put_javascript_shares_users . . . 973
scripts_put_javascript_transfer . . . 974
scripts_put_python3 . . . 975
scripts_put_python3_archive . . . 980
scripts_put_python3_git . . . 984
scripts_put_python3_projects . . . 985
scripts_put_python3_shares_groups . . . 985
scripts_put_python3_shares_users . . . 986
scripts_put_python3_transfer . . . 987
scripts_put_r . . . 988
scripts_put_r_archive . . . 994
scripts_put_r_git . . . 997
scripts_put_r_projects . . . 998
scripts_put_r_shares_groups . . . 999
scripts_put_r_shares_users . . . 1000
scripts_put_r_transfer . . . 1001
scripts_put_sql . . . 1002
scripts_put_sql_archive . . . 1007
scripts_put_sql_git . . . 1011
scripts_put_sql_projects . . . 1012
scripts_put_sql_shares_groups . . . 1012
scripts_put_sql_shares_users . . . 1013
scripts_put_sql_transfer . . . 1014
search_list . . . 1015
search_list_queries . . . 1017
search_list_types . . . 1018
services_delete . . . 1019
services_delete_deployments . . . 1019
services_delete_projects . . . 1020
services_delete_shares_groups . . . 1020
services_delete_shares_users . . . 1021
services_delete_tokens . . . 1021
services_get . . . 1022
services_get_deployments . . . 1024
services_list . . . 1025
services_list_dependencies . . . 1027

services_list_deployments . . . 1027
services_list_deployments_logs . . . 1029
services_list_projects . . . 1030
services_list_shares . . . 1031
services_list_tokens . . . 1032
services_patch . . . 1032
services_post . . . 1036
services_post_clone . . . 1040
services_post_deployments . . . 1042
services_post_redeploy . . . 1043
services_post_tokens . . . 1044
services_put . . . 1045
services_put_archive . . . 1048
services_put_projects . . . 1051
services_put_shares_groups . . . 1051
services_put_shares_users . . . 1052
services_put_transfer . . . 1053
sql . . . 1054
stash_local_dataframe . . . 1055
storage_hosts_delete . . . 1055
storage_hosts_delete_shares_groups . . . 1056
storage_hosts_delete_shares_users . . . 1056
storage_hosts_get . . . 1057
storage_hosts_list . . . 1057
storage_hosts_list_dependencies . . . 1058
storage_hosts_list_shares . . . 1059
storage_hosts_patch . . . 1060
storage_hosts_post . . . 1061
storage_hosts_put . . . 1062
storage_hosts_put_shares_groups . . . 1063
storage_hosts_put_shares_users . . . 1064
storage_hosts_put_transfer . . . 1065
studios_delete_deployments . . . 1066
studios_get . . . 1066
studios_get_deployments . . . 1068
studios_list_deployments . . . 1069
studios_list_deployments_logs . . . 1071
studios_patch . . . 1072
studios_post . . . 1074
studios_post_deployments . . . 1077
studios_put . . . 1078
studios_put_archive . . . 1081
tables_delete_projects . . . 1083
tables_delete_tags . . . 1083
tables_get . . . 1084
tables_get_enhancements_cass_ncoa . . . 1086
tables_get_enhancements_geocodings . . . 1087
tables_list . . . 1088

tables_list_columns . . . 1090
tables_list_projects . . . 1091
tables_patch . . . 1092
tables_post_enhancements_cass_ncoa . . . 1094
tables_post_enhancements_geocodings . . . 1095
tables_post_refresh . . . 1096
tables_post_scan . . . 1099
tables_put_projects . . . 1099
tables_put_tags . . . 1100
table_tags_delete . . . 1100
table_tags_get . . . 1101
table_tags_list . . . 1101
table_tags_post . . . 1102
templates_delete_reports . . . 1103
templates_delete_reports_shares_groups . . . 1104
templates_delete_reports_shares_users . . . 1104
templates_delete_scripts . . . 1105
templates_delete_scripts_projects . . . 1105
templates_delete_scripts_shares_groups . . . 1106
templates_delete_scripts_shares_users . . . 1106
templates_get_reports . . . 1107
templates_get_scripts . . . 1108
templates_list_reports . . . 1109
templates_list_reports_dependencies . . . 1111
templates_list_reports_shares . . . 1111
templates_list_scripts . . . 1112
templates_list_scripts_dependencies . . . 1114
templates_list_scripts_projects . . . 1114
templates_list_scripts_shares . . . 1115
templates_patch_reports . . . 1116
templates_patch_scripts . . . 1118
templates_post_reports . . . 1119
templates_post_scripts . . . 1121
templates_put_reports . . . 1123
templates_put_reports_shares_groups . . . 1124
templates_put_reports_shares_users . . . 1125
templates_put_reports_transfer . . . 1126
templates_put_scripts . . . 1127
templates_put_scripts_projects . . . 1129
templates_put_scripts_shares_groups . . . 1129
templates_put_scripts_shares_users . . . 1130
templates_put_scripts_transfer . . . 1131
transfer_table . . . 1132
usage_get_llm . . . 1133
usage_limits_get_llm . . . 1134
usage_limits_get_matching . . . 1135
usage_limits_list_llm . . . 1135
usage_limits_list_matching . . . 1136

usage_list_llm 1137
usage_list_llm_organization_summary 1138
usage_list_matching 1138
users_delete_2fa 1139
users_delete_api_keys 1141
users_delete_me_favorites 1142
users_delete_sessions 1143
users_get 1144
users_get_api_keys 1146
users_list 1147
users_list_api_keys 1149
users_list_me 1150
users_list_me_activity 1151
users_list_me_favorites 1152
users_list_me_organization_admins 1153
users_patch 1154
users_patch_me 1157
users_patch_me_favorites_ranking_bottom 1160
users_patch_me_favorites_ranking_higher 1161
users_patch_me_favorites_ranking_lower 1161
users_patch_me_favorites_ranking_top 1162
users_post 1162
users_post_access_email 1165
users_post_api_keys 1166
users_post_me_favorites 1167
users_post_unsuspend 1168
workflows_delete_projects 1168
workflows_delete_shares_groups 1169
workflows_delete_shares_users 1169
workflows_get 1170
workflows_get_executions 1171
workflows_get_executions_tasks 1173
workflows_get_git_commits 1174
workflows_list 1174
workflows_list_dependencies 1176
workflows_list_executions 1177
workflows_list_git 1178
workflows_list_git_commits 1179
workflows_list_projects 1179
workflows_list_shares 1180
workflows_patch 1181
workflows_patch_git 1184
workflows_post 1185
workflows_post_clone 1187
workflows_post_executions 1189
workflows_post_executions_cancel 1191
workflows_post_executions_resume 1192
workflows_post_executions_retry 1193

workflows_post_git_checkout_latest 1194
workflows_post_git_commits 1195
workflows_put 1195
workflows_put_archive 1198
workflows_put_git 1199
workflows_put_projects 1201
workflows_put_shares_groups 1201
workflows_put_shares_users 1202
workflows_put_transfer 1203
write_chunks 1204
write_civis 1205
write_civis_file 1207
write_job_output 1209

Index **1210**

`admin_list_organizations`
List organizations

Description

List organizations

Usage

```
admin_list_organizations(status = NULL, org_type = NULL)
```

Arguments

`status`	array optional. The status of the organization (active/trial/inactive).
`org_type`	array optional. The organization type (platform/ads/survey_vendor/other).

Value

An array containing the following fields:

`id`	integer, The ID of this organization.
`name`	string, The name of this organization.
`slug`	string, The slug of this organization.
`accountManagerId`	integer, The user ID of the Account Manager.
`csSpecialistId`	integer, The user ID of the Client Success Specialist.
`status`	string, The status of the organization (active/trial/inactive).
`orgType`	string, The organization type (platform/ads/survey_vendor/other).
`customBranding`	string, The custom branding settings.

`contractSize` integer, The monthly contract size.

`maxAnalystUsers`
integer, The max number of full platform users for the org.

`maxReportUsers` integer, The max number of report-only platform users for the org.

`vertical` string, The business vertical that the organization belongs to.

`csMetadata` string, Additional metadata about the organization in JSON format.

`removeFooterInEmails`
boolean, If true, emails sent by platform will not include Civis text.

`salesforceAccountId`
string, The Salesforce Account ID for this organization.

`tableauSiteId` string, The Tableau Site ID for this organization.

`fedrampEnabled` boolean, Flag denoting whether this organization is FedRAMP compliant.

`createdById` integer, The ID of the user who created this organization

`lastUpdatedById`
integer, The ID of the user who last updated this organization

`advancedSettings`
list, A list containing the following elements:

- dedicatedDjPoolEnabled boolean, If true, the Organization has a dedicated delayed jobs pool. Defaults to false.

`tableauRefreshHistory`
array, The number of tableau refreshes used this month.

`aliases_delete` *Delete an alias*

Description

Delete an alias

Usage

```
aliases_delete(id)
```

Arguments

`id` integer required. The id of the Alias object.

Value

An empty HTTP response

`aliases_delete_shares_groups`
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
aliases_delete_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

`aliases_delete_shares_users`
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
aliases_delete_shares_users(id, user_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. The ID of the user.

Value

An empty HTTP response

aliases_get *Get an Alias*

Description

Get an Alias

Usage

```
aliases_get(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The id of the Alias object.
objectId	integer, The id of the object
objectType	string, The type of the object. Valid types include: cass_ncoa, container_script, geocode, identity_resolution, dbt_script, python_script, r_script, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
alias	string, The alias of the object
userId	integer, The id of the user who created the alias
displayName	string, The display name of the Alias object. Defaults to object name if not provided.

aliases_get_object_type *Get details about an alias within an FCO type*

Description

Get details about an alias within an FCO type

Usage

```
aliases_get_object_type(object_type, alias)
```

Arguments

object_type	string required. The type of the object. Valid types include: cass_ncoa, container_script, geocode, identity_resolution, dbt_script, python_script, r_script, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
alias	string required. The alias of the object

Value

A list containing the following elements:

id	integer, The id of the Alias object.
objectId	integer, The id of the object
objectType	string, The type of the object. Valid types include: cass_ncoa, container_script, geocode, identity_resolution, dbt_script, python_script, r_script, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
alias	string, The alias of the object
userId	integer, The id of the user who created the alias
displayName	string, The display name of the Alias object. Defaults to object name if not provided.

---

aliases_list *List Aliases*

Description

List Aliases

Usage

```
aliases_list(
  object_type = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

object_type	string optional. Filter results by object type. Pass multiple object types with a comma-separatedlist. Valid types include: cass_ncoa, container_script, geocode, identity_resolution, dbt_script, python_script, r_script, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.

limit	integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id, object_type.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id	integer, The id of the Alias object.
objectId	integer, The id of the object
objectType	string, The type of the object. Valid types include: cass_ncoa, container_script, geocode, identity_resolution, dbt_script, python_script, r_script, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
alias	string, The alias of the object
userId	integer, The id of the user who created the alias
displayName	string, The display name of the Alias object. Defaults to object name if not provided.

---

aliases_list_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
aliases_list_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

`objectType`	string, Dependent object type
`fcoType`	string, Human readable dependent object type
`id`	integer, Dependent object ID
`name`	string, Dependent object name, or nil if the requesting user cannot read this object
`permissionLevel`	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
`description`	string, Additional information about the dependency, if relevant
`shareable`	boolean, Whether or not the requesting user can share this object.

`aliases_list_shares` *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
aliases_list_shares(id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,

totalUserShares
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`aliases_patch` *Update some attributes of this Alias*

Description

Update some attributes of this Alias

Usage

```
aliases_patch(
  id,
  object_id = NULL,
  object_type = NULL,
  alias = NULL,
  display_name = NULL
)
```

Arguments

`id`	integer required. The id of the Alias object.
`object_id`	integer optional. The id of the object
`object_type`	string optional. The type of the object. Valid types include: cass_ncoa, container_script, geocode, identity_resolution, dbt_script, python_script, r_script, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
`alias`	string optional. The alias of the object
`display_name`	string optional. The display name of the Alias object. Defaults to object name if not provided.

Value

A list containing the following elements:

`id`	integer, The id of the Alias object.
`objectId`	integer, The id of the object
`objectType`	string, The type of the object. Valid types include: cass_ncoa, container_script, geocode, identity_resolution, dbt_script, python_script, r_script, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.

alias	string, The alias of the object
userId	integer, The id of the user who created the alias
displayName	string, The display name of the Alias object. Defaults to object name if not provided.

`aliases_post` *Create an Alias*

Description

Create an Alias

Usage

```
aliases_post(object_id, object_type, alias, display_name = NULL)
```

Arguments

object_id	integer required. The id of the object
object_type	string required. The type of the object. Valid types include: cass_ncoa, container_script, geocode, identity_resolution, dbt_script, python_script, r_script, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
alias	string required. The alias of the object
display_name	string optional. The display name of the Alias object. Defaults to object name if not provided.

Value

A list containing the following elements:

id	integer, The id of the Alias object.
objectId	integer, The id of the object
objectType	string, The type of the object. Valid types include: cass_ncoa, container_script, geocode, identity_resolution, dbt_script, python_script, r_script, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
alias	string, The alias of the object
userId	integer, The id of the user who created the alias
displayName	string, The display name of the Alias object. Defaults to object name if not provided.

aliases_put *Replace all attributes of this Alias*

Description

Replace all attributes of this Alias

Usage

```
aliases_put(id, object_id, object_type, alias, display_name = NULL)
```

Arguments

id	integer required. The id of the Alias object.
object_id	integer required. The id of the object
object_type	string required. The type of the object. Valid types include: cass_ncoa, container_script, geocode, identity_resolution, dbt_script, python_script, r_script, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
alias	string required. The alias of the object
display_name	string optional. The display name of the Alias object. Defaults to object name if not provided.

Value

A list containing the following elements:

id	integer, The id of the Alias object.
objectId	integer, The id of the object
objectType	string, The type of the object. Valid types include: cass_ncoa, container_script, geocode, identity_resolution, dbt_script, python_script, r_script, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
alias	string, The alias of the object
userId	integer, The id of the user who created the alias
displayName	string, The display name of the Alias object. Defaults to object name if not provided.

`aliases_put_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
aliases_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`aliases_put_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
aliases_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_ids`	array required. An array of one or more user IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`aliases_put_transfer` *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
aliases_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies`
: array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object

- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`announcements_list` *List announcements*

Description

List announcements

Usage

```
announcements_list(
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

`limit`	integer optional. Number of results to return. Defaults to 10. Maximum allowed is 50.
`page_num`	integer optional. Page number of the results to return. Defaults to the first page, 1.
`order`	string optional. The field on which to order the result set. Defaults to released_at. Must be one of: released_at.
`order_dir`	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

`id`	integer, The ID of this announcement
`subject`	string, The subject of this announcement.
`body`	string, The body of this announcement.
`releasedAt`	string, The date and time this announcement was released.
`createdAt`	string,
`updatedAt`	string,

await *Call a function repeatedly until a status is reached.*

Description

`await` repeatedly calls a Civis API endpoint such as `scripts_get_sql_runs` that monitors the status of a script, job, import, or model. It blocks until the function returns a result with a successful or error status. If the script, job, import or model results in an error state, `await` throws an error with useful debugging information.

`await_all` is a vectorized version of `await`. It repeatedly calls a Civis API endpoint for all values of a vector, e.g. a vector of script, job, import, run, or model ids. It blocks until all calls have returned a result with a given status, and silently captures jobs that return errors.

Usage

```
await(
  f,
  ...,
  .status_key = "state",
  .success_states = c("succeeded", "success"),
  .error_states = c("failed", "cancelled"),
  .timeout = NULL,
  .interval = getOption("civis.default_polling_interval"),
  .verbose = FALSE
)

await_all(
  f,
  .x,
  .y = NULL,
  ...,
  .status_key = "state",
  .success_states = c("succeeded", "success"),
  .error_states = c("failed", "cancelled"),
  .timeout = NULL,
  .interval = NULL,
  .verbose = FALSE
)
```

Arguments

`f`	function to be called repeatedly until a status is reached.
`...`	arguments to `f`
`.status_key`	The name of the element of the list returned by `f` containing the status. For most Civis API endpoints, this is the default, `"state"`.

.success_states	list of states indicating remote work has completed successfully. For most Civis API endpoints, this set of states is the default, c("succeeded", "success").
.error_states	list of states indicating remote work is in an error state. For most Civis API endpoints, this set of states is the default, c("failed", "cancelled").
.timeout	Number of seconds after which to timeout.
.interval	The interval for retries (in seconds). If NULL (default), use exponentially increasing intervals with jitter (see 'Details')
.verbose	Print the status of f at a given retry with the retry time (default FALSE)
.x	a vector of values to be passed to f
.y	a vector of values to be passed to f (default NULL)

Details

await and await_all can wrap Civis API endpoints in generated_client.R. The default values for .status_key, .success_states, and .error_states are suitable for most endpoints. The final status of f can be obtained using get_status.

If an error state is reached, await throws a civis_await_error. await_all silently captures and returns a civis_await_error for any job reaching an error state as an element in the list of results.

If .timeout is specified and the job fails to reach a success state within the time limit, await throws a civis_timeout_error. Likewise, await_all throws a civis_timeout_error if all jobs fail to reach a success state within the time limit.

These errors can be caught using try or tryCatch. Useful debugging information can be returned using get_error and fetch_logs.

The set of possible states for jobs on Civis platform are: "succeeded", "success", "failed", "queued", "running", and "cancelled".

Unless .interval is specified, retries are attempted with exponentially increasing intervals using .25 * (1.2^i)) + runif(1, 0, .2), where i is the index of the current retry. Approximate intervals for a given number of retries are as follows:

- 1-5: .5s
- 6-10: 1-5s
- 11-19: 5-10s
- 20-29: 10s - 1m

The polling interval can be set to a fixed value globally with options("civis.default_polling_interval" = INTERVAL_IN_SECONDS).

Functions

- await_all(): Call a function repeatedly for all values of a vector until all have reached a completed status

See Also

get_status, get_error, fetch_logs

Examples

```
## Not run:

   # Executing a query
   q_id <- queries_post(db_id, query, n_rows, cred_id)[["id"]]
   r <- await(queries_get, id = q_id)
   get_status(r)

   r <- tryCatch(await(queries_get, id = q_id), error = function(e) e)
   get_error(r)

   r <- try(await(queries_get, id = q_id))
   get_error(r)

   jobs <- c(1234, 5678)
   runs <- c(1234, 5678)
   rs <- await_all(scripts_get_r_runs, .x = jobs, .y = runs)

## End(Not run)
```

---

cancel *Cancel the evaluation of a CivisFuture.*

Description

Cancel the evaluation of a CivisFuture.

Usage

```
cancel(future, ...)
```

Arguments

future	CivisFuture object.
...	unused for CivisFuture.

CivisFuture *Evaluate an expression in Civis Platform*

Description

Evaluate an expression in Civis Platform

Usage

```
CivisFuture(
  expr = NULL,
  envir = parent.frame(),
  substitute = FALSE,
  globals = TRUE,
  packages = NULL,
  lazy = FALSE,
  local = lifecycle::deprecated(),
  gc = FALSE,
  label = NULL,
  required_resources = list(cpu = 1024, memory = 2048, diskSpace = 4),
  docker_image_name = "civisanalytics/datascience-r",
  docker_image_tag = "latest",
  ...
)

## S3 method for class 'CivisFuture'
run(future, ...)

## S3 method for class 'CivisFuture'
result(future, ...)

## S3 method for class 'CivisFuture'
cancel(future, ...)

## S3 method for class 'CivisFuture'
resolved(future, ...)

## S3 method for class 'CivisFuture'
fetch_logs(object, ...)
```

Arguments

expr	An R expression.
envir	The environment from where global objects should be identified.
substitute	If TRUE, argument expr is substitute():ed, otherwise not.
globals	(optional) a logical, a character vector, or a named list to control how globals are handled. For details, see section 'Globals used by future expressions' in the help for future().
packages	(optional) a character vector specifying packages to be attached in the R environment evaluating the future.
lazy	If FALSE (default), the future is resolved eagerly (starting immediately), otherwise not.
local	deprecated as of civis v3.0.1
gc	If TRUE, the garbage collector is run after the value of the future has been collected, i.e., when it is no longer needed or when the future is terminated.

label	An optional character string label attached to the future.
required_resources	resources, see scripts_post_containers
docker_image_name	the image for the container script.
docker_image_tag	the tag for the Docker image.
...	arguments to scripts_post_containers
future	CivisFuture object.
object	CivisFuture

Value

A CivisFuture inheriting from Future that evaluates expr on the given container.

Methods (by generic)

- run(CivisFuture): Run a CivisFuture
- result(CivisFuture): Return the value of a CivisFuture
- cancel(CivisFuture): Cancel a CivisFuture
- resolved(CivisFuture): Check if a CivisFuture has resolved
- fetch_logs(CivisFuture): Fetch logs from a CivisFuture

---

civis_file *A file in the Civis Platform*

Description

Use civis_file to use a file in the Civis Platform with civis_ml.

Usage

```
civis_file(file_id)
```

Arguments

file_id	The id of a Civis file.

Value

A civis_file.

civis_file_manifest *A manifest file in the Civis Platform*

Description

Use `civis_file_manifest` to use a manifest file in the Civis Platform with civis_ml.

Usage

```
civis_file_manifest(file_id)
```

Arguments

`file_id`	The id of a Civis file.

Value

A `civis_file_manifest`

civis_ml *Interface for modeling in the Civis Platform*

Description

An interface for training and scoring data on Civis Platform using a set of Scikit-Learn estimators.

Usage

```
civis_ml(
  x,
  dependent_variable,
  model_type,
  primary_key = NULL,
  excluded_columns = NULL,
  parameters = NULL,
  fit_params = NULL,
  cross_validation_parameters = NULL,
  calibration = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
```

```
  polling_interval = NULL,
  validation_data = c("train", "skip"),
  n_jobs = NULL,
  verbose = FALSE,
  civisml_version = "prod"
)

civis_ml_fetch_existing(model_id, run_id = NULL)

## S3 method for class 'civis_ml'
predict(
  object,
  newdata,
  primary_key = NA,
  output_table = NULL,
  output_db = NULL,
  if_output_exists = c("fail", "append", "drop", "truncate"),
  n_jobs = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  polling_interval = NULL,
  verbose = FALSE,
  dvs_to_predict = NULL,
  ...
)
```

Arguments

x	See the Data Sources section below.
dependent_variable	The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.
model_type	The name of the CivisML workflow. See the Workflows section below.
primary_key	Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the primary_key of the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no primary_key.
excluded_columns	Optional, a vector of columns which will be considered ineligible to be independent variables.
parameters	Optional, parameters for the final stage estimator in a predefined model, e.g. `list(C = 2)` for a "sparse_logistic" model.
fit_params	Optional, a mapping from parameter names in the model's `fit` method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column')`.

cross_validation_parameters
: Optional, parameter grid for learner parameters, e.g. `list(n_estimators = c(100, 200, 500),learning_rate = c(0.01, 0.1), max_depth = c(2, 3))` or `"hyperband"` for supported models.

calibration
: Optional, if not `NULL`, calibrate output probabilities with the selected method, `sigmoid`, or `isotonic`. Valid only with classification models.

oos_scores_table
: Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

oos_scores_db
: Optional, the name of the database where the `oos_scores_table` will be created. If not provided, this will default to `database_name`.

oos_scores_if_exists
: Optional, action to take if `oos_scores_table` already exists. One of `"fail"`, `"append"`, `"drop"`, or `"truncate"`. The default is `"fail"`.

model_name
: Optional, the prefix of the Platform modeling jobs. It will have `" Train"` or `" Predict"` added to become the Script title.

cpu_requested
: Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

memory_requested
: Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

disk_requested
: Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

notifications
: Optional, model status notifications. See `scripts_post_custom` for further documentation about email and URL notification.

polling_interval
: Check for job completion every this number of seconds.

validation_data
: Optional, source for validation data. There are currently two options: `train` (the default), which uses training data for validation, and `skip`, which skips the validation step.

n_jobs
: Number of concurrent Platform jobs to use for training and validation, or multi-file / large table prediction. Defaults to `NULL`, which allows CivisML to dynamically calculate an appropriate number of workers to use (in general, as many as possible without using all resources in the cluster).

verbose
: Optional, If `TRUE`, supply debug outputs in Platform logs and make prediction child jobs visible.

civisml_version
: Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

model_id
: The `id` of CivisML model built previously.

run_id
: Optional, the `id` of a CivisML model run. If `NULL`, defaults to fetching the latest run.

object
: A `civis_ml` object.

newdata	See the Data Sources section below.
output_table	The table in which to put predictions.
output_db	The database containing output_table. If not provided, this will default to the database_name specified when the model was built.
if_output_exists	Action to take if the prediction table already exists. One of "fail", "append", "drop", or "truncate". The default is "fail".
dvs_to_predict	Optional, For scoring, this should be a vector of column names of dependent variables to include in the output table. It must be a subset of the dependent_variable vector provided for training. The scores for the returned subset will be identical to the scores which those outputs would have had if all outputs were written, but ignoring some of the model's outputs will let predictions complete faster and use less disk space. If not provided, the entire model output will be written to the output table.
...	Unused

Value

A civis_ml object, a list containing the following elements:

job	job metadata from scripts_get_custom.
run	run metadata from scripts_get_custom_runs.
outputs	CivisML metadata from scripts_list_custom_runs_outputs containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.
metrics	Parsed CivisML output from metrics.json containing metadata from validation. A list containing the following elements: • run list, metadata about the run. • data list, metadata about the training data. • model list, the fitted scikit-learn model with CV results. • metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc). • warnings list. • data_platform list, training data location.
model_info	Parsed CivisML output from model_info.json containing metadata from training. A list containing the following elements: • run list, metadata about the run. • data list, metadata about the training data. • model list, the fitted scikit-learn model. • metrics empty list. • warnings list. • data_platform list, training data location.

CivisML Workflows

You can use the following pre-defined models with `civis_ml`. All models start by imputing missing values with the mean of non-null values in a column. The `"sparse_*"` models include a LASSO regression step (using `glmnet`) to do feature selection before passing data to the final model. In some models, CivisML uses default parameters from those in Scikit-Learn, as indicated in the "Altered Defaults" column. All models also have `random_state=42`.

Specific workflows can also be called directly using the R workflow functions.

Name	R Workflow	Model Type	A
sparse_logistic	civis_ml_sparse_logistic	classification	LogisticR
gradient_boosting_classifier	civis_ml_gradient_boosting_classifier	classification	GradientBoosting
random_forest_classifier	civis_ml_random_forest_classifier	classification	RandomForest
extra_trees_classifier	civis_ml_extra_trees_classifier	classification	ExtraTrees
multilayer_perceptron_classifier		classification	muffnn.MLP
stacking_classifier		classification	Stacked
sparse_linear_regressor	civis_ml_sparse_linear_regressor	regression	LinearR
sparse_ridge_regressor	civis_ml_sparse_ridge_regressor	regression	
gradient_boosting_regressor	civis_ml_gradient_boosting_regressor	regression	GradientBoosting
random_forest_regressor	civis_ml_random_forest_regressor	regression	RandomForest
extra_trees_regressor	civis_ml_extra_trees_regressor	regression	ExtraTrees
multilayer_perceptron_regressor		regression	muffnn.MLP
stacking_regressor		regression	Stacked

Model names can be easily accessed using the global variables `CIVIS_ML_REGRESSORS` and `CIVIS_ML_CLASSIFIERS`.

Stacking

The `"stacking_classifier"` model stacks together the `"gradient_boosting_classifier"` and `"random_forest_classifier"` predefined models together with a `glmnet.LogitNet(alpha=0, n_splits=4, max_iter=10000, tol=1e-5, scoring='log_loss')`. Defaults for the predefined models are documented in `?civis_ml`. Each column is first standardized, and then the model predictions are combined using LogisticRegressionCV with `penalty='l2'` and `tol=1e-08`. The `"stacking_regressor"` works similarly, stacking together the `"gradient_boosting_regressor"` and `"random_forest_regressor"` models and a `glmnet.ElasticNet(alpha=0, n_splits=4, max_iter=10000, tol=1e-5, scoring='r2')`, combining them using NonNegativeLinearRegression. The estimators that are being stacked have the same names as the associated pre-defined models, and the meta-estimator steps are named "meta-estimator". Note that although default parameters are provided for multilayer perceptron models, it is highly recommended that multilayer perceptrons be run using hyperband.

Hyperparameter Tuning

You can tune hyperparameters using one of two methods: grid search or hyperband. CivisML will perform grid search if you pass a list of hyperparameters to the `cross_validation_parameters` parameter, where list elements are hyperparameter names, and the values are vectors of hyperparameter values to grid search over. You can run hyperparameter optimization in parallel by setting the `n_jobs` parameter to however many jobs you would like to run in parallel. By default, `n_jobs`

is dynamically calculated based on the resources available on your cluster, such that a modeling job will never take up more than 90

Hyperband is an efficient approach to hyperparameter optimization, and recommended over grid search where possible. CivisML will perform hyperband optimization if you pass the string `"hyperband"` to `cross_validation_parameters`. Hyperband is currently only supported for the following models: `"gradient_boosting_classifier"`, `"random_forest_classifier"`, `"extra_trees_classifier"`, `"multilayer_perceptron_classifier"`, `"stacking_classifier"`, `"gradient_boosting_regressor"`, `"random_forest_regressor"`, `"extra_trees_regressor"`, `"multilayer_perceptron_regressor"`, and `"stacking_regressor"`.

Hyperband cannot be used to tune GLMs. For this reason, preset GLMs do not have a hyperband option. Similarly, when `cross_validation_parameters='hyperband'` and the model is `stacking_classifier` or `stacking_regressor`, only the GBT and random forest steps of the stacker are tuned using hyperband. For the specific distributions used in the predefined hyperband models, see the detailed table in the Python client documentation.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

`data.frame` `civis_ml(x = df, ...)`

local csv file `civis_ml(x = "path/to/data.csv", ...)`

file in Civis Platform `civis_ml(x = civis_file(1234))`

table in Civis Platform `civis_ml(x = civis_table(table_name = "schema.table", database_name = "database"))`

Out of sample scores

Model outputs will always contain out-of-sample (or out of fold) scores, which are accessible through `fetch_oos_scores`. These may be stored in a Civis table on Redshift using the `oos_scores`, `oos_scores_db`, and `oos_scores_if_exists` parameters.

Predictions

A fitted model can be used to make predictions for data residing in any of the sources above and a `civis_file_manifest`. Similar to `civis_ml`, use the data source helpers as the `newdata` argument to `predict.civis_ml`.

A manifest file is a JSON file which specifies the location of many shards of the data to be used for prediction. A manifest file is the output of a Civis export job with `force_multifile = TRUE` set, e.g. from `civis_to_multifile_csv`. Large civis tables (provided using `table_name`) will automatically be exported to manifest files.

Prediction outputs will always be stored as gzipped CSVs in one or more civis files. Provide an `output_table` (and optionally an `output_db`, if it's different from `database_name`) to copy these predictions into a table on Redshift.

See Also

civis_file, civis_table, and civis_file_manifest for specifying data sources.

get_metric to access model validation metrics.

fetch_logs for retrieving logs for a (failed) model build, fetch_oos_scores for retrieving the out of sample (fold) scores for each training observation, and fetch_predictions for retrieving the predictions from a prediction job.

Examples

```
## Not run:
# From a data frame:
m <- civis_ml(df, model_type = "sparse_logistic",
              dependent_variable = "Species")

# From a table:
m <- civis_ml(civis_table("schema.table", "database_name"),
              model_type = "sparse_logistic", dependent_variable = "Species",
              oos_scores_table = "schema.scores_table",
              oos_scores_if_exists = "drop")

# From a local file:
m <- civis_ml("path/to/file.csv", model_type = "sparse_logistic",
              dependent_variable = "Species")

# From a Civis file:
file_id <- write_civis_file("path/to/file.csv", name = "file.csv")
m <- civis_ml(civis_file(file_id), model_type = "sparse_logistic",
              dependent_variable = "Species")

pred_job <- predict(m, newdata = df)
pred_job <- predict(m, civis_table("schema.table", "database_name"),
                    output_table = "schema.scores_table")
pred_job <- predict(m, civis_file(file_id),
                    output_table = "schema.scores_table")

m <- civis_ml_fetch_existing(model_id = m$job$id, m$run$id)
logs <- fetch_logs(m)
yhat <- fetch_oos_scores(m)
yhat <- fetch_predictions(pred_job)

## End(Not run)
```

CIVIS_ML_CLASSIFIERS *List of classification models.*

Description

List of classification models.

Usage

```
CIVIS_ML_CLASSIFIERS
```

Format

An object of class character of length 6.

`civis_ml_extra_trees_classifier` *CivisML Extra Trees Classifier*

Description

CivisML Extra Trees Classifier

Usage

```
civis_ml_extra_trees_classifier(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  n_estimators = 500,
  criterion = c("gini", "entropy"),
  max_depth = NULL,
  min_samples_split = 2,
  min_samples_leaf = 1,
  min_weight_fraction_leaf = 0,
  max_features = "sqrt",
  max_leaf_nodes = NULL,
  min_impurity_split = 1e-07,
  bootstrap = FALSE,
  random_state = 42,
  class_weight = NULL,
  fit_params = NULL,
  cross_validation_parameters = NULL,
  calibration = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
  verbose = FALSE,
```

```
  civisml_version = "prod"
)
```

Arguments

x	See the Data Sources section below.
dependent_variable	The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.
primary_key	Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the primary_key of the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no primary_key.
excluded_columns	Optional, a vector of columns which will be considered ineligible to be independent variables.
n_estimators	The number of boosting stages to perform. Gradient boosting is fairly robust to over-fitting, so a large number usually results in better predictive performance.
criterion	The function to measure the quality of a split. Supported criteria are `gini` for the Gini impurity and `entropy` for the information gain.
max_depth	Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance. The best value depends on the interaction of the input variables.
min_samples_split	The minimum number of samples required to split an internal node. If an integer, then consider `min_samples_split` as the minimum number. If a float, then `min_samples_split` is a percentage and `ceiling(min_samples_split * n_samples)` are the minimum number of samples for each split.
min_samples_leaf	The minimum number of samples required to be in a leaf node. If an integer, then consider `min_samples_leaf` as the minimum number. If a float, the `min_samples_leaf` is a percentage and `ceiling(min_samples_leaf * n_samples)` are the minimum number of samples for each leaf node.
min_weight_fraction_leaf	The minimum weighted fraction of the sum total of weights required to be at a leaf node.
max_features	The number of features to consider when looking for the best split. **integer** consider `max_features` at each split. **float** then `max_features` is a percentage and `max_features * n_features` are considered at each split. **auto** then `max_features = sqrt(n_features)` **sqrt** then `max_features = sqrt(n_features)` **log2** then `max_features = log2(n_features)` **NULL** then `max_features = n_features`

`max_leaf_nodes` Grow trees with `max_leaf_nodes` in best-first fashion. Best nodes are defined as relative reduction to impurity. If `max_leaf_nodes = NULL` then unlimited number of leaf nodes.

`min_impurity_split`
Threshold for early stopping in tree growth. A node will split if its impurity is above the threshold, otherwise it is a leaf.

`bootstrap` Whether bootstrap samples are used when building trees.

`random_state` The seed of the random number generator.

`class_weight` A `list` with `class_label = value` pairs, or `balanced`. When `class_weight = "balanced"`, the class weights will be inversely proportional to the class frequencies in the input data as:

$$\frac{n_samples}{n_classes * table(y)}$$

Note, the class weights are multiplied with `sample_weight` (passed via `fit_params`) if `sample_weight` is specified.

`fit_params` Optional, a mapping from parameter names in the model's `fit` method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column')`.

`cross_validation_parameters`
Optional, parameter grid for learner parameters, e.g. `list(n_estimators = c(100, 200, 500),learning_rate = c(0.01, 0.1), max_depth = c(2, 3))` or `"hyperband"` for supported models.

`calibration` Optional, if not `NULL`, calibrate output probabilities with the selected method, `sigmoid`, or `isotonic`. Valid only with classification models.

`oos_scores_table`
Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

`oos_scores_db` Optional, the name of the database where the `oos_scores_table` will be created. If not provided, this will default to `database_name`.

`oos_scores_if_exists`
Optional, action to take if `oos_scores_table` already exists. One of `"fail"`, `"append"`, `"drop"`, or `"truncate"`. The default is `"fail"`.

`model_name` Optional, the prefix of the Platform modeling jobs. It will have `" Train"` or `" Predict"` added to become the Script title.

`cpu_requested` Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

`memory_requested`
Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

`disk_requested` Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

`notifications` Optional, model status notifications. See `scripts_post_custom` for further documentation about email and URL notification.

`polling_interval`
: Check for job completion every this number of seconds.

`verbose`
: Optional, If `TRUE`, supply debug outputs in Platform logs and make prediction child jobs visible.

`civisml_version`
: Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value

A `civis_ml` object, a list containing the following elements:

`job`
: job metadata from `scripts_get_custom`.

`run`
: run metadata from `scripts_get_custom_runs`.

`outputs`
: CivisML metadata from `scripts_list_custom_runs_outputs` containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

`metrics`
: Parsed CivisML output from `metrics.json` containing metadata from validation. A list containing the following elements:
  - run list, metadata about the run.
  - data list, metadata about the training data.
  - model list, the fitted scikit-learn model with CV results.
  - metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
  - warnings list.
  - data_platform list, training data location.

`model_info`
: Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:
  - run list, metadata about the run.
  - data list, metadata about the training data.
  - model list, the fitted scikit-learn model.
  - metrics empty list.
  - warnings list.
  - data_platform list, training data location.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

`data.frame` `civis_ml(x = df, ...)`

local csv file `civis_ml(x = "path/to/data.csv", ...)`

file in Civis Platform `civis_ml(x = civis_file(1234))`

table in Civis Platform `civis_ml(x = civis_table(table_name = "schema.table", database_name = "database"))`

Examples

```
## Not run:
 df <- iris
 names(df) <- gsub("\\.", "_", names(df))

 m <- civis_ml_extra_trees_classifier(df,
   dependent_variable = "Species",
   n_estimators = 100,
   max_depth = 5,
   max_features = NULL)
 yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(
   n_estimators = c(100, 200, 500),
   max_depth = c(2, 3))

 m <- civis_ml_extra_trees_classifier(df,
   dependent_variable = "Species",
   max_features = NULL,
   cross_validation_parameters = cv_params)

pred_info <- predict(m,  civis_table("schema.table", "my_database"),
   output_table = "schema.scores_table")

## End(Not run)
```

---

civis_ml_extra_trees_regressor

CivisML Extra Trees Regressor

Description

CivisML Extra Trees Regressor

Usage

```
civis_ml_extra_trees_regressor(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  n_estimators = 500,
  criterion = c("mse", "mae"),
  max_depth = NULL,
  min_samples_split = 2,
  min_samples_leaf = 1,
  min_weight_fraction_leaf = 0,
```

```
  max_features = "sqrt",
  max_leaf_nodes = NULL,
  min_impurity_split = 1e-07,
  bootstrap = FALSE,
  random_state = 42,
  fit_params = NULL,
  cross_validation_parameters = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
  verbose = FALSE,
  civisml_version = "prod"
)
```

Arguments

x	See the Data Sources section below.
dependent_variable	The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.
primary_key	Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the primary_key of the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no primary_key.
excluded_columns	Optional, a vector of columns which will be considered ineligible to be independent variables.
n_estimators	The number of boosting stages to perform. Gradient boosting is fairly robust to over-fitting, so a large number usually results in better predictive performance.
criterion	The function used to measure the quality of a split. Supported criteria are `mse` for the mean squared error, and `mae` for the mean absolute error.
max_depth	Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance. The best value depends on the interaction of the input variables.
min_samples_split	The minimum number of samples required to split an internal node. If an integer, then consider `min_samples_split` as the minimum number. If a float, then `min_samples_split` is a percentage and `ceiling(min_samples_split * n_samples)` are the minimum number of samples for each split.

`min_samples_leaf`
: The minimum number of samples required to be in a leaf node. If an integer, then consider `min_samples_leaf` as the minimum number. If a float, the `min_samples_leaf` is a percentage and `ceiling(min_samples_leaf * n_samples)` are the minimum number of samples for each leaf node.

`min_weight_fraction_leaf`
: The minimum weighted fraction of the sum total of weights required to be at a leaf node.

`max_features`
: The number of features to consider when looking for the best split.

  integer consider `max_features` at each split.

  float then `max_features` is a percentage and `max_features * n_features` are considered at each split.

  auto then `max_features = sqrt(n_features)`

  sqrt then `max_features = sqrt(n_features)`

  log2 then `max_features = log2(n_features)`

  NULL then `max_features = n_features`

`max_leaf_nodes`
: Grow trees with `max_leaf_nodes` in best-first fashion. Best nodes are defined as relative reduction to impurity. If `max_leaf_nodes = NULL` then unlimited number of leaf nodes.

`min_impurity_split`
: Threshold for early stopping in tree growth. A node will split if its impurity is above the threshold, otherwise it is a leaf.

`bootstrap`
: Whether bootstrap samples are used when building trees.

`random_state`
: The seed of the random number generator.

`fit_params`
: Optional, a mapping from parameter names in the model's `fit` method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column')`.

`cross_validation_parameters`
: Optional, parameter grid for learner parameters, e.g. `list(n_estimators = c(100, 200, 500),learning_rate = c(0.01, 0.1), max_depth = c(2, 3))` or `"hyperband"` for supported models.

`oos_scores_table`
: Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

`oos_scores_db`
: Optional, the name of the database where the `oos_scores_table` will be created. If not provided, this will default to `database_name`.

`oos_scores_if_exists`
: Optional, action to take if `oos_scores_table` already exists. One of `"fail"`, `"append"`, `"drop"`, or `"truncate"`. The default is `"fail"`.

`model_name`
: Optional, the prefix of the Platform modeling jobs. It will have `" Train"` or `" Predict"` added to become the Script title.

`cpu_requested`
: Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

`memory_requested`
: Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

`disk_requested` Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

`notifications` Optional, model status notifications. See `scripts_post_custom` for further documentation about email and URL notification.

`polling_interval`
Check for job completion every this number of seconds.

`verbose` Optional, If `TRUE`, supply debug outputs in Platform logs and make prediction child jobs visible.

`civisml_version`
Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value

A `civis_ml` object, a list containing the following elements:

`job` job metadata from `scripts_get_custom`.

`run` run metadata from `scripts_get_custom_runs`.

`outputs` CivisML metadata from `scripts_list_custom_runs_outputs` containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

`metrics` Parsed CivisML output from `metrics.json` containing metadata from validation. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model with CV results.
- metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
- warnings list.
- data_platform list, training data location.

`model_info` Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model.
- metrics empty list.
- warnings list.
- data_platform list, training data location.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

`data.frame` `civis_ml(x = df, ...)`

local csv file `civis_ml(x = "path/to/data.csv", ...)`

file in Civis Platform `civis_ml(x = civis_file(1234))`

table in Civis Platform `civis_ml(x = civis_table(table_name = "schema.table", database_name = "database"))`

Examples

```
## Not run:
data(ChickWeight)

m <- civis_ml_extra_trees_regressor(ChickWeight,
  dependent_variable = "weight",
  n_estimators = 100,
  max_depth = 5,
  max_features = NULL)
yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(
  n_estimators = c(100, 200, 500),
  max_depth = c(2, 3))

m <- civis_ml_extra_trees_regressor(ChickWeight,
  dependent_variable = "weight",
  max_features = NULL,
  cross_validation_parameters = cv_params)

pred_info <- predict(m,  civis_table("schema.table", "my_database"),
   output_table = "schema.scores_table")

## End(Not run)
```

---

civis_ml_gradient_boosting_classifier

CivisML Gradient Boosting Classifier

Description

CivisML Gradient Boosting Classifier

Usage

```
civis_ml_gradient_boosting_classifier(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
```

```
  loss = c("deviance", "exponential"),
  learning_rate = 0.1,
  n_estimators = 500,
  subsample = 1,
  criterion = c("friedman_mse", "mse", "mae"),
  min_samples_split = 2,
  min_samples_leaf = 1,
  min_weight_fraction_leaf = 0,
  max_depth = 2,
  min_impurity_split = 1e-07,
  random_state = 42,
  max_features = "sqrt",
  max_leaf_nodes = NULL,
  presort = c("auto", TRUE, FALSE),
  fit_params = NULL,
  cross_validation_parameters = NULL,
  calibration = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
  verbose = FALSE,
  civisml_version = "prod"
)
```

Arguments

x	See the Data Sources section below.
dependent_variable	The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.
primary_key	Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the primary_key of the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no primary_key.
excluded_columns	Optional, a vector of columns which will be considered ineligible to be independent variables.
loss	The loss function to be optimized. `deviance` refers to deviance (logistic regression) for classification with probabilistic outputs. For `exponential`, gradient boosting recovers the AdaBoost algorithm.

`learning_rate` The learning rate shrinks the contribution of each tree by `learning_rate`. There is a trade-off between `learning_rate` and `n_estimators`.

`n_estimators` The number of boosting stages to perform. Gradient boosting is fairly robust to over-fitting, so a large number usually results in better predictive performance.

`subsample` The fraction of samples to be used for fitting individual base learners. If smaller than 1.0, this results in Stochastic Gradient Boosting. `subsample` interacts with the parameter `n_estimators`. Choosing `subsample < 1.0` leads to a reduction of variance and an increase in bias.

`criterion` The function to measure the quality of a split. The default value `criterion = "friedman_mse"` is generally the best as it can provide a better approximation in some cases.

`min_samples_split`
The minimum number of samples required to split an internal node. If an integer, then consider `min_samples_split` as the minimum number. If a float, then `min_samples_split` is a percentage and `ceiling(min_samples_split * n_samples)` are the minimum number of samples for each split.

`min_samples_leaf`
The minimum number of samples required to be in a leaf node. If an integer, then consider `min_samples_leaf` as the minimum number. If a float, the `min_samples_leaf` is a percentage and `ceiling(min_samples_leaf * n_samples)` are the minimum number of samples for each leaf node.

`min_weight_fraction_leaf`
The minimum weighted fraction of the sum total of weights required to be at a leaf node.

`max_depth` Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance. The best value depends on the interaction of the input variables.

`min_impurity_split`
Threshold for early stopping in tree growth. A node will split if its impurity is above the threshold, otherwise it is a leaf.

`random_state` The seed of the random number generator.

`max_features` The number of features to consider when looking for the best split.

integer consider `max_features` at each split.

float then `max_features` is a percentage and `max_features * n_features` are considered at each split.

auto then `max_features = sqrt(n_features)`

sqrt then `max_features = sqrt(n_features)`

log2 then `max_features = log2(n_features)`

NULL then `max_features = n_features`

`max_leaf_nodes` Grow trees with `max_leaf_nodes` in best-first fashion. Best nodes are defined as relative reduction to impurity. If `max_leaf_nodes = NULL` then unlimited number of leaf nodes.

`presort` Whether to presort the data to speed up the finding of best splits in fitting.

`fit_params` Optional, a mapping from parameter names in the model's `fit` method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column')`.

cross_validation_parameters
: Optional, parameter grid for learner parameters, e.g. `list(n_estimators = c(100, 200, 500),learning_rate = c(0.01, 0.1), max_depth = c(2, 3))` or `"hyperband"` for supported models.

calibration
: Optional, if not `NULL`, calibrate output probabilities with the selected method, `sigmoid`, or `isotonic`. Valid only with classification models.

oos_scores_table
: Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

oos_scores_db
: Optional, the name of the database where the `oos_scores_table` will be created. If not provided, this will default to `database_name`.

oos_scores_if_exists
: Optional, action to take if `oos_scores_table` already exists. One of `"fail"`, `"append"`, `"drop"`, or `"truncate"`. The default is `"fail"`.

model_name
: Optional, the prefix of the Platform modeling jobs. It will have `" Train"` or `" Predict"` added to become the Script title.

cpu_requested
: Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

memory_requested
: Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

disk_requested
: Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

notifications
: Optional, model status notifications. See `scripts_post_custom` for further documentation about email and URL notification.

polling_interval
: Check for job completion every this number of seconds.

verbose
: Optional, If `TRUE`, supply debug outputs in Platform logs and make prediction child jobs visible.

civisml_version
: Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value

A `civis_ml` object, a list containing the following elements:

job
: job metadata from `scripts_get_custom`.

run
: run metadata from `scripts_get_custom_runs`.

outputs
: CivisML metadata from `scripts_list_custom_runs_outputs` containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

metrics
: Parsed CivisML output from `metrics.json` containing metadata from validation. A list containing the following elements:

  - run list, metadata about the run.

- data list, metadata about the training data.
- model list, the fitted scikit-learn model with CV results.
- metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
- warnings list.
- data_platform list, training data location.

`model_info` Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model.
- metrics empty list.
- warnings list.
- data_platform list, training data location.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

`data.frame` `civis_ml(x = df, ...)`

local csv file `civis_ml(x = "path/to/data.csv", ...)`

file in Civis Platform `civis_ml(x = civis_file(1234))`

table in Civis Platform `civis_ml(x = civis_table(table_name = "schema.table", database_name = "database"))`

Examples

```
## Not run:
 df <- iris
 names(df) <- gsub("\\.", "_", names(df))

 m <- civis_ml_gradient_boosting_classifier(df,
   dependent_variable = "Species",
   learning_rate = .01,
   n_estimators = 100,
   subsample = .5,
   max_depth = 5,
   max_features = NULL)
 yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(
   n_estimators = c(100, 200, 500),
   learning_rate = c(.01, .1),
   max_depth = c(2, 3))
```

```
m <- civis_ml_gradient_boosting_classifier(df,
   dependent_variable = "Species",
   subsample = .5,
   max_features = NULL,
   cross_validation_parameters = cv_params)

pred_info <- predict(m,  civis_table("schema.table", "my_database"),
   output_table = "schema.scores_table")

## End(Not run)
```

---

civis_ml_gradient_boosting_regressor
CivisML Gradient Boosting Regressor

Description

CivisML Gradient Boosting Regressor

Usage

```
civis_ml_gradient_boosting_regressor(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  loss = c("ls", "lad", "huber", "quantile"),
  learning_rate = 0.1,
  n_estimators = 500,
  subsample = 1,
  criterion = c("friedman_mse", "mse", "mae"),
  min_samples_split = 2,
  min_samples_leaf = 1,
  min_weight_fraction_leaf = 0,
  max_depth = 2,
  min_impurity_split = 1e-07,
  random_state = 42,
  max_features = "sqrt",
  alpha = 0.9,
  max_leaf_nodes = NULL,
  presort = c("auto", TRUE, FALSE),
  fit_params = NULL,
  cross_validation_parameters = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
```

```
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
  verbose = FALSE,
  civisml_version = "prod"
)
```

Arguments

x	See the Data Sources section below.
dependent_variable	The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.
primary_key	Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the primary_key of the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no primary_key.
excluded_columns	Optional, a vector of columns which will be considered ineligible to be independent variables.
loss	The loss function to be optimized. `ls` refers to least squares regression. `lad` (least absolute deviation) is a highly robust loss function solely based on order information of the input variables. `huber` is a combination of the two. `quantile` allows quantile regression (use `alpha` to specify the quantile).
learning_rate	The learning rate shrinks the contribution of each tree by `learning_rate`. There is a trade-off between `learning_rate` and `n_estimators`.
n_estimators	The number of boosting stages to perform. Gradient boosting is fairly robust to over-fitting, so a large number usually results in better predictive performance.
subsample	The fraction of samples to be used for fitting individual base learners. If smaller than 1.0, this results in Stochastic Gradient Boosting. `subsample` interacts with the parameter `n_estimators`. Choosing `subsample < 1.0` leads to a reduction of variance and an increase in bias.
criterion	The function to measure the quality of a split. The default value `criterion = "friedman_mse"` is generally the best as it can provide a better approximation in some cases.
min_samples_split	The minimum number of samples required to split an internal node. If an integer, then consider `min_samples_split` as the minimum number. If a float, then `min_samples_split` is a percentage and `ceiling(min_samples_split * n_samples)` are the minimum number of samples for each split.
min_samples_leaf	The minimum number of samples required to be in a leaf node. If an integer, then consider `min_samples_leaf` as the minimum number. If a float, the `min_samples_leaf` is a percentage and `ceiling(min_samples_leaf * n_samples)` are the minimum number of samples for each leaf node.

`min_weight_fraction_leaf`
: The minimum weighted fraction of the sum total of weights required to be at a leaf node.

`max_depth`
: Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance. The best value depends on the interaction of the input variables.

`min_impurity_split`
: Threshold for early stopping in tree growth. A node will split if its impurity is above the threshold, otherwise it is a leaf.

`random_state`
: The seed of the random number generator.

`max_features`
: The number of features to consider when looking for the best split.

  integer consider `max_features` at each split.

  float then `max_features` is a percentage and `max_features * n_features` are considered at each split.

  auto then `max_features = sqrt(n_features)`

  sqrt then `max_features = sqrt(n_features)`

  log2 then `max_features = log2(n_features)`

  NULL then `max_features = n_features`

`alpha`
: The alpha-quantile of the `huber` loss function and the `quantile` loss function. Ignored unless `loss = "huber"` or `loss = "quantile"`

`max_leaf_nodes`
: Grow trees with `max_leaf_nodes` in best-first fashion. Best nodes are defined as relative reduction to impurity. If `max_leaf_nodes = NULL` then unlimited number of leaf nodes.

`presort`
: Whether to presort the data to speed up the finding of best splits in fitting.

`fit_params`
: Optional, a mapping from parameter names in the model's `fit` method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column')`.

`cross_validation_parameters`
: Optional, parameter grid for learner parameters, e.g. `list(n_estimators = c(100, 200, 500),learning_rate = c(0.01, 0.1), max_depth = c(2, 3))` or `"hyperband"` for supported models.

`oos_scores_table`
: Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

`oos_scores_db`
: Optional, the name of the database where the `oos_scores_table` will be created. If not provided, this will default to `database_name`.

`oos_scores_if_exists`
: Optional, action to take if `oos_scores_table` already exists. One of `"fail"`, `"append"`, `"drop"`, or `"truncate"`. The default is `"fail"`.

`model_name`
: Optional, the prefix of the Platform modeling jobs. It will have `" Train"` or `" Predict"` added to become the Script title.

`cpu_requested`
: Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

`memory_requested`
: Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

`disk_requested` Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

`notifications` Optional, model status notifications. See `scripts_post_custom` for further documentation about email and URL notification.

`polling_interval`
Check for job completion every this number of seconds.

`verbose` Optional, If `TRUE`, supply debug outputs in Platform logs and make prediction child jobs visible.

`civisml_version`
Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value

A `civis_ml` object, a list containing the following elements:

`job` job metadata from `scripts_get_custom`.

`run` run metadata from `scripts_get_custom_runs`.

`outputs` CivisML metadata from `scripts_list_custom_runs_outputs` containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

`metrics` Parsed CivisML output from `metrics.json` containing metadata from validation. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model with CV results.
- metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
- warnings list.
- data_platform list, training data location.

`model_info` Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model.
- metrics empty list.
- warnings list.
- data_platform list, training data location.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

`data.frame` `civis_ml(x = df, ...)`

local csv file `civis_ml(x = "path/to/data.csv", ...)`

file in Civis Platform `civis_ml(x = civis_file(1234))`

table in Civis Platform `civis_ml(x = civis_table(table_name = "schema.table", database_name = "database"))`

Examples

```
## Not run:
data(ChickWeight)

m <- civis_ml_gradient_boosting_regressor(ChickWeight,
  dependent_variable = "weight",
  learning_rate = .01,
  n_estimators = 100,
  subsample = .5,
  max_depth = 5,
  max_features = NULL)
yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(
  n_estimators = c(100, 200, 500),
  learning_rate = c(.01, .1),
  max_depth = c(2, 3))

m <- civis_ml_gradient_boosting_regressor(ChickWeight,
  dependent_variable = "weight",
  subsample = .5,
  max_features = NULL,
  cross_validation_parameters = cv_params)

pred_info <- predict(m,  civis_table("schema.table", "my_database"),
   output_table = "schema.scores_table")

## End(Not run)
```

`civis_ml_random_forest_classifier`

CivisML Random Forest Classifier

Description

CivisML Random Forest Classifier

Usage

```
civis_ml_random_forest_classifier(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  n_estimators = 500,
  criterion = c("gini", "entropy"),
  max_depth = NULL,
  min_samples_split = 2,
  min_samples_leaf = 1,
  min_weight_fraction_leaf = 0,
  max_features = "sqrt",
  max_leaf_nodes = NULL,
  min_impurity_split = 1e-07,
  bootstrap = TRUE,
  random_state = 42,
  class_weight = NULL,
  fit_params = NULL,
  cross_validation_parameters = NULL,
  calibration = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
  verbose = FALSE,
  civisml_version = "prod"
)
```

Arguments

x	See the Data Sources section below.
dependent_variable	The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.
primary_key	Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the primary_key of the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no primary_key.
excluded_columns	Optional, a vector of columns which will be considered ineligible to be independent variables.

`n_estimators` The number of boosting stages to perform. Gradient boosting is fairly robust to over-fitting, so a large number usually results in better predictive performance.

`criterion` The function to measure the quality of a split. Supported criteria are `gini` for the Gini impurity and `entropy` for the information gain.

`max_depth` Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance. The best value depends on the interaction of the input variables.

`min_samples_split`
The minimum number of samples required to split an internal node. If an integer, then consider `min_samples_split` as the minimum number. If a float, then `min_samples_split` is a percentage and `ceiling(min_samples_split * n_samples)` are the minimum number of samples for each split.

`min_samples_leaf`
The minimum number of samples required to be in a leaf node. If an integer, then consider `min_samples_leaf` as the minimum number. If a float, the `min_samples_leaf` is a percentage and `ceiling(min_samples_leaf * n_samples)` are the minimum number of samples for each leaf node.

`min_weight_fraction_leaf`
The minimum weighted fraction of the sum total of weights required to be at a leaf node.

`max_features` The number of features to consider when looking for the best split.

integer consider `max_features` at each split.

float then `max_features` is a percentage and `max_features * n_features` are considered at each split.

auto then `max_features = sqrt(n_features)`

sqrt then `max_features = sqrt(n_features)`

log2 then `max_features = log2(n_features)`

NULL then `max_features = n_features`

`max_leaf_nodes` Grow trees with `max_leaf_nodes` in best-first fashion. Best nodes are defined as relative reduction to impurity. If `max_leaf_nodes = NULL` then unlimited number of leaf nodes.

`min_impurity_split`
Threshold for early stopping in tree growth. A node will split if its impurity is above the threshold, otherwise it is a leaf.

`bootstrap` Whether bootstrap samples are used when building trees.

`random_state` The seed of the random number generator.

`class_weight` A `list` with `class_label = value` pairs, or `balanced`. When `class_weight = "balanced"`, the class weights will be inversely proportional to the class frequencies in the input data as:

$$\frac{n_samples}{n_classes * table(y)}$$

Note, the class weights are multiplied with `sample_weight` (passed via `fit_params`) if `sample_weight` is specified.

`fit_params` Optional, a mapping from parameter names in the model's `fit` method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column')`.

`cross_validation_parameters`
Optional, parameter grid for learner parameters, e.g. `list(n_estimators = c(100, 200, 500),learning_rate = c(0.01, 0.1), max_depth = c(2, 3))` or `"hyperband"` for supported models.

`calibration` Optional, if not `NULL`, calibrate output probabilities with the selected method, `sigmoid`, or `isotonic`. Valid only with classification models.

`oos_scores_table`
Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

`oos_scores_db` Optional, the name of the database where the `oos_scores_table` will be created. If not provided, this will default to `database_name`.

`oos_scores_if_exists`
Optional, action to take if `oos_scores_table` already exists. One of `"fail"`, `"append"`, `"drop"`, or `"truncate"`. The default is `"fail"`.

`model_name` Optional, the prefix of the Platform modeling jobs. It will have `" Train"` or `" Predict"` added to become the Script title.

`cpu_requested` Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

`memory_requested`
Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

`disk_requested` Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

`notifications` Optional, model status notifications. See `scripts_post_custom` for further documentation about email and URL notification.

`polling_interval`
Check for job completion every this number of seconds.

`verbose` Optional, If `TRUE`, supply debug outputs in Platform logs and make prediction child jobs visible.

`civisml_version`
Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value

A `civis_ml` object, a list containing the following elements:

`job` job metadata from `scripts_get_custom`.

`run` run metadata from `scripts_get_custom_runs`.

`outputs` CivisML metadata from `scripts_list_custom_runs_outputs` containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

`metrics` Parsed CivisML output from `metrics.json` containing metadata from validation. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model with CV results.
- metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
- warnings list.
- data_platform list, training data location.

`model_info` Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model.
- metrics empty list.
- warnings list.
- data_platform list, training data location.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

`data.frame` `civis_ml(x = df, ...)`

local csv file `civis_ml(x = "path/to/data.csv", ...)`

file in Civis Platform `civis_ml(x = civis_file(1234))`

table in Civis Platform `civis_ml(x = civis_table(table_name = "schema.table", database_name = "database"))`

Examples

```
## Not run:
 df <- iris
 names(df) <- gsub("\\.", "_", names(df))

 m <- civis_ml_random_forest_classifier(df,
   dependent_variable = "Species",
   n_estimators = 100,
   max_depth = 5,
   max_features = NULL)
 yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(
   n_estimators = c(100, 200, 500),
   max_depth = c(2, 3))
```

```
 m <- civis_ml_random_forest_classifier(df,
   dependent_variable = "Species",
   max_features = NULL,
   cross_validation_parameters = cv_params)

pred_info <- predict(m,  civis_table("schema.table", "my_database"),
   output_table = "schema.scores_table")

## End(Not run)
```

`civis_ml_random_forest_regressor` *CivisML Random Forest Regressor*

Description

CivisML Random Forest Regressor

Usage

```
civis_ml_random_forest_regressor(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  n_estimators = 500,
  criterion = c("mse", "mae"),
  max_depth = NULL,
  min_samples_split = 2,
  min_samples_leaf = 1,
  min_weight_fraction_leaf = 0,
  max_features = "sqrt",
  max_leaf_nodes = NULL,
  min_impurity_split = 1e-07,
  bootstrap = TRUE,
  random_state = 42,
  fit_params = NULL,
  cross_validation_parameters = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
```

```
  verbose = FALSE,
  civisml_version = "prod"
)
```

Arguments

x	See the Data Sources section below.
dependent_variable	The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.
primary_key	Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the primary_key of the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no primary_key.
excluded_columns	Optional, a vector of columns which will be considered ineligible to be independent variables.
n_estimators	The number of boosting stages to perform. Gradient boosting is fairly robust to over-fitting, so a large number usually results in better predictive performance.
criterion	The function used to measure the quality of a split. Supported criteria are `mse` for the mean squared error, and `mae` for the mean absolute error.
max_depth	Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance. The best value depends on the interaction of the input variables.
min_samples_split	The minimum number of samples required to split an internal node. If an integer, then consider `min_samples_split` as the minimum number. If a float, then `min_samples_split` is a percentage and `ceiling(min_samples_split * n_samples)` are the minimum number of samples for each split.
min_samples_leaf	The minimum number of samples required to be in a leaf node. If an integer, then consider `min_samples_leaf` as the minimum number. If a float, the `min_samples_leaf` is a percentage and `ceiling(min_samples_leaf * n_samples)` are the minimum number of samples for each leaf node.
min_weight_fraction_leaf	The minimum weighted fraction of the sum total of weights required to be at a leaf node.
max_features	The number of features to consider when looking for the best split. **integer** consider `max_features` at each split. **float** then `max_features` is a percentage and `max_features * n_features` are considered at each split. **auto** then `max_features = sqrt(n_features)` **sqrt** then `max_features = sqrt(n_features)` **log2** then `max_features = log2(n_features)`

NULL then `max_features = n_features`

`max_leaf_nodes` Grow trees with `max_leaf_nodes` in best-first fashion. Best nodes are defined as relative reduction to impurity. If `max_leaf_nodes = NULL` then unlimited number of leaf nodes.

`min_impurity_split`
Threshold for early stopping in tree growth. A node will split if its impurity is above the threshold, otherwise it is a leaf.

`bootstrap` Whether bootstrap samples are used when building trees.

`random_state` The seed of the random number generator.

`fit_params` Optional, a mapping from parameter names in the model's `fit` method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column')`.

`cross_validation_parameters`
Optional, parameter grid for learner parameters, e.g. `list(n_estimators = c(100, 200, 500),learning_rate = c(0.01, 0.1), max_depth = c(2, 3))` or `"hyperband"` for supported models.

`oos_scores_table`
Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

`oos_scores_db` Optional, the name of the database where the `oos_scores_table` will be created. If not provided, this will default to `database_name`.

`oos_scores_if_exists`
Optional, action to take if `oos_scores_table` already exists. One of `"fail"`, `"append"`, `"drop"`, or `"truncate"`. The default is `"fail"`.

`model_name` Optional, the prefix of the Platform modeling jobs. It will have `" Train"` or `" Predict"` added to become the Script title.

`cpu_requested` Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

`memory_requested`
Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

`disk_requested` Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

`notifications` Optional, model status notifications. See `scripts_post_custom` for further documentation about email and URL notification.

`polling_interval`
Check for job completion every this number of seconds.

`verbose` Optional, If `TRUE`, supply debug outputs in Platform logs and make prediction child jobs visible.

`civisml_version`
Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value

A `civis_ml` object, a list containing the following elements:

`job` job metadata from `scripts_get_custom`.

`run` run metadata from `scripts_get_custom_runs`.

`outputs` CivisML metadata from `scripts_list_custom_runs_outputs` containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

`metrics` Parsed CivisML output from `metrics.json` containing metadata from validation. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model with CV results.
- metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
- warnings list.
- data_platform list, training data location.

`model_info` Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model.
- metrics empty list.
- warnings list.
- data_platform list, training data location.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

`data.frame` `civis_ml(x = df, ...)`

local csv file `civis_ml(x = "path/to/data.csv", ...)`

file in Civis Platform `civis_ml(x = civis_file(1234))`

table in Civis Platform `civis_ml(x = civis_table(table_name = "schema.table", database_name = "database"))`

Examples

```
## Not run:
data(ChickWeight)

m <- civis_ml_random_forest_regressor(ChickWeight,
  dependent_variable = "weight",
```

```
  n_estimators = 100,
  max_depth = 5,
  max_features = NULL)
yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(
  n_estimators = c(100, 200, 500),
  max_depth = c(2, 3))

m <- civis_ml_random_forest_regressor(ChickWeight,
  dependent_variable = "weight",
  max_features = NULL,
  cross_validation_parameters = cv_params)

pred_info <- predict(m,  civis_table("schema.table", "my_database"),
   output_table = "schema.scores_table")

## End(Not run)
```

`CIVIS_ML_REGRESSORS` *List of civis_ml regression models.*

Description

List of civis_ml regression models.

Usage

```
CIVIS_ML_REGRESSORS
```

Format

An object of class `character` of length 7.

`civis_ml_sparse_linear_regressor` *CivisML Sparse Linear Regression*

Description

CivisML Sparse Linear Regression

Usage

```
civis_ml_sparse_linear_regressor(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  fit_intercept = TRUE,
  normalize = FALSE,
  fit_params = NULL,
  cross_validation_parameters = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
  verbose = FALSE,
  civisml_version = "prod"
)
```

Arguments

x	See the Data Sources section below.
dependent_variable	The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.
primary_key	Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the primary_key of the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no primary_key.
excluded_columns	Optional, a vector of columns which will be considered ineligible to be independent variables.
fit_intercept	Should an intercept term be included in the model. If `FALSE`, no intercept will be included, in this case the data are expected to already be centered.
normalize	If `TRUE`, the regressors will be normalized before fitting the model. `normalize` is ignored when `fit_intercept = FALSE`.
fit_params	Optional, a mapping from parameter names in the model's `fit` method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column')`.
cross_validation_parameters	Optional, parameter grid for learner parameters, e.g. `list(n_estimators = c(100, 200, 500),learning_rate = c(0.01, 0.1), max_depth = c(2, 3))` or `"hyperband"` for supported models.

`oos_scores_table`
: Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

`oos_scores_db`
: Optional, the name of the database where the `oos_scores_table` will be created. If not provided, this will default to `database_name`.

`oos_scores_if_exists`
: Optional, action to take if `oos_scores_table` already exists. One of `"fail"`, `"append"`, `"drop"`, or `"truncate"`. The default is `"fail"`.

`model_name`
: Optional, the prefix of the Platform modeling jobs. It will have `" Train"` or `" Predict"` added to become the Script title.

`cpu_requested`
: Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

`memory_requested`
: Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

`disk_requested`
: Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

`notifications`
: Optional, model status notifications. See `scripts_post_custom` for further documentation about email and URL notification.

`polling_interval`
: Check for job completion every this number of seconds.

`verbose`
: Optional, If `TRUE`, supply debug outputs in Platform logs and make prediction child jobs visible.

`civisml_version`
: Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value

A `civis_ml` object, a list containing the following elements:

`job`
: job metadata from `scripts_get_custom`.

`run`
: run metadata from `scripts_get_custom_runs`.

`outputs`
: CivisML metadata from `scripts_list_custom_runs_outputs` containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

`metrics`
: Parsed CivisML output from `metrics.json` containing metadata from validation. A list containing the following elements:

  - run list, metadata about the run.
  - data list, metadata about the training data.
  - model list, the fitted scikit-learn model with CV results.
  - metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
  - warnings list.
  - data_platform list, training data location.

`model_info` Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model.
- metrics empty list.
- warnings list.
- data_platform list, training data location.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

`data.frame` `civis_ml(x = df, ...)`

local csv file `civis_ml(x = "path/to/data.csv", ...)`

file in Civis Platform `civis_ml(x = civis_file(1234))`

table in Civis Platform `civis_ml(x = civis_table(table_name = "schema.table", database_name = "database"))`

Examples

```
## Not run:
 data(ChickWeight)
 m <- civis_ml_sparse_linear_regressor(ChickWeight, dependent_variable = "weight")
 yhat <- fetch_oos_scores(m)

# make a prediction job, storing in a redshift table
pred_info <- predict(m, newdata = civis_table("schema.table", "my_database"),
   output_table = "schema.scores_table")


## End(Not run)
```

---

civis_ml_sparse_logistic

CivisML Sparse Logistic

Description

CivisML Sparse Logistic

Usage

```
civis_ml_sparse_logistic(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  penalty = c("l2", "l1"),
  dual = FALSE,
  tol = 1e-08,
  C = 499999950,
  fit_intercept = TRUE,
  intercept_scaling = 1,
  class_weight = NULL,
  random_state = 42,
  solver = c("liblinear", "newton-cg", "lbfgs", "sag"),
  max_iter = 100,
  multi_class = c("ovr", "multinomial"),
  fit_params = NULL,
  cross_validation_parameters = NULL,
  calibration = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
  verbose = FALSE,
  civisml_version = "prod"
)
```

Arguments

x	See the Data Sources section below.
dependent_variable	The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.
primary_key	Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the primary_key of the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no primary_key.
excluded_columns	Optional, a vector of columns which will be considered ineligible to be independent variables.

penalty — Used to specify the norm used in the penalization. The newton-cg, sag, and lbfgs solvers support only l2 penalties.

dual — Dual or primal formulation. Dual formulation is only implemented for l2 penalty with the liblinear solver. dual = FALSE should be preferred when n_samples > n_features.

tol — Tolerance for stopping criteria.

C — Inverse of regularization strength, must be a positive float. Smaller values specify stronger regularization.

fit_intercept — Should a constant or intercept term be included in the model.

intercept_scaling — Useful only when the solver = "liblinear" and fit_intercept = TRUE. In this case, a constant term with the value intercept_scaling is added to the design matrix.

class_weight — A list with class_label = value pairs, or balanced. When class_weight = "balanced", the class weights will be inversely proportional to the class frequencies in the input data as:

$$\frac{n_samples}{n_classes * table(y)}$$

Note, the class weights are multiplied with sample_weight (passed via fit_params) if sample_weight is specified.

random_state — The seed of the random number generator to use when shuffling the data. Used only in solver = "sag" and solver = "liblinear".

solver — Algorithm to use in the optimization problem. For small data liblinear is a good choice. sag is faster for larger problems. For multiclass problems, only newton-cg, sag, and lbfgs handle multinomial loss. liblinear is limited to one-versus-rest schemes. newton-cg, lbfgs, and sag only handle the l2 penalty.

Note that sag fast convergence is only guaranteed on features with approximately the same scale.

max_iter — The maximum number of iterations taken for the solvers to converge. Useful for the newton-cg, sag, and lbfgs solvers.

multi_class — The scheme for multi-class problems. When ovr, then a binary problem is fit for each label. When multinomial, a single model is fit minimizing the multinomial loss. Note, multinomial only works with the newton-cg, sag, and lbfgs solvers.

fit_params — Optional, a mapping from parameter names in the model's fit method to the column names which hold the data, e.g. list(sample_weight = 'survey_weight_column').

cross_validation_parameters — Optional, parameter grid for learner parameters, e.g. list(n_estimators = c(100, 200, 500),learning_rate = c(0.01, 0.1), max_depth = c(2, 3)) or "hyperband" for supported models.

calibration — Optional, if not NULL, calibrate output probabilities with the selected method, sigmoid, or isotonic. Valid only with classification models.

oos_scores_table
: Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

oos_scores_db
: Optional, the name of the database where the oos_scores_table will be created. If not provided, this will default to database_name.

oos_scores_if_exists
: Optional, action to take if oos_scores_table already exists. One of "fail", "append", "drop", or "truncate". The default is "fail".

model_name
: Optional, the prefix of the Platform modeling jobs. It will have " Train" or " Predict" added to become the Script title.

cpu_requested
: Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

memory_requested
: Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

disk_requested
: Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

notifications
: Optional, model status notifications. See scripts_post_custom for further documentation about email and URL notification.

polling_interval
: Check for job completion every this number of seconds.

verbose
: Optional, If TRUE, supply debug outputs in Platform logs and make prediction child jobs visible.

civisml_version
: Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value

A civis_ml object, a list containing the following elements:

job
: job metadata from scripts_get_custom.

run
: run metadata from scripts_get_custom_runs.

outputs
: CivisML metadata from scripts_list_custom_runs_outputs containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

metrics
: Parsed CivisML output from metrics.json containing metadata from validation. A list containing the following elements:

  - run list, metadata about the run.
  - data list, metadata about the training data.
  - model list, the fitted scikit-learn model with CV results.
  - metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
  - warnings list.
  - data_platform list, training data location.

`model_info` Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model.
- metrics empty list.
- warnings list.
- data_platform list, training data location.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

`data.frame` `civis_ml(x = df, ...)`

local csv file `civis_ml(x = "path/to/data.csv", ...)`

file in Civis Platform `civis_ml(x = civis_file(1234))`

table in Civis Platform `civis_ml(x = civis_table(table_name = "schema.table", database_name = "database"))`

Examples

```
## Not run:

df <- iris
names(df) <- gsub("\\.", "_", names(df))

m <- civis_ml_sparse_logistic(df, "Species")
yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(C = c(.01, 1, 10, 100, 1000))

m <- civis_ml_sparse_logistic(df, "Species",
  cross_validation_parameters = cv_params)

# make a prediction job, storing in a redshift table
pred_info <- predict(m, newdata = civis_table("schema.table", "my_database"),
   output_table = "schema.scores_table")


## End(Not run)
```

civis_ml_sparse_ridge_regressor *CivisML Sparse Ridge Regression*

Description

CivisML Sparse Ridge Regression

Usage

```
civis_ml_sparse_ridge_regressor(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  alpha = 1,
  fit_intercept = TRUE,
  normalize = FALSE,
  max_iter = NULL,
  tol = 0.001,
  solver = c("auto", "svd", "cholesky", "lsqr", "sparse_cg", "sag"),
  random_state = 42,
  fit_params = NULL,
  cross_validation_parameters = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
  verbose = FALSE,
  civisml_version = "prod"
)
```

Arguments

x	See the Data Sources section below.
dependent_variable	The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.
primary_key	Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the primary_key of

the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no primary_key.

`excluded_columns`
: Optional, a vector of columns which will be considered ineligible to be independent variables.

`alpha`
: The regularization strength, must be a vector of floats of length n_targets or a single float. Larger values specify stronger regularization.

`fit_intercept`
: Should an intercept term be included in the model. If `FALSE`, no intercept will be included, in this case the data are expected to already be centered.

`normalize`
: If `TRUE`, the regressors will be normalized before fitting the model. `normalize` is ignored when `fit_intercept = FALSE`.

`max_iter`
: Maximum number of iterations for conjugate gradient solver. For `sparse_cg` and `lsqr` solvers, the default value is predetermined. For the `sag` solver, the default value is 1000.

`tol`
: Precision of the solution.

`solver`
: Solver to use for the optimization problem.

  auto chooses the solver automatically based on the type of data.

  svd uses Singular Value Decomposition of X to compute the Ridge coefficients. More stable for singular matrices than `cholesky`.

  cholesky uses the standard decomposition to obtain a closed-form solution.

  sparse_cg uses the conjugate gradient solver. As an iterative algorithm, this solver is more appropriate than `cholesky` for large-scale data.

  lsqr uses the dedicated regularized least-squares routine.

  sag uses Stochastic Average Gradient descent. It also uses an iterative procedure, and is often faster than other solvers when both n_samples and n_features are large. Note that `sag` fast convergence is only guaranteed on features with approximately the same scale

`random_state`
: The seed of the pseudo random number generator to use when shuffling the data. Used only when `solver = "sag"`.

`fit_params`
: Optional, a mapping from parameter names in the model's `fit` method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column')`.

`cross_validation_parameters`
: Optional, parameter grid for learner parameters, e.g. `list(n_estimators = c(100, 200, 500),learning_rate = c(0.01, 0.1), max_depth = c(2, 3))` or `"hyperband"` for supported models.

`oos_scores_table`
: Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

`oos_scores_db`
: Optional, the name of the database where the `oos_scores_table` will be created. If not provided, this will default to `database_name`.

`oos_scores_if_exists`
: Optional, action to take if `oos_scores_table` already exists. One of `"fail"`, `"append"`, `"drop"`, or `"truncate"`. The default is `"fail"`.

`model_name` Optional, the prefix of the Platform modeling jobs. It will have `"Train"` or `"Predict"` added to become the Script title.

`cpu_requested` Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

`memory_requested` Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

`disk_requested` Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

`notifications` Optional, model status notifications. See `scripts_post_custom` for further documentation about email and URL notification.

`polling_interval` Check for job completion every this number of seconds.

`verbose` Optional, If `TRUE`, supply debug outputs in Platform logs and make prediction child jobs visible.

`civisml_version` Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value

A `civis_ml` object, a list containing the following elements:

`job` job metadata from `scripts_get_custom`.

`run` run metadata from `scripts_get_custom_runs`.

`outputs` CivisML metadata from `scripts_list_custom_runs_outputs` containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

`metrics` Parsed CivisML output from `metrics.json` containing metadata from validation. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model with CV results.
- metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
- warnings list.
- data_platform list, training data location.

`model_info` Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model.
- metrics empty list.
- warnings list.
- data_platform list, training data location.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

`data.frame` `civis_ml(x = df, ...)`

local csv file `civis_ml(x = "path/to/data.csv", ...)`

file in Civis Platform `civis_ml(x = civis_file(1234))`

table in Civis Platform `civis_ml(x = civis_table(table_name = "schema.table", database_name = "database"))`

Examples

```
## Not run:
 data(ChickWeight)
 m <- civis_ml_sparse_ridge_regressor(ChickWeight, dependent_variable = "weight", alpha = 999)
 yhat <- fetch_oos_scores(m)

 # Grid search
 cv_params <- list(alpha = c(.001, .01, .1, 1))
 m <- civis_ml_sparse_ridge_regressor(ChickWeight,
   dependent_variable = "weight",
   cross_validation_parameters = cv_params,
   calibration = NULL)

# make a prediction job, storing in a redshift table
pred_info <- predict(m, newdata = civis_table("schema.table", "my_database"),
   output_table = "schema.scores_table")

## End(Not run)
```

`civis_platform` *Evaluate an expression in Civis Platform*

Description

This is used as with the `future` API as an argument to `plan`.

Usage

```
civis_platform(...)
```

Arguments

`...`	Arguments to `CivisFuture` and then `scripts_post_containers`

Value

The result of evaluating `expr`.

Examples

```
## Not run:

 plan(civis_platform)

 # Specify required resources, image, and tag.
 fut <- future({2 + 2},
   required_resources = list(cpu = 1024, memory = 2048),
   docker_image_name = "civisanalytics/datascience-r",
   docker_image_tag = "3")

 # Evaluate the future later
 fut <- future({2 + 2}, lazy = TRUE)
 run(fut)

 # check if a future has resolved
 resolved(fut)

 # block until the future resolves, and return the value or throw error
 value(fut)

 # cancel the job
 cancel(fut)

 # grab the run logs
 fetch_logs(fut)

 # handle errors
 fut <- future({stop("Error!")})
 e <- tryCatch(value(fut), error = function(e) e)
 get_error(e)

## End(Not run)
```

`civis_script` *Civis Script*

Description

Create a civis_script object from a job and run id.

Usage

```
civis_script(id, run_id = NULL)
```

Arguments

id	integer, job id of the script.
run_id	integer, id of the run. If NULL (default), the most recent run will be used.

Details

A civis_script can be any custom, container, R, Python, SQL, or Javacript platform script.

Value

a civis_script object.

See Also

read_civis

Other script_utils: fetch_output_file_ids(), run_civis(), run_template()

---

civis_table *A table in the Civis Platform*

Description

Use civis_table to use a table in the Civis Platform with civis_ml.

Usage

```
civis_table(
  table_name,
  database_name = NULL,
  sql_where = NULL,
  sql_limit = NULL
)
```

Arguments

table_name	The table name, as "schema.table".
database_name	The name of the database holding table_name.
sql_where	A SQL WHERE clause used to scope the rows of the training or prediction table. Note, the WHERE keyword is added at runtime.
sql_limit	A SQL_LIMIT clause for querying the training or prediction set. Note, the LIMIT keyword is added at runtime.

Value

A civis_table.

`civis_to_multifile_csv` *Split a Redshift table into multiple CSV parts on S3*

Description

Split a Redshift table into multiple CSV parts on S3 and return their locations as file IDs and presigned S3 urls.

Usage

```
civis_to_multifile_csv(
  sql,
  database,
  job_name = NULL,
  hidden = TRUE,
  include_header = TRUE,
  compression = "gzip",
  delimiter = ",",
  unquoted = FALSE,
  prefix = NULL
)
```

Arguments

`sql`	string, The SQL select string to be executed.
`database`	string, Name of database where query is run.
`job_name`	string, optional. Name to identify scripted sql job.
`hidden`	logical, Whether to hide the query in platform.
`include_header`	logical, optional. Whether to include headers as an element in the returned list.
`compression`	string, optional, Type of compression to use, if any. One of 'none', 'zip', or 'gzip'.
`delimiter`	string, optional. Which delimiter to use. One of ',', '\t' or '|'.
`unquoted`	logical, optional. Whether or not to quote fields.
`prefix`	string, optional. A user specified filename prefix for the output files to have.

Details

When tables are large, unloading by splitting them first is faster. When we split a table, each Redshift compute node can dump its data into S3 in parallel with the others. By doing so, we avoid having all compute nodes sending the data through the leader node, which is slow.

This function returns a list that contains the location of the CSV parts as file IDs and presigned S3 urls. The user can use either the file IDs or the presigned S3 urls to download the CSV parts. The content of the list returned by this function is similar to that of the manifest file returned by Amazon S3 UNLOAD statements.

Value

A list with the items:

- header: column headers if 'include_header' is TRUE
- query: the executed query
- entries: a list containing presigned urls for each csv part
- compression: the type of compression on each csv part
- delimiter: the delimiter used to separate fields
- unquoted: whether fields are quoted

Examples

```
## Not run:
# Download a table into multiple csv parts
sql <- "SELECT * FROM schema.table"
database <- "important_database"
manifest <- civis_to_multifile_csv(sql=sql, database=database)
files <- lapply(manifest[["entries"]], function(x) {
  download_civis(x$id, x$name)
  x$name
})

## End(Not run)
```

---

clusters_delete_kubernetes_partitions

Delete a Cluster Partition

Description

Delete a Cluster Partition

Usage

```
clusters_delete_kubernetes_partitions(id, cluster_partition_id)
```

Arguments

id	integer required. The ID of the cluster which this partition belongs to.
cluster_partition_id	integer required. The ID of this cluster partition.

Value

An empty HTTP response

clusters_get_kubernetes *Describe a Kubernetes Cluster*

Description

Describe a Kubernetes Cluster

Usage

```
clusters_get_kubernetes(id, include_usage_stats = NULL)
```

Arguments

id	integer required.
include_usage_stats	boolean optional. When true, usage stats are returned in instance config objects. Defaults to false.

Value

A list containing the following elements:

id	integer, The ID of this cluster.
organizationId	string, The id of this cluster's organization.
organizationName	string, The name of this cluster's organization.
organizationSlug	string, The slug of this cluster's organization.
rawClusterSlug	string, The slug of this cluster's raw configuration.
customPartitions	boolean, Whether this cluster has a custom partition configuration.
clusterPartitions	array, An array containing the following fields: • clusterPartitionId integer, The ID of this cluster partition. • name string, The name of the cluster partition. • labels array, Labels associated with this partition. • instanceConfigs array, The instances configured for this cluster partition. • defaultInstanceConfigId integer, The id of the InstanceConfig that is the default for this partition.
isNatEnabled	boolean, Whether this cluster needs a NAT gateway or not.
hours	number, The number of hours used this month for this cluster.

clusters_get_kubernetes_instance_configs
Describe an Instance Config

Description

Describe an Instance Config

Usage

```
clusters_get_kubernetes_instance_configs(
  instance_config_id,
  include_usage_stats = NULL
)
```

Arguments

instance_config_id	integer required. The ID of this instance config.
include_usage_stats	boolean optional. When true, usage stats are returned in instance config objects. Defaults to false.

Value

A list containing the following elements:

instanceConfigId	integer, The ID of this InstanceConfig.
instanceType	string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, c5.18xlarge, and g6.2xlarge.
minInstances	integer, The minimum number of instances of that type in this cluster.
maxInstances	integer, The maximum number of instances of that type in this cluster.
instanceMaxMemory	integer, The amount of memory (RAM) available to a single instance of that type in megabytes.
instanceMaxCpu	integer, The number of processor shares available to a single instance of that type in millicores.
instanceMaxDisk	integer, The amount of disk available to a single instance of that type in gigabytes.
usageStats	list, A list containing the following elements:

- pendingMemoryRequested integer, The sum of memory requests (in MB) for pending deployments in this instance config.

- pendingCpuRequested integer, The sum of cpu requests (in millicores) for pending deployments in this instance config.
- runningMemoryRequested integer, The sum of memory requests (in MB) for running deployments in this instance config.
- runningCpuRequested integer, The sum of cpu requests (in millicores) for running deployments in this instance config.
- pendingDeployments integer, The number of pending deployments in this instance config.
- runningDeployments integer, The number of running deployments in this instance config.

`clusterPartitionId`
: integer, The ID of this InstanceConfig's cluster partition

`clusterPartitionName`
: string, The name of this InstanceConfig's cluster partition

---

`clusters_get_kubernetes_partitions` *Describe a Cluster Partition*

Description

Describe a Cluster Partition

Usage

```
clusters_get_kubernetes_partitions(
  id,
  cluster_partition_id,
  include_usage_stats = NULL
)
```

Arguments

`id`
: integer required. The ID of the cluster which this partition belongs to.

`cluster_partition_id`
: integer required. The ID of this cluster partition.

`include_usage_stats`
: boolean optional. When true, usage stats are returned in instance config objects. Defaults to false.

Value

A list containing the following elements:

`clusterPartitionId`
: integer, The ID of this cluster partition.

`name` string, The name of the cluster partition.

`labels` array, Labels associated with this partition.

`instanceConfigs`
array, An array containing the following fields:

- instanceConfigId integer, The ID of this InstanceConfig.
- instanceType string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, c5.18xlarge, and g6.2xlarge.
- minInstances integer, The minimum number of instances of that type in this cluster.
- maxInstances integer, The maximum number of instances of that type in this cluster.
- instanceMaxMemory integer, The amount of memory (RAM) available to a single instance of that type in megabytes.
- instanceMaxCpu integer, The number of processor shares available to a single instance of that type in millicores.
- instanceMaxDisk integer, The amount of disk available to a single instance of that type in gigabytes.
- usageStats object,

`defaultInstanceConfigId`
integer, The id of the InstanceConfig that is the default for this partition.

`clusters_list_kubernetes` *List Kubernetes Clusters*

Description

List Kubernetes Clusters

Usage

```
clusters_list_kubernetes(
  organization_id = NULL,
  organization_slug = NULL,
  raw_cluster_slug = NULL,
  exclude_inactive_orgs = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

`organization_id`
: integer optional. The ID of this cluster's organization. Cannot be used along with the organization slug.

`organization_slug`
: string optional. The slug of this cluster's organization. Cannot be used along with the organization ID.

`raw_cluster_slug`
: string optional. The slug of this cluster's raw configuration.

`exclude_inactive_orgs`
: boolean optional. When true, excludes KubeClusters associated with inactive orgs. Defaults to false.

`limit`
: integer optional. Number of results to return. Defaults to its maximum of 50.

`page_num`
: integer optional. Page number of the results to return. Defaults to the first page, 1.

`order`
: string optional. The field on which to order the result set. Defaults to organization_id. Must be one of: organization_id, created_at.

`order_dir`
: string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

`id`
: integer, The ID of this cluster.

`organizationId`
: string, The id of this cluster's organization.

`organizationName`
: string, The name of this cluster's organization.

`organizationSlug`
: string, The slug of this cluster's organization.

`rawClusterSlug`
: string, The slug of this cluster's raw configuration.

`customPartitions`
: boolean, Whether this cluster has a custom partition configuration.

`clusterPartitions`
: array, An array containing the following fields:

  - clusterPartitionId integer, The ID of this cluster partition.
  - name string, The name of the cluster partition.
  - labels array, Labels associated with this partition.
  - instanceConfigs array, The instances configured for this cluster partition.
  - defaultInstanceConfigId integer, The id of the InstanceConfig that is the default for this partition.

`isNatEnabled`
: boolean, Whether this cluster needs a NAT gateway or not.

`clusters_list_kubernetes_compute_hours`
List compute hours for a Kubernetes Cluster

Description

List compute hours for a Kubernetes Cluster

Usage

```
clusters_list_kubernetes_compute_hours(id, include_usage_stats = NULL)
```

Arguments

`id`	integer required.
`include_usage_stats`	boolean optional. When true, usage stats are returned in instance config objects. Defaults to false.

Value

An array containing the following fields:

`totalNormalizedHours`	integer, The total number of normalized hours used by this cluster.
`normalizedHoursByInstanceType`	string, Denotes the instance type the normalized hours are attributed to.
`updatedAt`	string,
`monthAndYear`	string, The month and year the normalized hours are attributed to.

`clusters_list_kubernetes_compute_metrics`
Get historical compute metrics for a Kubernetes Cluster

Description

Get historical compute metrics for a Kubernetes Cluster

Usage

```
clusters_list_kubernetes_compute_metrics(
  id,
  start_date = NULL,
  end_date = NULL,
  type = NULL
)
```

Arguments

id	integer required. The ID of the Kubernetes cluster.
start_date	string optional. UTC start date in YYYY-MM-DD format (inclusive).
end_date	string optional. UTC end date in YYYY-MM-DD format (inclusive). Must be within 31 days of start_date.
type	array optional. Compute metric types to retrieve. Allowed values are: ["Jobs"]. Defaults to returning metrics for all available types. The default may change as support for additional types is added.

Value

A list containing the following elements:

url	string, Presigned URL to download the compute metrics CSV file from S3.

---

clusters_list_kubernetes_deployments

List the deployments associated with a Kubernetes Cluster

Description

List the deployments associated with a Kubernetes Cluster

Usage

```
clusters_list_kubernetes_deployments(
  id,
  base_type = NULL,
  state = NULL,
  start_date = NULL,
  end_date = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The id of the cluster.
base_type	string optional. If specified, return deployments of these base types. It accepts a comma-separated list, possible values are 'Notebook', 'Service', 'Run'.
state	string optional. If specified, return deployments in these states. It accepts a comma-separated list, possible values are pending, running, terminated, sleeping

start_date	string optional. If specified, return deployments created after this date. Must be 31 days or less before end date. Defaults to 7 days prior to end date. The date must be provided in the format YYYY-MM-DD.
end_date	string optional. If specified, return deployments created before this date. Defaults to 7 days after start date, or today if start date is not specified. The date must be provided in the format YYYY-MM-DD.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id	integer, The id of this deployment.
name	string, The name of the deployment.
baseId	integer, The id of the base object associated with the deployment.
baseType	string, The base type of this deployment.
state	string, The state of the deployment.
cpu	integer, The CPU in millicores required by the deployment.
memory	integer, The memory in MB required by the deployment.
diskSpace	integer, The disk space in GB required by the deployment.
instanceType	string, The EC2 instance type requested for the deployment.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
maxMemoryUsage	number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
maxCpuUsage	number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
createdAt	string,
updatedAt	string,

clusters_list_kubernetes_deployment_stats

Get stats about deployments associated with a Kubernetes Cluster

Description

Get stats about deployments associated with a Kubernetes Cluster

Usage

```
clusters_list_kubernetes_deployment_stats(id)
```

Arguments

id	integer required. The ID of this cluster.

Value

An array containing the following fields:

baseType	string, The base type of this deployment
state	string, State of the deployment
count	integer, Number of deployments of base type and state
totalCpu	integer, Total amount of CPU in millicores for deployments of base type and state
totalMemory	integer, Total amount of Memory in megabytes for deployments of base type and state

clusters_list_kubernetes_instance_configs_active_workloads

List active workloads in an Instance Config

Description

List active workloads in an Instance Config

Usage

```
clusters_list_kubernetes_instance_configs_active_workloads(id, state = NULL)
```

Arguments

id	integer required. The id of the instance config.
state	string optional. If specified, return workloads in these states. It accepts a comma-separated list, possible values are pending, running

Value

An array containing the following fields:

`id`	integer, The id of this deployment.
`baseType`	string, The base type of this deployment.
`baseId`	integer, The id of the base object associated with this deployment.
`baseObjectName`	string, The name of the base object associated with this deployment. Null if you do not have permission to read the object.
`jobType`	string, If the base object is a job run you have permission to read, the type of the job. One of "python_script", "r_script", "container_script", or "custom_script".
`jobId`	integer, If the base object is a job run you have permission to read, the id of the job.
`jobCancelRequestedAt`	string, If the base object is a job run you have permission to read, and it was requested to be cancelled, the timestamp of that request.
`state`	string, The state of this deployment.
`cpu`	integer, The CPU in millicores requested by this deployment.
`memory`	integer, The memory in MB requested by this deployment.
`diskSpace`	integer, The disk space in GB requested by this deployment.
`user`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`createdAt`	string, The timestamp of when the deployment began.
`cancellable`	boolean, True if you have permission to cancel this deployment.

`clusters_list_kubernetes_instance_configs_historical_graphs` *Get graphs of historical resource usage in an Instance Config*

Description

Get graphs of historical resource usage in an Instance Config

Usage

```
clusters_list_kubernetes_instance_configs_historical_graphs(
  instance_config_id,
  timeframe = NULL
)
```

Arguments

`instance_config_id`
: integer required. The ID of this instance config.

`timeframe`
: string optional. The span of time that the graphs cover. Must be one of 1_day, 1_week.

Value

A list containing the following elements:

`cpuGraphUrl`
: string, URL for the graph of historical CPU usage in this instance config.

`memGraphUrl`
: string, URL for the graph of historical memory usage in this instance config.

`clusters_list_kubernetes_instance_configs_historical_metrics` *Get graphs of historical resource usage in an Instance Config*

Description

Get graphs of historical resource usage in an Instance Config

Usage

```
clusters_list_kubernetes_instance_configs_historical_metrics(
  instance_config_id,
  timeframe = NULL,
  metric = NULL
)
```

Arguments

`instance_config_id`
: integer required. The ID of this instance config.

`timeframe`
: string optional. The span of time that the graphs cover. Must be one of 1_day, 1_week.

`metric`
: string optional. The metric to retrieve. Must be one of cpu, memory.

Value

A list containing the following elements:

`instanceConfigId`
: integer, The ID of this instance config.

`metric`
: string, URL for the graph of historical CPU usage in this instance config.

`timeframe`
: string, The span of time that the graphs cover. Must be one of 1_day, 1_week.

`unit`
: string, The unit of the values.

`metrics` list, A list containing the following elements:

- used list . A list containing the following elements:
  - times array, The times associated with data points, in seconds since epoch.
  - values array, The values of the data points.
- requested list . A list containing the following elements:
  - times array, The times associated with data points, in seconds since epoch.
  - values array, The values of the data points.
- capacity list . A list containing the following elements:
  - times array, The times associated with data points, in seconds since epoch.
  - values array, The values of the data points.

`clusters_list_kubernetes_instance_configs_user_statistics`
Get statistics about the current users of an Instance Config

Description

Get statistics about the current users of an Instance Config

Usage

```
clusters_list_kubernetes_instance_configs_user_statistics(
  instance_config_id,
  order = NULL,
  order_dir = NULL
)
```

Arguments

`instance_config_id` integer required. The ID of this instance config.

`order` string optional. The field on which to order the result set. Defaults to running_deployments. Must be one of pending_memory_requested, pending_cpu_requested, running_memory_requested, running_cpu_requested, pending_deployments, running_deployments.

`order_dir` string optional. Direction in which to sort, either asc (ascending) or desc (descending). Defaults to desc.

Value

An array containing the following fields:

`userId` string, The owning user's ID

`userName` string, The owning user's name

`pendingDeployments`
integer, The number of deployments belonging to the owning user in "pending" state

`pendingMemoryRequested`
integer, The sum of memory requests (in MB) for deployments belonging to the owning user in "pending" state

`pendingCpuRequested`
integer, The sum of CPU requests (in millicores) for deployments belonging to the owning user in "pending" state

`runningDeployments`
integer, The number of deployments belonging to the owning user in "running" state

`runningMemoryRequested`
integer, The sum of memory requests (in MB) for deployments belonging to the owning user in "running" state

`runningCpuRequested`
integer, The sum of CPU requests (in millicores) for deployments belonging to the owning user in "running" state

---

`clusters_list_kubernetes_partitions`

List Cluster Partitions for given cluster

Description

List Cluster Partitions for given cluster

Usage

```
clusters_list_kubernetes_partitions(id, include_usage_stats = NULL)
```

Arguments

`id` integer required.

`include_usage_stats`
boolean optional. When true, usage stats are returned in instance config objects. Defaults to false.

Value

An array containing the following fields:

`clusterPartitionId`
: integer, The ID of this cluster partition.

`name`
: string, The name of the cluster partition.

`labels`
: array, Labels associated with this partition.

`instanceConfigs`
: array, An array containing the following fields:

- instanceConfigId integer, The ID of this InstanceConfig.
- instanceType string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, c5.18xlarge, and g6.2xlarge.
- minInstances integer, The minimum number of instances of that type in this cluster.
- maxInstances integer, The maximum number of instances of that type in this cluster.
- instanceMaxMemory integer, The amount of memory (RAM) available to a single instance of that type in megabytes.
- instanceMaxCpu integer, The number of processor shares available to a single instance of that type in millicores.
- instanceMaxDisk integer, The amount of disk available to a single instance of that type in gigabytes.
- usageStats object,

`defaultInstanceConfigId`
: integer, The id of the InstanceConfig that is the default for this partition.

---

clusters_patch_kubernetes_partitions

Update a Cluster Partition

Description

Update a Cluster Partition

Usage

```
clusters_patch_kubernetes_partitions(
  id,
  cluster_partition_id,
  instance_configs = NULL,
  name = NULL,
  labels = NULL
)
```

Arguments

`id` integer required. The ID of the cluster which this partition belongs to.

`cluster_partition_id`
integer required. The ID of this cluster partition.

`instance_configs`
array optional. An array containing the following fields:

- instanceType string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, c5.18xlarge, and g6.2xlarge.
- minInstances integer, The minimum number of instances of that type in this cluster.
- maxInstances integer, The maximum number of instances of that type in this cluster.

`name` string optional. The name of the cluster partition.

`labels` array optional. Labels associated with this partition.

Value

A list containing the following elements:

`clusterPartitionId`
integer, The ID of this cluster partition.

`name` string, The name of the cluster partition.

`labels` array, Labels associated with this partition.

`instanceConfigs`
array, An array containing the following fields:

- instanceConfigId integer, The ID of this InstanceConfig.
- instanceType string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, c5.18xlarge, and g6.2xlarge.
- minInstances integer, The minimum number of instances of that type in this cluster.
- maxInstances integer, The maximum number of instances of that type in this cluster.
- instanceMaxMemory integer, The amount of memory (RAM) available to a single instance of that type in megabytes.
- instanceMaxCpu integer, The number of processor shares available to a single instance of that type in millicores.
- instanceMaxDisk integer, The amount of disk available to a single instance of that type in gigabytes.
- usageStats object,

`defaultInstanceConfigId`
integer, The id of the InstanceConfig that is the default for this partition.

clusters_post_kubernetes_partitions
Create a Cluster Partition for given cluster

Description

Create a Cluster Partition for given cluster

Usage

```
clusters_post_kubernetes_partitions(id, instance_configs, name, labels)
```

Arguments

id	integer required. The ID of the cluster which this partition belongs to.
instance_configs	array required. An array containing the following fields: • instanceType string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, c5.18xlarge, and g6.2xlarge. • minInstances integer, The minimum number of instances of that type in this cluster. • maxInstances integer, The maximum number of instances of that type in this cluster.
name	string required. The name of the cluster partition.
labels	array required. Labels associated with this partition.

Value

A list containing the following elements:

clusterPartitionId	integer, The ID of this cluster partition.
name	string, The name of the cluster partition.
labels	array, Labels associated with this partition.
instanceConfigs	array, An array containing the following fields: • instanceConfigId integer, The ID of this InstanceConfig. • instanceType string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, c5.18xlarge, and g6.2xlarge. • minInstances integer, The minimum number of instances of that type in this cluster. • maxInstances integer, The maximum number of instances of that type in this cluster. • instanceMaxMemory integer, The amount of memory (RAM) available to a single instance of that type in megabytes.

- instanceMaxCpu integer, The number of processor shares available to a single instance of that type in millicores.
- instanceMaxDisk integer, The amount of disk available to a single instance of that type in gigabytes.
- usageStats object,

`defaultInstanceConfigId`
: integer, The id of the InstanceConfig that is the default for this partition.

`coef.civis_ml` *Get civis_ml model coefficients*

Description

Get civis_ml model coefficients

Usage

```
## S3 method for class 'civis_ml'
coef(object, complete = TRUE, ...)
```

Arguments

`object`	civis_ml_model
`complete`	see documentation for generic `coef()`
`...`	other arguments

Details

Outputs coefficients with names in the style of ‘stats::coef‘

Value

a matrix of coefficients, or ‘NULL‘ if none available from CivisML

credentials_delete *Delete a credential*

Description

Delete a credential

Usage

```
credentials_delete(id)
```

Arguments

id	integer required. The ID of the credential.

Value

An empty HTTP response

credentials_delete_shares_groups *Revoke the permissions a group has on this object*

Description

Revoke the permissions a group has on this object

Usage

```
credentials_delete_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

credentials_delete_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
credentials_delete_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

credentials_get

Get a credential

Description

Get a credential

Usage

```
credentials_get(id)
```

Arguments

id	integer required. The ID of the credential.

Value

A list containing the following elements:

id	integer, The ID of the credential.
name	string, The name identifying the credential
type	string, The credential's type.
username	string, The username for the credential.
description	string, A long description of the credential.

owner	string, The username of the user who this credential belongs to. Using user.username is preferred.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
remoteHostId	integer, The ID of the remote host associated with this credential.
remoteHostName	string, The name of the remote host associated with this credential.
state	string, The U.S. state for the credential. Only for VAN credentials.
createdAt	string, The creation time for this credential.
updatedAt	string, The last modification time for this credential.
default	boolean, Whether or not the credential is a default. Only for Database credentials.
oauth	boolean, Whether or not the credential is an OAuth credential.

`credentials_list` *List credentials*

Description

List credentials

Usage

```
credentials_list(
  type = NULL,
  remote_host_id = NULL,
  default = NULL,
  system_credentials = NULL,
  users = NULL,
  name = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

type	string optional. The type (or types) of credentials to return. One or more of: Amazon Web Services S3, Bitbucket, CASS/NCOA PAF, Certificate, Civis Platform, Custom, Database, Google, Github, Salesforce User, Salesforce Client, and TableauUser. Specify multiple values as a comma-separated list (e.g., "A,B").
remote_host_id	integer optional. The ID of the remote host associated with the credentials to return.
default	boolean optional. If true, will return a list with a single credential which is the current user's default credential.
system_credentials	boolean optional. If true, will only return system credentials. System credentials can only be created and viewed by Civis Admins.
users	string optional. A comma-separated list of user ids. If specified, returns set of credentials owned by the users that requesting user has at least read access on.
name	string optional. If specified, will be used to filter the credentials returned. Will search across name and will return any full name containing the search string.
limit	integer optional. Number of results to return. Defaults to its maximum of 1000.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, created_at, name.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the credential.
name	string, The name identifying the credential
type	string, The credential's type.
username	string, The username for the credential.
description	string, A long description of the credential.
owner	string, The username of the user who this credential belongs to. Using user.username is preferred.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
remoteHostId	integer, The ID of the remote host associated with this credential.
remoteHostName	string, The name of the remote host associated with this credential.

state	string, The U.S. state for the credential. Only for VAN credentials.
createdAt	string, The creation time for this credential.
updatedAt	string, The last modification time for this credential.
default	boolean, Whether or not the credential is a default. Only for Database credentials.
oauth	boolean, Whether or not the credential is an OAuth credential.

`credentials_list_dependencies`
List dependent objects for this object

Description

List dependent objects for this object

Usage

```
credentials_list_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

credentials_list_shares *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
credentials_list_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

credentials_list_types *Get list of Credential Types*

Description

Get list of Credential Types

Usage

```
credentials_list_types()
```

Value

A list containing the following elements:

types	array, list of acceptable credential types

credentials_patch *Update some attributes of a credential*

Description

Update some attributes of a credential

Usage

```
credentials_patch(
  id,
  name = NULL,
  type = NULL,
  description = NULL,
  username = NULL,
  password = NULL,
  remote_host_id = NULL,
  user_id = NULL,
  state = NULL,
  system_credential = NULL,
  default = NULL,
  oauth = NULL
)
```

Arguments

id	integer required. The ID of the credential.
name	string optional. The name identifying the credential.
type	string optional. The type of credential. Note: only these credentials can be created or edited via this API ["Amazon Web Services S3", "CASS/NCOA PAF", "Certificate", "Civis Platform", "Custom", "Database", "Google", "Salesforce User", "Salesforce Client", "TableauUser"]
description	string optional. A long description of the credential.
username	string optional. The username for the credential.
password	string optional. The password for the credential.
remote_host_id	integer optional. The ID of the remote host associated with the credential.
user_id	integer optional. The ID of the user the credential is created for. Note: This attribute is only accepted if you are a Civis Admin User.
state	string optional. The U.S. state for the credential. Only for VAN credentials.
system_credential	boolean optional. Boolean flag that sets a credential to be a system credential. System credentials can only be created by Civis Admins and will create a credential owned by the Civis Robot user.
default	boolean optional. Whether or not the credential is a default. Only for Database credentials.
oauth	boolean optional. Whether or not the credential is an OAuth credential.

Value

A list containing the following elements:

id	integer, The ID of the credential.
name	string, The name identifying the credential
type	string, The credential's type.
username	string, The username for the credential.
description	string, A long description of the credential.
owner	string, The username of the user who this credential belongs to. Using user.username is preferred.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
remoteHostId	integer, The ID of the remote host associated with this credential.
remoteHostName	string, The name of the remote host associated with this credential.

state	string, The U.S. state for the credential. Only for VAN credentials.
createdAt	string, The creation time for this credential.
updatedAt	string, The last modification time for this credential.
default	boolean, Whether or not the credential is a default. Only for Database credentials.
oauth	boolean, Whether or not the credential is an OAuth credential.

`credentials_post` *Create a credential*

Description

Create a credential

Usage

```
credentials_post(
  type,
  username,
  password,
  name = NULL,
  description = NULL,
  remote_host_id = NULL,
  user_id = NULL,
  state = NULL,
  system_credential = NULL,
  default = NULL,
  oauth = NULL
)
```

Arguments

type	string required. The type of credential. Note: only these credentials can be created or edited via this API ["Amazon Web Services S3", "CASS/NCOA PAF", "Certificate", "Civis Platform", "Custom", "Database", "Google", "Salesforce User", "Salesforce Client", "TableauUser"]
username	string required. The username for the credential.
password	string required. The password for the credential.
name	string optional. The name identifying the credential.
description	string optional. A long description of the credential.
remote_host_id	integer optional. The ID of the remote host associated with the credential.
user_id	integer optional. The ID of the user the credential is created for. Note: This attribute is only accepted if you are a Civis Admin User.
state	string optional. The U.S. state for the credential. Only for VAN credentials.

`system_credential`
: boolean optional. Boolean flag that sets a credential to be a system credential. System credentials can only be created by Civis Admins and will create a credential owned by the Civis Robot user.

`default`
: boolean optional. Whether or not the credential is a default. Only for Database credentials.

`oauth`
: boolean optional. Whether or not the credential is an OAuth credential.

Value

A list containing the following elements:

`id`
: integer, The ID of the credential.

`name`
: string, The name identifying the credential

`type`
: string, The credential's type.

`username`
: string, The username for the credential.

`description`
: string, A long description of the credential.

`owner`
: string, The username of the user who this credential belongs to. Using user.username is preferred.

`user`
: list, A list containing the following elements:

  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

`remoteHostId`
: integer, The ID of the remote host associated with this credential.

`remoteHostName`
: string, The name of the remote host associated with this credential.

`state`
: string, The U.S. state for the credential. Only for VAN credentials.

`createdAt`
: string, The creation time for this credential.

`updatedAt`
: string, The last modification time for this credential.

`default`
: boolean, Whether or not the credential is a default. Only for Database credentials.

`oauth`
: boolean, Whether or not the credential is an OAuth credential.

credentials_post_authenticate *Authenticate against a remote host*

Description

Authenticate against a remote host

Usage

```
credentials_post_authenticate(url, remote_host_type, username, password)
```

Arguments

url	string required. The URL to your host.
remote_host_type	string required. The type of remote host. One of: RemoteHostTypes::Bigquery, RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce
username	string required. The username for the credential.
password	string required. The password for the credential.

Value

A list containing the following elements:

id	integer, The ID of the credential.
name	string, The name identifying the credential
type	string, The credential's type.
username	string, The username for the credential.
description	string, A long description of the credential.
owner	string, The username of the user who this credential belongs to. Using user.username is preferred.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
remoteHostId	integer, The ID of the remote host associated with this credential.
remoteHostName	string, The name of the remote host associated with this credential.

state	string, The U.S. state for the credential. Only for VAN credentials.
createdAt	string, The creation time for this credential.
updatedAt	string, The last modification time for this credential.
default	boolean, Whether or not the credential is a default. Only for Database credentials.
oauth	boolean, Whether or not the credential is an OAuth credential.

`credentials_post_temporary` *Generate a temporary credential for accessing S3*

Description

Generate a temporary credential for accessing S3

Usage

```
credentials_post_temporary(id, duration = NULL)
```

Arguments

id	integer required. The ID of the credential.
duration	integer optional. The number of seconds the temporary credential should be valid. Defaults to 15 minutes. Must not be less than 15 minutes or greater than 36 hours.

Value

A list containing the following elements:

accessKey	string, The identifier of the credential.
secretAccessKey	string, The secret part of the credential.
sessionToken	string, The session token identifier.

`credentials_put` *Update an existing credential*

Description

Update an existing credential

Usage

```
credentials_put(
  id,
  type,
  username,
  password,
  name = NULL,
  description = NULL,
  remote_host_id = NULL,
  user_id = NULL,
  state = NULL,
  system_credential = NULL,
  default = NULL,
  oauth = NULL
)
```

Arguments

`id`	integer required. The ID of the credential.
`type`	string required. The type of credential. Note: only these credentials can be created or edited via this API ["Amazon Web Services S3", "CASS/NCOA PAF", "Certificate", "Civis Platform", "Custom", "Database", "Google", "Salesforce User", "Salesforce Client", "TableauUser"]
`username`	string required. The username for the credential.
`password`	string required. The password for the credential.
`name`	string optional. The name identifying the credential.
`description`	string optional. A long description of the credential.
`remote_host_id`	integer optional. The ID of the remote host associated with the credential.
`user_id`	integer optional. The ID of the user the credential is created for. Note: This attribute is only accepted if you are a Civis Admin User.
`state`	string optional. The U.S. state for the credential. Only for VAN credentials.
`system_credential`	boolean optional. Boolean flag that sets a credential to be a system credential. System credentials can only be created by Civis Admins and will create a credential owned by the Civis Robot user.
`default`	boolean optional. Whether or not the credential is a default. Only for Database credentials.
`oauth`	boolean optional. Whether or not the credential is an OAuth credential.

Value

A list containing the following elements:

id	integer, The ID of the credential.
name	string, The name identifying the credential
type	string, The credential's type.
username	string, The username for the credential.
description	string, A long description of the credential.
owner	string, The username of the user who this credential belongs to. Using user.username is preferred.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
remoteHostId	integer, The ID of the remote host associated with this credential.
remoteHostName	string, The name of the remote host associated with this credential.
state	string, The U.S. state for the credential. Only for VAN credentials.
createdAt	string, The creation time for this credential.
updatedAt	string, The last modification time for this credential.
default	boolean, Whether or not the credential is a default. Only for Database credentials.
oauth	boolean, Whether or not the credential is an OAuth credential.

---

credentials_put_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
credentials_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
`totalGroupShares`	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`credentials_put_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
credentials_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

credentials_put_transfer

Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
credentials_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. ID of target user
include_dependencies	boolean required. Whether or not to give manage permissions on all dependencies
email_body	string optional. Custom body text for e-mail sent on transfer.
send_email	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies — array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

databases_delete_schemas_tables_projects

Remove a Table from a project

Description

Remove a Table from a project

Usage

```
databases_delete_schemas_tables_projects(
  id,
  project_id,
  schema_name,
  table_name
)
```

Arguments

id	integer required. The ID of the database
project_id	integer required. The ID of the project.
schema_name	string required. The name of the schema
table_name	string required. The name of the table

Value

An empty HTTP response

---

databases_get *Show database information*

Description

Show database information

Usage

```
databases_get(id)
```

Arguments

id	integer required. The ID for the database.

Value

A list containing the following elements:

id	integer, The ID for the database.
name	string, The name of the database in Platform.
adapter	string, The type of the database.
clusterIdentifier	string, The cluster identifier of the database.
host	string, The host of the database server.
port	integer, The port of the database.
databaseName	string, The internal name of the database.
managed	boolean, True if the database is Civis-managed. False otherwise.

`databases_get_schemas_tables` *Show basic table info*

Description

Show basic table info

Usage

```
databases_get_schemas_tables(id, schema_name, table_name, credential_id = NULL)
```

Arguments

`id`	integer required. The ID of the database
`schema_name`	string required. The name of the schema
`table_name`	string required. The name of the table
`credential_id`	integer optional. If provided, schemas will be filtered based on the given credential.

Value

A list containing the following elements:

`id`	integer, The ID of the table.
`databaseId`	integer, The ID of the database.
`schema`	string, The name of the schema containing the table.
`name`	string, Name of the table.
`description`	string, The description of the table, as specified by the table owner
`isView`	boolean, True if this table represents a view. False if it represents a regular table.
`rowCount`	integer, The number of rows in the table.
`columnCount`	integer, The number of columns in the table.
`sizeMb`	number, The size of the table in megabytes.
`owner`	string, The database username of the table's owner.
`distkey`	string, The column used as the Amazon Redshift distkey.
`sortkeys`	string, The column used as the Amazon Redshift sortkey.
`refreshStatus`	string, How up-to-date the table's statistics on row counts, null counts, distinct counts, and values distributions are. One of: refreshing, stale, or current.
`lastRefresh`	string, The time of the last statistics refresh.
`dataUpdatedAt`	string, The last time that Civis Platform captured a change in this table.Only applicable for Redshift tables; please see the Civis help desk for more info.

`schemaUpdatedAt`
: string, The last time that Civis Platform captured a change to the table attributes/structure.Only applicable for Redshift tables; please see the Civis help desk for more info.

`refreshId`
: string, The ID of the most recent statistics refresh.

`lastRun`
: list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`primaryKeys`
: array, The primary keys for this table.

`lastModifiedKeys`
: array, The columns indicating an entry's modification status for this table.

`tableTags`
: array, An array containing the following fields:

- id integer, Table Tag ID
- name string, Table Tag Name

`ontologyMapping`
: list, The ontology-key to column-name mapping. See /ontology for the list of valid ontology keys.

`columns`
: array, An array containing the following fields:

- name string, Name of the column.
- civisDataType string, The generic data type of the column (ex. "string"). Since this is database-agnostic, it may be helpful when loading data to R/Python.
- sqlType string, The database-specific SQL type of the column (ex. "varchar(30)").
- sampleValues array, A sample of values from the column.
- encoding string, The compression encoding for this columnSee: http://docs.aws.amazon.com/redshift
- description string, The description of the column, as specified by the table owner
- order integer, Relative position of the column in the table.
- minValue string, Smallest value in the column.
- maxValue string, Largest value in the column.
- avgValue number, This parameter is deprecated.
- stddev number, This parameter is deprecated.
- valueDistributionPercent object, A mapping between each value in the column and the percentage of rows with that value.Only present for tables with fewer than approximately 25,000,000 rows and for columns with fewer than twenty distinct values.
- coverageCount integer, Number of non-null values in the column.
- nullCount integer, Number of null values in the column.

- possibleDependentVariableTypes array, Possible dependent variable types the column may be used to model. Null if it may not be used as a dependent variable.
- useableAsIndependentVariable boolean, Whether the column may be used as an independent variable to train a model.
- useableAsPrimaryKey boolean, Whether the column may be used as an primary key to identify table rows.
- valueDistribution object, An object mapping distinct values in the column to the number of times they appear in the column
- distinctCount integer, Number of distinct values in the column. NULL values are counted and treated as a single distinct value.

`joins` array, An array containing the following fields:

- id integer,
- leftTableId integer,
- leftIdentifier string,
- rightTableId integer,
- rightIdentifier string,
- on string,
- leftJoin boolean,
- createdAt string,
- updatedAt string,

`multipartKey` array,

`enhancements` array, An array containing the following fields:

- type string,
- createdAt string,
- updatedAt string,
- joinId integer,

`viewDef` string,

`tableDef` string,

`outgoingTableMatches`
array, An array containing the following fields:

- sourceTableId integer, Source table
- targetType string, Target type
- targetId integer, Target ID
- target object,
- job object,

databases_get_schema_privileges
Show schema privileges

Description

Show schema privileges

Usage

```
databases_get_schema_privileges(id, schema_name, credential_id = NULL)
```

Arguments

id	integer required. The ID of the database
schema_name	string required. The name of the schema
credential_id	integer optional. If provided, schemas will be filtered based on the given credential.

Value

A list containing the following elements:

grantee	string, Name of the granted user or group
granteeType	string, User or group
privileges	array, Privileges that the grantee has on this resource
grantablePrivileges	array, Privileges that the grantee can grant to others for this resource

databases_get_status_graphs_timeframe
Get the status graphs for this database

Description

Get the status graphs for this database

Usage

```
databases_get_status_graphs_timeframe(id, timeframe)
```

Arguments

id	integer required. The ID of the database.
timeframe	string required. The span of time that the graphs cover. Must be one of 1_hour, 4_hours, 1_day, 2_days, 1_week.

Value

A list containing the following elements:

cpuGraphUrl	string, URL for the aws redshift cpu utliization graph.
diskGraphUrl	string, URL for the aws redshift disk usage graph.
queueLengthGraphUrl	string, URL for the aws redshift queue length graph.
statusGraphUrl	string, URL for the aws redshift status graph.
maintenanceGraphUrl	string, URL for the aws redshift maintenance graph.
queryDurationGraphUrl	string, URL for the aws redshift table count graph.

---

databases_get_table_privilegesschema_name

Show table privileges

Description

Show table privileges

Usage

```
databases_get_table_privilegesschema_name(
  id,
  schema_name,
  table_name,
  credential_id = NULL
)
```

Arguments

id	integer required. The ID of the database
schema_name	string required. The name of the schema
table_name	string required. The name of the table
credential_id	integer optional. If provided, schemas will be filtered based on the given credential.

Value

A list containing the following elements:

grantee	string, Name of the granted user or group
granteeType	string, User or group
privileges	array, Privileges that the grantee has on this resource
grantablePrivileges	array, Privileges that the grantee can grant to others for this resource

databases_get_whitelist_ips *View details about a whitelisted IP*

Description

View details about a whitelisted IP

Usage

```
databases_get_whitelist_ips(id, whitelisted_ip_id)
```

Arguments

id	integer required. The ID of the database this rule is applied to.
whitelisted_ip_id	integer required. The ID of this whitelisted IP address.

Value

A list containing the following elements:

id	integer, The ID of this whitelisted IP address.
remoteHostId	integer, The ID of the database this rule is applied to.
securityGroupId	string, The ID of the security group this rule is applied to.
subnetMask	string, The subnet mask that is allowed by this rule.
authorizedBy	string, The user who authorized this rule.
isActive	boolean, True if the rule is applied, false if it has been revoked.
createdAt	string, The time this rule was created.
updatedAt	string, The time this rule was last updated.

databases_list *List databases*

Description

List databases

Usage

```
databases_list()
```

Value

An array containing the following fields:

id	integer, The ID for the database.
name	string, The name of the database in Platform.
adapter	string, The type of the database.
clusterIdentifier	string, The cluster identifier of the database.
host	string, The host of the database server.
port	integer, The port of the database.
databaseName	string, The internal name of the database.
managed	boolean, True if the database is Civis-managed. False otherwise.

---

databases_list_advanced_settings

Get the advanced settings for this database

Description

Get the advanced settings for this database

Usage

```
databases_list_advanced_settings(id)
```

Arguments

id	integer required. The ID of the database this advanced settings object belongs to.

Value

A list containing the following elements:

exportCachingEnabled	boolean, Whether or not caching is enabled for export jobs run on this database server.

databases_list_groups *List groups in the specified database*

Description

List groups in the specified database

Usage

```
databases_list_groups(id)
```

Arguments

id	integer required. The ID for the database.

Value

An array containing the following fields:

groupName	string, The name of the group.
members	array, The members of the group.

databases_list_schemas *List schemas in this database*

Description

List schemas in this database

Usage

```
databases_list_schemas(id, name = NULL, credential_id = NULL)
```

Arguments

id	integer required. The ID of the database.
name	string optional. If specified, will be used to filter the schemas returned.Substring matching is supported (e.g., "name=schema" will return both "schema1" and "schema2"). Does not apply to BigQuery databases.
credential_id	integer optional. If provided, schemas will be filtered based on the given credential.

Value

An array containing the following fields:

schema	string, The name of a schema.

databases_list_schemas_tables
List tables in this schema

Description

List tables in this schema

Usage

```
databases_list_schemas_tables(id, schema_name, credential_id = NULL)
```

Arguments

id	integer required. The ID of the database
schema_name	string required. The name of the schema
credential_id	integer optional. If provided, schemas will be filtered based on the given credential.

Value

An array containing the following fields:

name	string, The name of the table.
schema	string, The name of the schema containing the table.
isView	boolean, True if this table represents a view. False if it represents a regular table.
databaseId	integer, The ID of the database server.

databases_list_schemas_tables_projects
List the projects a Table belongs to

Description

List the projects a Table belongs to

Usage

```
databases_list_schemas_tables_projects(
  id,
  schema_name,
  table_name,
  hidden = NULL
)
```

Arguments

id	integer required. The ID of the database
schema_name	string required. The name of the schema
table_name	string required. The name of the table
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

---

databases_list_tables *List tables in the specified database, deprecated use "GET /tables" instead*

Description

List tables in the specified database, deprecated use "GET /tables" instead

Usage

```
databases_list_tables(
  id,
  name = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the database.
name	string optional. If specified, will be used to filter the tables returned. Substring matching is supported (e.g., "name=table" will return both "table1" and "my table").
limit	integer optional. Number of results to return. Defaults to 200. Maximum allowed is 1000.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to name. Must be one of: name.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id	integer, The ID of the table.
databaseId	integer, The ID of the database.
schema	string, The name of the schema containing the table.
name	string, Name of the table.
description	string, The description of the table, as specified by the table owner
isView	boolean, True if this table represents a view. False if it represents a regular table.
rowCount	integer, The number of rows in the table.
columnCount	integer, The number of columns in the table.
sizeMb	number, The size of the table in megabytes.
owner	string, The database username of the table's owner.
distkey	string, The column used as the Amazon Redshift distkey.
sortkeys	string, The column used as the Amazon Redshift sortkey.
refreshStatus	string, How up-to-date the table's statistics on row counts, null counts, distinct counts, and values distributions are. One of: refreshing, stale, or current.

lastRefresh	string, The time of the last statistics refresh.
refreshId	string, The ID of the most recent statistics refresh.
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
tableTags	array, An array containing the following fields: • id integer, Table Tag ID • name string, Table Tag Name

---

`databases_list_tables_search`

List tables in the specified database, deprecated use "GET /tables" instead

---

Description

List tables in the specified database, deprecated use "GET /tables" instead

Usage

```
databases_list_tables_search(id, name = NULL, column_name = NULL)
```

Arguments

id	integer required. The ID of the database.
name	string optional. If specified, will be used to filter the tables returned. Substring matching is supported (e.g., "name=table" will return both "table1" and "my table").
column_name	string optional. Search for tables containing a column with the given name.

Value

An array containing the following fields:

id	integer, The ID of the table.
databaseId	integer, The ID of the database.
schema	string, The name of the schema containing the table.
name	string, Name of the table.
description	string, The description of the table, as specified by the table owner

isView	boolean, True if this table represents a view. False if it represents a regular table.
rowCount	integer, The number of rows in the table.
columnCount	integer, The number of columns in the table.
sizeMb	number, The size of the table in megabytes.
owner	string, The database username of the table's owner.
distkey	string, The column used as the Amazon Redshift distkey.
sortkeys	string, The column used as the Amazon Redshift sortkey.
refreshStatus	string, How up-to-date the table's statistics on row counts, null counts, distinct counts, and values distributions are. One of: refreshing, stale, or current.
lastRefresh	string, The time of the last statistics refresh.
refreshId	string, The ID of the most recent statistics refresh.
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
tableTags	array, An array containing the following fields: • id integer, Table Tag ID • name string, Table Tag Name
columnNames	array, The names of each column in the table.

`databases_list_users` *Show list of database users*

Description

Show list of database users

Usage

```
databases_list_users(id, active = NULL)
```

Arguments

id	integer required. The ID of the database.
active	boolean optional. If true returns active users. If false returns deactivated users. Defaults to true.

Value

A list containing the following elements:

username	string, Username
active	boolean, Whether the user is active or deactivated

databases_list_whitelist_ips

List whitelisted IPs for the specified database

Description

List whitelisted IPs for the specified database

Usage

```
databases_list_whitelist_ips(id)
```

Arguments

id	integer required. The ID for the database.

Value

An array containing the following fields:

id	integer, The ID of this whitelisted IP address.
remoteHostId	integer, The ID of the database this rule is applied to.
securityGroupId	string, The ID of the security group this rule is applied to.
subnetMask	string, The subnet mask that is allowed by this rule.
createdAt	string, The time this rule was created.
updatedAt	string, The time this rule was last updated.

databases_patch_advanced_settings

Update the advanced settings for this database

Description

Update the advanced settings for this database

Usage

```
databases_patch_advanced_settings(id, export_caching_enabled = NULL)
```

Arguments

id	integer required. The ID of the database this advanced settings object belongs to.
export_caching_enabled	boolean optional. Whether or not caching is enabled for export jobs run on this database server.

Value

A list containing the following elements:

`exportCachingEnabled`
: boolean, Whether or not caching is enabled for export jobs run on this database server.

`databases_patch_schemas_tables` *Update a table*

Description

Update a table

Usage

```
databases_patch_schemas_tables(
  id,
  schema_name,
  table_name,
  credential_id = NULL,
  description = NULL
)
```

Arguments

`id`	integer required. The ID of the database
`schema_name`	string required. The name of the schema
`table_name`	string required. The name of the table
`credential_id`	integer optional. If provided, schemas will be filtered based on the given credential.
`description`	string optional. The user-defined description of the table.

Value

A list containing the following elements:

`id`	integer, The ID of the table.
`databaseId`	integer, The ID of the database.
`schema`	string, The name of the schema containing the table.
`name`	string, Name of the table.
`description`	string, The description of the table, as specified by the table owner
`isView`	boolean, True if this table represents a view. False if it represents a regular table.

`rowCount` integer, The number of rows in the table.

`columnCount` integer, The number of columns in the table.

`sizeMb` number, The size of the table in megabytes.

`owner` string, The database username of the table's owner.

`distkey` string, The column used as the Amazon Redshift distkey.

`sortkeys` string, The column used as the Amazon Redshift sortkey.

`refreshStatus` string, How up-to-date the table's statistics on row counts, null counts, distinct counts, and values distributions are. One of: refreshing, stale, or current.

`lastRefresh` string, The time of the last statistics refresh.

`dataUpdatedAt` string, The last time that Civis Platform captured a change in this table.Only applicable for Redshift tables; please see the Civis help desk for more info.

`schemaUpdatedAt` string, The last time that Civis Platform captured a change to the table attributes/structure.Only applicable for Redshift tables; please see the Civis help desk for more info.

`refreshId` string, The ID of the most recent statistics refresh.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`primaryKeys` array, The primary keys for this table.

`lastModifiedKeys` array, The columns indicating an entry's modification status for this table.

`tableTags` array, An array containing the following fields:

- id integer, Table Tag ID
- name string, Table Tag Name

`ontologyMapping` list, The ontology-key to column-name mapping. See /ontology for the list of valid ontology keys.

`columns` array, An array containing the following fields:

- name string, Name of the column.
- civisDataType string, The generic data type of the column (ex. "string"). Since this is database-agnostic, it may be helpful when loading data to R/Python.
- sqlType string, The database-specific SQL type of the column (ex. "varchar(30)").
- sampleValues array, A sample of values from the column.
- encoding string, The compression encoding for this columnSee: http://docs.aws.amazon.com/redshift

- description string, The description of the column, as specified by the table owner
- order integer, Relative position of the column in the table.
- minValue string, Smallest value in the column.
- maxValue string, Largest value in the column.
- avgValue number, This parameter is deprecated.
- stddev number, This parameter is deprecated.
- valueDistributionPercent object, A mapping between each value in the column and the percentage of rows with that value.Only present for tables with fewer than approximately 25,000,000 rows and for columns with fewer than twenty distinct values.
- coverageCount integer, Number of non-null values in the column.
- nullCount integer, Number of null values in the column.
- possibleDependentVariableTypes array, Possible dependent variable types the column may be used to model. Null if it may not be used as a dependent variable.
- useableAsIndependentVariable boolean, Whether the column may be used as an independent variable to train a model.
- useableAsPrimaryKey boolean, Whether the column may be used as an primary key to identify table rows.
- valueDistribution object, An object mapping distinct values in the column to the number of times they appear in the column
- distinctCount integer, Number of distinct values in the column. NULL values are counted and treated as a single distinct value.

`joins` array, An array containing the following fields:

- id integer,
- leftTableId integer,
- leftIdentifier string,
- rightTableId integer,
- rightIdentifier string,
- on string,
- leftJoin boolean,
- createdAt string,
- updatedAt string,

`multipartKey` array,

`enhancements` array, An array containing the following fields:

- type string,
- createdAt string,
- updatedAt string,
- joinId integer,

`viewDef` string,

`tableDef` string,

`outgoingTableMatches`
: array, An array containing the following fields:

  - sourceTableId integer, Source table
  - targetType string, Target type
  - targetId integer, Target ID
  - target object,
  - job object,

`databases_post_schemas_scan`
Creates and enqueues a schema scanner job

Description

Creates and enqueues a schema scanner job

Usage

```
databases_post_schemas_scan(id, schema, stats_priority = NULL)
```

Arguments

`id`	integer required. The ID of the database.
`schema`	string required. The name of the schema.
`stats_priority`	string optional. When to sync table statistics for every table in the schema. Valid options are the following. Option: 'flag' means to flag stats for the next scheduled run of a full table scan on the database. Option: 'block' means to block this job on stats syncing. Option: 'queue' means to queue a separate job for syncing stats and do not block this job on the queued job. Defaults to 'flag'

Value

A list containing the following elements:

`jobId`	integer, The ID of the job created.
`runId`	integer, The ID of the run created.

databases_put_advanced_settings

Edit the advanced settings for this database

Description

Edit the advanced settings for this database

Usage

```
databases_put_advanced_settings(id, export_caching_enabled)
```

Arguments

id	integer required. The ID of the database this advanced settings object belongs to.
export_caching_enabled	boolean required. Whether or not caching is enabled for export jobs run on this database server.

Value

A list containing the following elements:

exportCachingEnabled	boolean, Whether or not caching is enabled for export jobs run on this database server.

databases_put_schemas_tables_projects

Add a Table to a project

Description

Add a Table to a project

Usage

```
databases_put_schemas_tables_projects(id, project_id, schema_name, table_name)
```

Arguments

id	integer required. The ID of the database
project_id	integer required. The ID of the project.
schema_name	string required. The name of the schema
table_name	string required. The name of the table

Value

An empty HTTP response

`default_credential` *The current user's default credential.*

Description

The current user's default credential.

Usage

```
default_credential()
```

Value

credential_id, id of the default credential for the current user.

`download_civis` *Download a table or a file from the Civis Platform to local disk*

Description

`download_civis` downloads a file based on the type of its first argument, which can be a string `"schema.table"`, a SQL query `sql(...)`, or a numeric file ID.

A table or a query from Redshift will be downloaded onto disk as a CSV. A file from Platform files endpoint will be downloaded as is.

A default database can be set using `options(civis.default_db = "my_database")`. If there is only one database available, this database will automatically be used as the default.

Usage

```
download_civis(x, ...)

## S3 method for class 'character'
download_civis(
  x,
  database = NULL,
  file,
  overwrite = FALSE,
  progress = FALSE,
  split = FALSE,
  job_name = NULL,
  hidden = TRUE,
```

```
  verbose = FALSE,
  ...
)

## S3 method for class 'sql'
download_civis(
  x,
  database = NULL,
  file,
  overwrite = FALSE,
  progress = FALSE,
  split = FALSE,
  job_name = NULL,
  hidden = TRUE,
  verbose = FALSE,
  ...
)

## S3 method for class 'numeric'
download_civis(x, file, overwrite = FALSE, progress = FALSE, ...)
```

Arguments

x	"schema.table", sql("query"), or a file id.
...	Currently ignored.
database	string, The database. If NULL, tries to use the default database.
file	string, The file to write to.
overwrite	logical, Whether to overwrite the existing file.
progress	logical, Whether to display a progress bar.
split	logical, Whether to download a big table by splitting it into multiple CSV parts first. See civis_to_multifile_csv for details.
job_name	string, Name of the job (default: "Civis Download Via R Client").
hidden	logical, Whether the job is hidden on Platform.
verbose	logical, Whether to print detailed updates of job status.

Value

The file where the downloaded files or tables are written to. It is returned invisibly.

Methods (by class)

- download_civis(character): Download a table from Redshift to disk as CSV.
- download_civis(sql): Download the result of a SQL query from Redshift to disk as CSV.
- download_civis(numeric): Download a file from Platform files endpoint to disk.

See Also

Other io: query_civis(), query_civis_file(), read_civis(), write_civis(), write_civis_file()

Examples

```
## Not run:
# Download all columns in a single table into a CSV
download_civis("schema.table", database = "my_database",
               file = "~/Downloads/my_table.csv")

# Download data from a SQL select statement into a CSV
query <- sql("SELECT * FROM table JOIN other_table USING id WHERE var1 < 23")
download_civis(query, database = "my_database",
               file = "~/Downloads/my_table.csv")

# Set a default database
options(civis.default_db = "my_database")

# Download any file from the files endpoint.
file_id <- write_civis_file(df)
download_civis(file_id, file = "df.rds", progress = TRUE)
df2 <- readRDS("df.rds")
identical(df, df2)

## End(Not run)
```

---

endpoints_list *List API endpoints*

Description

List API endpoints

Usage

```
endpoints_list()
```

Value

An empty HTTP response

enhancements_delete_cass_ncoa_projects

Remove a CASS/NCOA Enhancement from a project

Description

Remove a CASS/NCOA Enhancement from a project

Usage

```
enhancements_delete_cass_ncoa_projects(id, project_id)
```

Arguments

id	integer required. The ID of the CASS/NCOA Enhancement.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

enhancements_delete_cass_ncoa_runs

Cancel a run

Description

Cancel a run

Usage

```
enhancements_delete_cass_ncoa_runs(id, run_id)
```

Arguments

id	integer required. The ID of the CASS NCOA job.
run_id	integer required. The ID of the run.

Value

An empty HTTP response

enhancements_delete_cass_ncoa_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
enhancements_delete_cass_ncoa_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

enhancements_delete_cass_ncoa_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
enhancements_delete_cass_ncoa_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

enhancements_delete_civis_data_match_projects

Remove a Civis Data Match Enhancement from a project

Description

Remove a Civis Data Match Enhancement from a project

Usage

```
enhancements_delete_civis_data_match_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Civis Data Match Enhancement.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

enhancements_delete_civis_data_match_runs

Cancel a run

Description

Cancel a run

Usage

```
enhancements_delete_civis_data_match_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Civis Data Match job.
run_id	integer required. The ID of the run.

Value

An empty HTTP response

enhancements_delete_civis_data_match_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
enhancements_delete_civis_data_match_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

enhancements_delete_civis_data_match_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
enhancements_delete_civis_data_match_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

enhancements_delete_geocode_projects

Remove a Geocode Enhancement from a project

Description

Remove a Geocode Enhancement from a project

Usage

```
enhancements_delete_geocode_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Geocode Enhancement.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

enhancements_delete_geocode_runs

Cancel a run

Description

Cancel a run

Usage

```
enhancements_delete_geocode_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Geocode job.
run_id	integer required. The ID of the run.

Value

An empty HTTP response

enhancements_delete_geocode_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
enhancements_delete_geocode_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

enhancements_delete_geocode_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
enhancements_delete_geocode_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

enhancements_delete_identity_resolution_projects

Remove an Identity Resolution Enhancement from a project

Description

Remove an Identity Resolution Enhancement from a project

Usage

```
enhancements_delete_identity_resolution_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Identity Resolution Enhancement.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

enhancements_delete_identity_resolution_runs

Cancel a run

Description

Cancel a run

Usage

```
enhancements_delete_identity_resolution_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Identity Resolution job.
run_id	integer required. The ID of the run.

Value

An empty HTTP response

enhancements_delete_identity_resolution_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
enhancements_delete_identity_resolution_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

enhancements_delete_identity_resolution_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
enhancements_delete_identity_resolution_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

enhancements_get_cass_ncoa *Get a CASS/NCOA Enhancement*

Description

Get a CASS/NCOA Enhancement

Usage

```
enhancements_get_cass_ncoa(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for the enhancement.
name	string, The name of the enhancement job.
type	string, The type of the enhancement (e.g CASS-NCOA)
createdAt	string, The time this enhancement was created.
updatedAt	string, The time the enhancement was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the enhancement's last run
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list, A list containing the following elements: • urls array, URLs to receive a POST request at job completion

- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`source` list, A list containing the following elements:

- databaseTable list . A list containing the following elements:
  - schema string, The schema name of the source table.
  - table string, The name of the source table.
  - remoteHostId integer, The ID of the database host for the table.
  - credentialId integer, The id of the credentials to be used when performing the enhancement.
  - multipartKey array, The source table primary key.

`destination` list, A list containing the following elements:

- databaseTable list . A list containing the following elements:
  - schema string, The schema name for the output data.
  - table string, The table name for the output data.

`columnMapping` list, A list containing the following elements:

- address1 string, The first address line.
- address2 string, The second address line.
- city string, The city of an address.
- state string, The state of an address.

- zip string, The zip code of an address.
- name string, The full name of the resident at this address. If needed, separate multiple columns with '+', e.g. 'first_name+last_name'
- company string, The name of the company located at this address.

`useDefaultColumnMapping`
: boolean, Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

`performNcoa`
: boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.

`ncoaCredentialId`
: integer, Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

`outputLevel`
: string, The set of fields persisted by a CASS or NCOA enhancement.For CASS enhancements, one of 'cass' or 'all.'For NCOA enhancements, one of 'cass', 'ncoa' , 'coalesced' or 'all'.By default, all fields will be returned.

`limitingSQL`
: string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

`batchSize`
: integer, The maximum number of records processed at a time. Note that this parameter is not available to all users.

`archived`
: string, The archival status of the requested item(s).

`parentId`
: integer, Parent ID that triggers this enhancement.

`enhancements_get_cass_ncoa_runs`
Check status of a run

Description

Check status of a run

Usage

```
enhancements_get_cass_ncoa_runs(id, run_id)
```

Arguments

`id`
: integer required. The ID of the CASS NCOA job.

`run_id`
: integer required. The ID of the run.

Value

A list containing the following elements:

id	integer, The ID of the run.
cassNcoaId	integer, The ID of the CASS NCOA job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

---

enhancements_get_civis_data_match

Get a Civis Data Match Enhancement

Description

Get a Civis Data Match Enhancement

Usage

```
enhancements_get_civis_data_match(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for the enhancement.
name	string, The name of the enhancement job.
type	string, The type of the enhancement (e.g CASS-NCOA)
createdAt	string, The time this enhancement was created.
updatedAt	string, The time the enhancement was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username.

- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`inputFieldMapping` list, The field (i.e., column) mapping for the input table. See https://api.civisanalytics.com/enhancements/f mapping for a list of valid field types and descriptions. Each field type should be mapped to a string specifying a column name in the input table. For field types that support multiple values (e.g., the "phone" field), a list of column names can be provided (e.g., {"phone": ["home_phone", "mobile_phone"], ...}).

inputTable	list, A list containing the following elements: • databaseName string, The Redshift database name for the table. • schema string, The schema name for the table. • table string, The table name.
matchTargetId	integer, The ID of the Civis Data match target. See /match_targets for IDs.
outputTable	list, A list containing the following elements: • databaseName string, The Redshift database name for the table. • schema string, The schema name for the table. • table string, The table name.
maxMatches	integer, The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.
threshold	number, The score threshold (between 0 and 1). Matches below this threshold will not be returned. The default value is 0.5.
archived	boolean, Whether the Civis Data Match Job has been archived.
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
parentId	integer, Parent ID that triggers this enhancement.

`enhancements_get_civis_data_match_runs`
Check status of a run

Description

Check status of a run

Usage

```
enhancements_get_civis_data_match_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Civis Data Match job.
run_id	integer required. The ID of the run.

Value

A list containing the following elements:

id	integer, The ID of the run.
civisDataMatchId	integer, The ID of the Civis Data Match job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

---

enhancements_get_geocode

Get a Geocode Enhancement

Description

Get a Geocode Enhancement

Usage

```
enhancements_get_geocode(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for the enhancement.
name	string, The name of the enhancement job.
type	string, The type of the enhancement (e.g CASS-NCOA)
createdAt	string, The time this enhancement was created.
updatedAt	string, The time the enhancement was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username.

- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`remoteHostId` integer, The ID of the remote host.

`credentialId` integer, The ID of the remote host credential.

`sourceSchemaAndTable` string, The source database schema and table.

multipartKey	array, The source table primary key.
limitingSQL	string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
targetSchema	string, The output table schema.
targetTable	string, The output table name.
country	string, The country of the addresses to be geocoded; either 'us' or 'ca'.
provider	string, The geocoding provider; one of postgis and geocoder_ca.
outputAddress	boolean, Whether to output the parsed address. Only guaranteed for the 'postgis' provider.
archived	string, The archival status of the requested item(s).
parentId	integer, Parent ID that triggers this enhancement.

`enhancements_get_geocode_runs`
Check status of a run

Description

Check status of a run

Usage

```
enhancements_get_geocode_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Geocode job.
run_id	integer required. The ID of the run.

Value

A list containing the following elements:

id	integer, The ID of the run.
geocodeId	integer, The ID of the Geocode job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

enhancements_get_identity_resolution
Get an Identity Resolution Enhancement

Description

Get an Identity Resolution Enhancement

Usage

```
enhancements_get_identity_resolution(id, is_legacy_id = NULL)
```

Arguments

id	integer required. The ID for the enhancement.
is_legacy_id	boolean optional. Whether the given ID is for the Identity Resolution job in the legacy service app.

Value

A list containing the following elements:

id	integer, The ID for the enhancement.
name	string, The name of the enhancement job.
type	string, The type of the enhancement (e.g CASS-NCOA)
createdAt	string, The time this enhancement was created.
updatedAt	string, The time the enhancement was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the enhancement's last run
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`threshold` number, A value that determines the extent to which similar records get assigned the same resolved ID. Must be within 0.5 and 1, inclusive. Defaults to 0.8 if unspecified.Higher values may result in fewer cases where records about different individuals erroneously receive the same resolved ID, but also more more cases where records about the same individual receive different resolved IDs.

`sources` array, An array containing the following fields:

- name string, A user-specified name for the source.
- description string, A description of the source.
- databaseName string, The name of the source database.
- schemaName string, The name of the source schema.
- tableName string, The name of the source table.
- fieldMapping object, A mapping of PII fields to columns in this table. Valid keys are primary_key, first_name, middle_name, last_name, gender, phone, email, birth_date, birth_year, birth_month, birth_day, house_number, street, unit, full_address, city, state, state_code, zip, lat, lon, and name_suffix

`matchTargetId` integer, The ID of the Civis Data (Custom) match target. See /match_targets for IDs.

`enforcedLinks` array, An array containing the following fields:

- source1 string, Name of the first source. Must be defined in Sources list.
- source1JoinCol string, Column from the first source to join on.
- source2 string, Name of the second source. Must be defined in Sources list
- source2JoinCol string, Column from the second source to join on.

`customerGraph` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`goldenTable` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.
- fields array,

`linkScores` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`legacyId` integer, ID of this pipeline in the legacy IDR service application.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.
- config string, How the Identity Resolution job was configured for this run.
- sampleRecordsQuery string, A SQL query to produce a sample of records to inspect.
- expandClusterQuery string, A customizable query to view PII associated with resolved ids.
- runMetrics list . A list containing the following elements:
  - numRecords integer, The number of input records for this run.
  - uniqueIds integer, The number of distinct unique IDs in the input records for this run.
  - uniqueDeduplicatedIds integer, The number of resolved IDs associated with more than one unique ID in the input.
  - maxClusterSize integer, The number of records in the largest cluster of resolved IDs.

  - avgClusterSize number, The average number of records with the same resolved ID.
  - clusterSizeFrequencies object, A mapping from numbers of records with the same resolved ID (i.e., sizes of clusters) to numbers of such clusters. For example, if there were 10 clusters with 2 records each, 2 would be a key in the mapping, and 10 would be its value.
- errorSection string, If there was a failure, this will denote which section of the Identity Resolution job failed. One of: data_preparation, compute_setup or data_processing.

`enhancements_get_identity_resolution_runs` *Check status of a run*

Description

Check status of a run

Usage

```
enhancements_get_identity_resolution_runs(id, run_id)
```

Arguments

`id`	integer required. The ID of the Identity Resolution job.
`run_id`	integer required. The ID of the run.

Value

A list containing the following elements:

`id`	integer, The ID of the run.
`identityResolutionId`	integer, The ID of the Identity Resolution job.
`state`	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
`isCancelRequested`	boolean, True if run cancel requested, else false.
`createdAt`	string, The time the run was created.
`startedAt`	string, The time the run started at.
`finishedAt`	string, The time the run completed.
`error`	string, The error, if any, returned by the run.
`config`	string, How the Identity Resolution job was configured for this run.
`sampleRecordsQuery`	string, A SQL query to produce a sample of records to inspect.

`expandClusterQuery`
: string, A customizable query to view PII associated with resolved ids.

`runMetrics`
: list, A list containing the following elements:

- numRecords integer, The number of input records for this run.
- uniqueIds integer, The number of distinct unique IDs in the input records for this run.
- uniqueDeduplicatedIds integer, The number of resolved IDs associated with more than one unique ID in the input.
- maxClusterSize integer, The number of records in the largest cluster of resolved IDs.
- avgClusterSize number, The average number of records with the same resolved ID.
- clusterSizeFrequencies object, A mapping from numbers of records with the same resolved ID (i.e., sizes of clusters) to numbers of such clusters. For example, if there were 10 clusters with 2 records each, 2 would be a key in the mapping, and 10 would be its value.

`errorSection`
: string, If there was a failure, this will denote which section of the Identity Resolution job failed. One of: data_preparation, compute_setup or data_processing.

`enhancements_list` *List Enhancements*

Description

List Enhancements

Usage

```
enhancements_list(
  type = NULL,
  author = NULL,
  status = NULL,
  archived = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

`type`
: string optional. If specified, return items of these types.

`author`
: string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.

status	string optional. If specified, returns items with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'failed', 'succeeded', 'idle', 'scheduled'.
archived	string optional. The archival status of the requested item(s).
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at, last_run.updated_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID for the enhancement.
name	string, The name of the enhancement job.
type	string, The type of the enhancement (e.g CASS-NCOA)
createdAt	string, The time this enhancement was created.
updatedAt	string, The time the enhancement was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the enhancement's last run
archived	string, The archival status of the requested item(s).

---

enhancements_list_cass_ncoa_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
enhancements_list_cass_ncoa_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

---

enhancements_list_cass_ncoa_projects

List the projects a CASS/NCOA Enhancement belongs to

Description

List the projects a CASS/NCOA Enhancement belongs to

Usage

```
enhancements_list_cass_ncoa_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the CASS/NCOA Enhancement.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name.

- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

`enhancements_list_cass_ncoa_runs`

List runs for the given CASS NCOA job

Description

List runs for the given CASS NCOA job

Usage

```
enhancements_list_cass_ncoa_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the CASS NCOA job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
cassNcoaId	integer, The ID of the CASS NCOA job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

---

enhancements_list_cass_ncoa_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

```
enhancements_list_cass_ncoa_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the CASS NCOA job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

enhancements_list_cass_ncoa_runs_outputs
List the outputs for a run

Description

List the outputs for a run

Usage

```
enhancements_list_cass_ncoa_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the job.
run_id	integer required. The ID of the run.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

enhancements_list_cass_ncoa_shares
List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
enhancements_list_cass_ncoa_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers list, A list containing the following elements:

- users array,
- groups array,

writers list, A list containing the following elements:

- users array,
- groups array,

owners list, A list containing the following elements:

- users array,
- groups array,

totalUserShares
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

enhancements_list_civis_data_match_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
enhancements_list_civis_data_match_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

enhancements_list_civis_data_match_projects

List the projects a Civis Data Match Enhancement belongs to

Description

List the projects a Civis Data Match Enhancement belongs to

Usage

```
enhancements_list_civis_data_match_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the Civis Data Match Enhancement.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

---

enhancements_list_civis_data_match_runs

List runs for the given Civis Data Match job

Description

List runs for the given Civis Data Match job

Usage

```
enhancements_list_civis_data_match_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the Civis Data Match job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
civisDataMatchId	integer, The ID of the Civis Data Match job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

---

enhancements_list_civis_data_match_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

```
enhancements_list_civis_data_match_runs_logs(
  id,
  run_id,
  last_id = NULL,
  limit = NULL
)
```

Arguments

id	integer required. The ID of the Civis Data Match job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

---

enhancements_list_civis_data_match_runs_outputs

List the outputs for a run

Description

List the outputs for a run

Usage

```
enhancements_list_civis_data_match_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the job.
run_id	integer required. The ID of the run.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.

order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

---

enhancements_list_civis_data_match_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
enhancements_list_civis_data_match_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,

totalUserShares
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

enhancements_list_field_mapping

List the fields in a field mapping for Civis Data Match, Data Unification, and Table Deduplication jobs

---

Description

List the fields in a field mapping for Civis Data Match, Data Unification, and Table Deduplication jobs

Usage

```
enhancements_list_field_mapping()
```

Value

An array containing the following fields:

field	string, The name of the field.
description	string, The description of the field.

---

enhancements_list_geocode_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
enhancements_list_geocode_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

---

enhancements_list_geocode_projects

List the projects a Geocode Enhancement belongs to

Description

List the projects a Geocode Enhancement belongs to

Usage

```
enhancements_list_geocode_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the Geocode Enhancement.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.

description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

`enhancements_list_geocode_runs`
List runs for the given Geocode job

Description

List runs for the given Geocode job

Usage

```
enhancements_list_geocode_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the Geocode job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
geocodeId	integer, The ID of the Geocode job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

---

enhancements_list_geocode_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

```
enhancements_list_geocode_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the Geocode job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

`enhancements_list_geocode_runs_outputs` *List the outputs for a run*

Description

List the outputs for a run

Usage

```
enhancements_list_geocode_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

`id`	integer required. The ID of the job.
`run_id`	integer required. The ID of the run.
`limit`	integer optional. Number of results to return. Defaults to its maximum of 50.
`page_num`	integer optional. Page number of the results to return. Defaults to the first page, 1.
`order`	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
`order_dir`	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

`objectType`	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
`objectId`	integer, The ID of the output.
`name`	string, The name of the output.
`link`	string, The hypermedia link to the output.
`value`	string,

enhancements_list_geocode_shares
List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
enhancements_list_geocode_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers — list, A list containing the following elements:

- users array,
- groups array,

writers — list, A list containing the following elements:

- users array,
- groups array,

owners — list, A list containing the following elements:

- users array,
- groups array,

totalUserShares — integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares — integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

enhancements_list_identity_resolution
List Identity Resolution Enhancements

Description

List Identity Resolution Enhancements

Usage

```
enhancements_list_identity_resolution(
  archived = NULL,
  status = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

archived	string optional. The archival status of the requested item(s).
status	string optional. If specified, returns items with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'failed', 'succeeded', 'idle', 'scheduled'.
limit	integer optional. Number of results to return. Defaults to 50. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, created_at, last_run_updated_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID for the enhancement.
name	string, The name of the enhancement job.
type	string, The type of the enhancement (e.g CASS-NCOA)
createdAt	string, The time this enhancement was created.
updatedAt	string, The time the enhancement was last updated.
author	list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`archived` string, The archival status of the requested item(s).

`sources` array, An array containing the following fields:

- name string, A user-specified name for the source.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.
- sampleRecordsQuery string, A SQL query to produce a sample of records to inspect.
- expandClusterQuery string, A customizable query to view PII associated with resolved ids.
- runMetrics list . A list containing the following elements:
  - numRecords integer, The number of input records for this run.
  - uniqueIds integer, The number of distinct unique IDs in the input records for this run.
  - uniqueDeduplicatedIds integer, The number of resolved IDs associated with more than one unique ID in the input.
  - maxClusterSize integer, The number of records in the largest cluster of resolved IDs.
  - avgClusterSize number, The average number of records with the same resolved ID.
  - clusterSizeFrequencies object, A mapping from numbers of records with the same resolved ID (i.e., sizes of clusters) to numbers of such clusters. For example, if there were 10 clusters with 2 records each, 2 would be a key in the mapping, and 10 would be its value.

`enhancements_list_identity_resolution_dependencies`

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
enhancements_list_identity_resolution_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

---

enhancements_list_identity_resolution_projects

List the projects an Identity Resolution Enhancement belongs to

---

Description

List the projects an Identity Resolution Enhancement belongs to

Usage

```
enhancements_list_identity_resolution_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the Identity Resolution Enhancement.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

`id`	integer, The ID for this project.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The name of this project.
`description`	string, A description of the project.
`users`	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`autoShare`	boolean,
`createdAt`	string,
`updatedAt`	string,
`archived`	string, The archival status of the requested item(s).

`enhancements_list_identity_resolution_runs`
List runs for the given Identity Resolution job

Description

List runs for the given Identity Resolution job

Usage

```
enhancements_list_identity_resolution_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the Identity Resolution job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
identityResolutionId	integer, The ID of the Identity Resolution job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
sampleRecordsQuery	string, A SQL query to produce a sample of records to inspect.
expandClusterQuery	string, A customizable query to view PII associated with resolved ids.
runMetrics	list, A list containing the following elements:

- numRecords integer, The number of input records for this run.
- uniqueIds integer, The number of distinct unique IDs in the input records for this run.
- uniqueDeduplicatedIds integer, The number of resolved IDs associated with more than one unique ID in the input.
- maxClusterSize integer, The number of records in the largest cluster of resolved IDs.
- avgClusterSize number, The average number of records with the same resolved ID.
- clusterSizeFrequencies object, A mapping from numbers of records with the same resolved ID (i.e., sizes of clusters) to numbers of such clusters. For example, if there were 10 clusters with 2 records each, 2 would be a key in the mapping, and 10 would be its value.

enhancements_list_identity_resolution_runs_logs
Get the logs for a run

Description

Get the logs for a run

Usage

```
enhancements_list_identity_resolution_runs_logs(
  id,
  run_id,
  last_id = NULL,
  limit = NULL
)
```

Arguments

id	integer required. The ID of the Identity Resolution job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

`enhancements_list_identity_resolution_shares` *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
enhancements_list_identity_resolution_shares(id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

enhancements_list_types *List available enhancement types*

Description

List available enhancement types

Usage

```
enhancements_list_types()
```

Value

An array containing the following fields:

`name` string, The name of the type.

enhancements_patch_cass_ncoa *Update some attributes of this CASS/NCOA Enhancement*

Description

Update some attributes of this CASS/NCOA Enhancement

Usage

```
enhancements_patch_cass_ncoa(
  id,
  name = NULL,
  schedule = NULL,
  notifications = NULL,
  source = NULL,
  destination = NULL,
  column_mapping = NULL,
  use_default_column_mapping = NULL,
  perform_ncoa = NULL,
  ncoa_credential_id = NULL,
  output_level = NULL,
  limiting_sql = NULL,
  batch_size = NULL,
  parent_id = NULL
)
```

Arguments

id	integer required. The ID for the enhancement.
name	string optional. The name of the enhancement job.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully. • successEmailFromName string, Name from which success emails are sent; defaults to "Civis." • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job. • failureEmailAddresses array, Addresses to notify by e-mail when the job fails. • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes. • successOn boolean, If success email notifications are on. Defaults to user's preferences. • failureOn boolean, If failure email notifications are on. Defaults to user's preferences.
source	list optional. A list containing the following elements: • databaseTable list . A list containing the following elements: – schema string, The schema name of the source table. – table string, The name of the source table. – remoteHostId integer, The ID of the database host for the table. – credentialId integer, The id of the credentials to be used when performing the enhancement. – multipartKey array, The source table primary key.
destination	list optional. A list containing the following elements: • databaseTable list . A list containing the following elements: – schema string, The schema name for the output data.

- table string, The table name for the output data.

`column_mapping` list optional. A list containing the following elements:

- address1 string, The first address line.
- address2 string, The second address line.
- city string, The city of an address.
- state string, The state of an address.
- zip string, The zip code of an address.
- name string, The full name of the resident at this address. If needed, separate multiple columns with '+', e.g. 'first_name+last_name'
- company string, The name of the company located at this address.

`use_default_column_mapping`
boolean optional. Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

`perform_ncoa` boolean optional. Whether to update addresses for records matching the National Change of Address (NCOA) database.

`ncoa_credential_id`
integer optional. Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

`output_level` string optional. The set of fields persisted by a CASS or NCOA enhancement.For CASS enhancements, one of 'cass' or 'all.'For NCOA enhancements, one of 'cass', 'ncoa' , 'coalesced' or 'all'.By default, all fields will be returned.

`limiting_sql` string optional. The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

`batch_size` integer optional. The maximum number of records processed at a time. Note that this parameter is not available to all users.

`parent_id` integer optional. Parent ID that triggers this enhancement.

Value

A list containing the following elements:

`id` integer, The ID for the enhancement.

`name` string, The name of the enhancement job.

`type` string, The type of the enhancement (e.g CASS-NCOA)

`createdAt` string, The time this enhancement was created.

`updatedAt` string, The time the enhancement was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`source` list, A list containing the following elements:

- databaseTable list . A list containing the following elements:
  - schema string, The schema name of the source table.
  - table string, The name of the source table.
  - remoteHostId integer, The ID of the database host for the table.
  - credentialId integer, The id of the credentials to be used when performing the enhancement.

- multipartKey array, The source table primary key.

`destination` list, A list containing the following elements:

- databaseTable list . A list containing the following elements:
  - schema string, The schema name for the output data.
  - table string, The table name for the output data.

`columnMapping` list, A list containing the following elements:

- address1 string, The first address line.
- address2 string, The second address line.
- city string, The city of an address.
- state string, The state of an address.
- zip string, The zip code of an address.
- name string, The full name of the resident at this address. If needed, separate multiple columns with '+', e.g. 'first_name+last_name'
- company string, The name of the company located at this address.

`useDefaultColumnMapping` boolean, Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

`performNcoa` boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.

`ncoaCredentialId` integer, Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

`outputLevel` string, The set of fields persisted by a CASS or NCOA enhancement.For CASS enhancements, one of 'cass' or 'all.'For NCOA enhancements, one of 'cass', 'ncoa' , 'coalesced' or 'all'.By default, all fields will be returned.

`limitingSQL` string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

`batchSize` integer, The maximum number of records processed at a time. Note that this parameter is not available to all users.

`archived` string, The archival status of the requested item(s).

`parentId` integer, Parent ID that triggers this enhancement.

`enhancements_patch_civis_data_match` *Update some attributes of this Civis Data Match Enhancement*

Description

Update some attributes of this Civis Data Match Enhancement

Usage

```
enhancements_patch_civis_data_match(
  id,
  name = NULL,
  schedule = NULL,
  notifications = NULL,
  input_field_mapping = NULL,
  input_table = NULL,
  match_target_id = NULL,
  output_table = NULL,
  max_matches = NULL,
  threshold = NULL,
  archived = NULL,
  parent_id = NULL
)
```

Arguments

id	integer required. The ID for the enhancement.
name	string optional. The name of the enhancement job.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully. • successEmailFromName string, Name from which success emails are sent; defaults to "Civis." • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job. • failureEmailAddresses array, Addresses to notify by e-mail when the job fails. • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.

- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`input_field_mapping`
: list optional. The field (i.e., column) mapping for the input table. See https://api.civisanalytics.com/enhanc mapping for a list of valid field types and descriptions. Each field type should be mapped to a string specifying a column name in the input table. For field types that support multiple values (e.g., the "phone" field), a list of column names can be provided (e.g., {"phone": ["home_phone", "mobile_phone"], ...}).

`input_table`
: list optional. A list containing the following elements:

- databaseName string, The Redshift database name for the table.
- schema string, The schema name for the table.
- table string, The table name.

`match_target_id`
: integer optional. The ID of the Civis Data match target. See /match_targets for IDs.

`output_table`
: list optional. A list containing the following elements:

- databaseName string, The Redshift database name for the table.
- schema string, The schema name for the table.
- table string, The table name.

`max_matches`
: integer optional. The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

`threshold`
: number optional. The score threshold (between 0 and 1). Matches below this threshold will not be returned. The default value is 0.5.

`archived`
: boolean optional. Whether the Civis Data Match Job has been archived.

`parent_id`
: integer optional. Parent ID that triggers this enhancement.

Value

A list containing the following elements:

`id`
: integer, The ID for the enhancement.

`name`
: string, The name of the enhancement job.

`type`
: string, The type of the enhancement (e.g CASS-NCOA)

`createdAt`
: string, The time this enhancement was created.

`updatedAt`
: string, The time the enhancement was last updated.

`author`
: list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`inputFieldMapping` list, The field (i.e., column) mapping for the input table. See https://api.civisanalytics.com/enhancements/f mapping for a list of valid field types and descriptions. Each field type should be mapped to a string specifying a column name in the input table. For field types that support multiple values (e.g., the "phone" field), a list of column names can be provided (e.g., {"phone": ["home_phone", "mobile_phone"], ...}).

inputTable	list, A list containing the following elements: • databaseName string, The Redshift database name for the table. • schema string, The schema name for the table. • table string, The table name.
matchTargetId	integer, The ID of the Civis Data match target. See /match_targets for IDs.
outputTable	list, A list containing the following elements: • databaseName string, The Redshift database name for the table. • schema string, The schema name for the table. • table string, The table name.
maxMatches	integer, The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.
threshold	number, The score threshold (between 0 and 1). Matches below this threshold will not be returned. The default value is 0.5.
archived	boolean, Whether the Civis Data Match Job has been archived.
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
parentId	integer, Parent ID that triggers this enhancement.

`enhancements_patch_geocode` *Update some attributes of this Geocode Enhancement*

Description

Update some attributes of this Geocode Enhancement

Usage

```
enhancements_patch_geocode(
  id,
  name = NULL,
  schedule = NULL,
  notifications = NULL,
  remote_host_id = NULL,
  credential_id = NULL,
  source_schema_and_table = NULL,
  multipart_key = NULL,
```

```
  limiting_sql = NULL,
  target_schema = NULL,
  target_table = NULL,
  country = NULL,
  provider = NULL,
  output_address = NULL,
  parent_id = NULL
)
```

Arguments

id	integer required. The ID for the enhancement.
name	string optional. The name of the enhancement job.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully. • successEmailFromName string, Name from which success emails are sent; defaults to "Civis." • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job. • failureEmailAddresses array, Addresses to notify by e-mail when the job fails. • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes. • successOn boolean, If success email notifications are on. Defaults to user's preferences. • failureOn boolean, If failure email notifications are on. Defaults to user's preferences.
remote_host_id	integer optional. The ID of the remote host.
credential_id	integer optional. The ID of the remote host credential.
source_schema_and_table	string optional. The source database schema and table.

multipart_key	array optional. The source table primary key.
limiting_sql	string optional. The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
target_schema	string optional. The output table schema.
target_table	string optional. The output table name.
country	string optional. The country of the addresses to be geocoded; either 'us' or 'ca'.
provider	string optional. The geocoding provider; one of postgis and geocoder_ca.
output_address	boolean optional. Whether to output the parsed address. Only guaranteed for the 'postgis' provider.
parent_id	integer optional. Parent ID that triggers this enhancement.

Value

A list containing the following elements:

id	integer, The ID for the enhancement.
name	string, The name of the enhancement job.
type	string, The type of the enhancement (e.g CASS-NCOA)
createdAt	string, The time this enhancement was created.
updatedAt	string, The time the enhancement was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the enhancement's last run
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list, A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown.

- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`remoteHostId` integer, The ID of the remote host.

`credentialId` integer, The ID of the remote host credential.

`sourceSchemaAndTable` string, The source database schema and table.

`multipartKey` array, The source table primary key.

`limitingSQL` string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

`targetSchema` string, The output table schema.

`targetTable` string, The output table name.

`country` string, The country of the addresses to be geocoded; either 'us' or 'ca'.

`provider` string, The geocoding provider; one of postgis and geocoder_ca.

`outputAddress` boolean, Whether to output the parsed address. Only guaranteed for the 'postgis' provider.

`archived` string, The archival status of the requested item(s).

`parentId` integer, Parent ID that triggers this enhancement.

enhancements_patch_identity_resolution
Update some attributes of this Identity Resolution Enhancement

Description

Update some attributes of this Identity Resolution Enhancement

Usage

```
enhancements_patch_identity_resolution(
  id,
  name = NULL,
  schedule = NULL,
  notifications = NULL,
  threshold = NULL,
  sources = NULL,
  match_target_id = NULL,
  enforced_links = NULL,
  customer_graph = NULL,
  golden_table = NULL,
  link_scores = NULL
)
```

Arguments

id	integer required. The ID for the enhancement.
name	string optional. The name of the enhancement job.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.

- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`threshold` number optional. A value that determines the extent to which similar records get assigned the same resolved ID. Must be within 0.5 and 1, inclusive. Defaults to 0.8 if unspecified.Higher values may result in fewer cases where records about different individuals erroneously receive the same resolved ID, but also more more cases where records about the same individual receive different resolved IDs.

`sources` array optional. An array containing the following fields:

- name string, A user-specified name for the source.
- description string, A description of the source.
- databaseName string, The name of the source database.
- schemaName string, The name of the source schema.
- tableName string, The name of the source table.
- fieldMapping object, A mapping of PII fields to columns in this table. Valid keys are primary_key, first_name, middle_name, last_name, gender, phone, email, birth_date, birth_year, birth_month, birth_day, house_number, street, unit, full_address, city, state, state_code, zip, lat, lon, and name_suffix

`match_target_id` integer optional. The ID of the Civis Data (Custom) match target. See /match_targets for IDs.

`enforced_links` array optional. An array containing the following fields:

- source1 string, Name of the first source. Must be defined in Sources list.
- source1JoinCol string, Column from the first source to join on.
- source2 string, Name of the second source. Must be defined in Sources list
- source2JoinCol string, Column from the second source to join on.

`customer_graph` list optional. A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`golden_table` list optional. A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.

- tableName string, The name of the destination table.
- fields array,

`link_scores` list optional. A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

Value

A list containing the following elements:

`id` integer, The ID for the enhancement.

`name` string, The name of the enhancement job.

`type` string, The type of the enhancement (e.g CASS-NCOA)

`createdAt` string, The time this enhancement was created.

`updatedAt` string, The time the enhancement was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.

- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`threshold` number, A value that determines the extent to which similar records get assigned the same resolved ID. Must be within 0.5 and 1, inclusive. Defaults to 0.8 if unspecified.Higher values may result in fewer cases where records about different individuals erroneously receive the same resolved ID, but also more more cases where records about the same individual receive different resolved IDs.

`sources` array, An array containing the following fields:

- name string, A user-specified name for the source.
- description string, A description of the source.
- databaseName string, The name of the source database.
- schemaName string, The name of the source schema.
- tableName string, The name of the source table.
- fieldMapping object, A mapping of PII fields to columns in this table. Valid keys are primary_key, first_name, middle_name, last_name, gender, phone, email, birth_date, birth_year, birth_month, birth_day, house_number, street, unit, full_address, city, state, state_code, zip, lat, lon, and name_suffix

`matchTargetId` integer, The ID of the Civis Data (Custom) match target. See /match_targets for IDs.

`enforcedLinks` array, An array containing the following fields:

- source1 string, Name of the first source. Must be defined in Sources list.
- source1JoinCol string, Column from the first source to join on.
- source2 string, Name of the second source. Must be defined in Sources list
- source2JoinCol string, Column from the second source to join on.

`customerGraph` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.

- tableName string, The name of the destination table.

`goldenTable` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.
- fields array,

`linkScores` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`legacyId` integer, ID of this pipeline in the legacy IDR service application.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.
- config string, How the Identity Resolution job was configured for this run.
- sampleRecordsQuery string, A SQL query to produce a sample of records to inspect.
- expandClusterQuery string, A customizable query to view PII associated with resolved ids.
- runMetrics list . A list containing the following elements:
  - numRecords integer, The number of input records for this run.
  - uniqueIds integer, The number of distinct unique IDs in the input records for this run.
  - uniqueDeduplicatedIds integer, The number of resolved IDs associated with more than one unique ID in the input.
  - maxClusterSize integer, The number of records in the largest cluster of resolved IDs.
  - avgClusterSize number, The average number of records with the same resolved ID.
  - clusterSizeFrequencies object, A mapping from numbers of records with the same resolved ID (i.e., sizes of clusters) to numbers of such clusters. For example, if there were 10 clusters with 2 records each, 2 would be a key in the mapping, and 10 would be its value.
- errorSection string, If there was a failure, this will denote which section of the Identity Resolution job failed. One of: data_preparation, compute_setup or data_processing.

enhancements_post_cass_ncoa *Create a CASS/NCOA Enhancement*

Description

Create a CASS/NCOA Enhancement

Usage

```
enhancements_post_cass_ncoa(
  name,
  source,
  schedule = NULL,
  notifications = NULL,
  destination = NULL,
  column_mapping = NULL,
  use_default_column_mapping = NULL,
  perform_ncoa = NULL,
  ncoa_credential_id = NULL,
  output_level = NULL,
  limiting_sql = NULL,
  batch_size = NULL,
  parent_id = NULL
)
```

Arguments

name	string required. The name of the enhancement job.
source	list required. A list containing the following elements: • databaseTable list . A list containing the following elements: – schema string, The schema name of the source table. – table string, The name of the source table. – remoteHostId integer, The ID of the database host for the table. – credentialId integer, The id of the credentials to be used when performing the enhancement. – multipartKey array, The source table primary key.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.

- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`destination` list optional. A list containing the following elements:

- databaseTable list . A list containing the following elements:
  - schema string, The schema name for the output data.
  - table string, The table name for the output data.

`column_mapping` list optional. A list containing the following elements:

- address1 string, The first address line.
- address2 string, The second address line.
- city string, The city of an address.
- state string, The state of an address.
- zip string, The zip code of an address.
- name string, The full name of the resident at this address. If needed, separate multiple columns with '+', e.g. 'first_name+last_name'
- company string, The name of the company located at this address.

`use_default_column_mapping`
boolean optional. Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

`perform_ncoa` boolean optional. Whether to update addresses for records matching the National Change of Address (NCOA) database.

`ncoa_credential_id`
integer optional. Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

output_level	string optional. The set of fields persisted by a CASS or NCOA enhancement.For CASS enhancements, one of 'cass' or 'all.'For NCOA enhancements, one of 'cass', 'ncoa' , 'coalesced' or 'all'.By default, all fields will be returned.
limiting_sql	string optional. The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
batch_size	integer optional. The maximum number of records processed at a time. Note that this parameter is not available to all users.
parent_id	integer optional. Parent ID that triggers this enhancement.

Value

A list containing the following elements:

id	integer, The ID for the enhancement.
name	string, The name of the enhancement job.
type	string, The type of the enhancement (e.g CASS-NCOA)
createdAt	string, The time this enhancement was created.
updatedAt	string, The time the enhancement was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the enhancement's last run
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list, A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully. • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."

- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`source` list, A list containing the following elements:

- databaseTable list . A list containing the following elements:
  - schema string, The schema name of the source table.
  - table string, The name of the source table.
  - remoteHostId integer, The ID of the database host for the table.
  - credentialId integer, The id of the credentials to be used when performing the enhancement.
  - multipartKey array, The source table primary key.

`destination` list, A list containing the following elements:

- databaseTable list . A list containing the following elements:
  - schema string, The schema name for the output data.
  - table string, The table name for the output data.

`columnMapping` list, A list containing the following elements:

- address1 string, The first address line.
- address2 string, The second address line.
- city string, The city of an address.
- state string, The state of an address.
- zip string, The zip code of an address.
- name string, The full name of the resident at this address. If needed, separate multiple columns with '+', e.g. 'first_name+last_name'
- company string, The name of the company located at this address.

`useDefaultColumnMapping`
boolean, Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

performNcoa	boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.
ncoaCredentialId	integer, Credential to use when performing NCOA updates. Required if 'performNcoa' is true.
outputLevel	string, The set of fields persisted by a CASS or NCOA enhancement.For CASS enhancements, one of 'cass' or 'all.'For NCOA enhancements, one of 'cass', 'ncoa' , 'coalesced' or 'all'.By default, all fields will be returned.
limitingSQL	string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
batchSize	integer, The maximum number of records processed at a time. Note that this parameter is not available to all users.
archived	string, The archival status of the requested item(s).
parentId	integer, Parent ID that triggers this enhancement.

`enhancements_post_cass_ncoa_cancel`
Cancel a run

Description

Cancel a run

Usage

```
enhancements_post_cass_ncoa_cancel(id)
```

Arguments

id	integer required. The ID of the job.

Value

A list containing the following elements:

id	integer, The ID of the run.
state	string, The state of the run, one of 'queued', 'running' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.

enhancements_post_cass_ncoa_runs *Start a run*

Description

Start a run

Usage

```
enhancements_post_cass_ncoa_runs(id)
```

Arguments

id	integer required. The ID of the CASS NCOA job.

Value

A list containing the following elements:

id	integer, The ID of the run.
cassNcoaId	integer, The ID of the CASS NCOA job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

enhancements_post_civis_data_match *Create a Civis Data Match Enhancement*

Description

Create a Civis Data Match Enhancement

Usage

```
enhancements_post_civis_data_match(
  name,
  input_field_mapping,
  input_table,
  match_target_id,
  output_table,
  schedule = NULL,
  notifications = NULL,
  max_matches = NULL,
  threshold = NULL,
  archived = NULL,
  parent_id = NULL
)
```

Arguments

name	string required. The name of the enhancement job.
input_field_mapping	list required. The field (i.e., column) mapping for the input table. See https://api.civisanalytics.com/enhanc mapping for a list of valid field types and descriptions. Each field type should be mapped to a string specifying a column name in the input table. For field types that support multiple values (e.g., the "phone" field), a list of column names can be provided (e.g., {"phone": ["home_phone", "mobile_phone"], ...}).
input_table	list required. A list containing the following elements: • databaseName string, The Redshift database name for the table. • schema string, The schema name for the table. • table string, The table name.
match_target_id	integer required. The ID of the Civis Data match target. See /match_targets for IDs.
output_table	list required. A list containing the following elements: • databaseName string, The Redshift database name for the table. • schema string, The schema name for the table. • table string, The table name.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`max_matches` integer optional. The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

`threshold` number optional. The score threshold (between 0 and 1). Matches below this threshold will not be returned. The default value is 0.5.

`archived` boolean optional. Whether the Civis Data Match Job has been archived.

`parent_id` integer optional. Parent ID that triggers this enhancement.

Value

A list containing the following elements:

`id` integer, The ID for the enhancement.

`name` string, The name of the enhancement job.

`type` string, The type of the enhancement (e.g CASS-NCOA)

`createdAt` string, The time this enhancement was created.

`updatedAt` string, The time the enhancement was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`inputFieldMapping` list, The field (i.e., column) mapping for the input table. See https://api.civisanalytics.com/enhancements/f mapping for a list of valid field types and descriptions. Each field type should be mapped to a string specifying a column name in the input table. For field types that support multiple values (e.g., the "phone" field), a list of column names can be provided (e.g., {"phone": ["home_phone", "mobile_phone"], ...}).

`inputTable` list, A list containing the following elements:

- databaseName string, The Redshift database name for the table.

- schema string, The schema name for the table.
- table string, The table name.

`matchTargetId` integer, The ID of the Civis Data match target. See /match_targets for IDs.

`outputTable` list, A list containing the following elements:

- databaseName string, The Redshift database name for the table.
- schema string, The schema name for the table.
- table string, The table name.

`maxMatches` integer, The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

`threshold` number, The score threshold (between 0 and 1). Matches below this threshold will not be returned. The default value is 0.5.

`archived` boolean, Whether the Civis Data Match Job has been archived.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`parentId` integer, Parent ID that triggers this enhancement.

`enhancements_post_civis_data_match_cancel`
Cancel a run

Description

Cancel a run

Usage

```
enhancements_post_civis_data_match_cancel(id)
```

Arguments

`id`	integer required. The ID of the job.

Value

A list containing the following elements:

`id` integer, The ID of the run.

`state` string, The state of the run, one of 'queued', 'running' or 'cancelled'.

`isCancelRequested` boolean, True if run cancel requested, else false.

`enhancements_post_civis_data_match_clone` *Clone this Civis Data Match Enhancement*

Description

Clone this Civis Data Match Enhancement

Usage

```
enhancements_post_civis_data_match_clone(
  id,
  clone_schedule = NULL,
  clone_triggers = NULL,
  clone_notifications = NULL
)
```

Arguments

`id`	integer required. The ID for the enhancement.
`clone_schedule`	boolean optional. If true, also copy the schedule to the new enhancement.
`clone_triggers`	boolean optional. If true, also copy the triggers to the new enhancement.
`clone_notifications`	boolean optional. If true, also copy the notifications to the new enhancement.

Value

A list containing the following elements:

`id`	integer, The ID for the enhancement.
`name`	string, The name of the enhancement job.
`type`	string, The type of the enhancement (e.g CASS-NCOA)
`createdAt`	string, The time this enhancement was created.
`updatedAt`	string, The time the enhancement was last updated.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`state`	string, The status of the enhancement's last run
`schedule`	list, A list containing the following elements: • scheduled boolean, If the item is scheduled.

- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`inputFieldMapping`
list, The field (i.e., column) mapping for the input table. See https://api.civisanalytics.com/enhancements/f mapping for a list of valid field types and descriptions. Each field type should be mapped to a string specifying a column name in the input table. For field types that support multiple values (e.g., the "phone" field), a list of column names can be provided (e.g., {"phone": ["home_phone", "mobile_phone"], ...}).

`inputTable` list, A list containing the following elements:

- databaseName string, The Redshift database name for the table.
- schema string, The schema name for the table.

- table string, The table name.

`matchTargetId` integer, The ID of the Civis Data match target. See /match_targets for IDs.

`outputTable` list, A list containing the following elements:

- databaseName string, The Redshift database name for the table.
- schema string, The schema name for the table.
- table string, The table name.

`maxMatches` integer, The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

`threshold` number, The score threshold (between 0 and 1). Matches below this threshold will not be returned. The default value is 0.5.

`archived` boolean, Whether the Civis Data Match Job has been archived.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`parentId` integer, Parent ID that triggers this enhancement.

`enhancements_post_civis_data_match_runs` *Start a run*

Description

Start a run

Usage

```
enhancements_post_civis_data_match_runs(id)
```

Arguments

`id` integer required. The ID of the Civis Data Match job.

Value

A list containing the following elements:

`id` integer, The ID of the run.

`civisDataMatchId` integer, The ID of the Civis Data Match job.

state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

---

enhancements_post_geocode *Create a Geocode Enhancement*

Description

Create a Geocode Enhancement

Usage

```
enhancements_post_geocode(
  name,
  remote_host_id,
  credential_id,
  source_schema_and_table,
  schedule = NULL,
  notifications = NULL,
  multipart_key = NULL,
  limiting_sql = NULL,
  target_schema = NULL,
  target_table = NULL,
  country = NULL,
  provider = NULL,
  output_address = NULL,
  parent_id = NULL
)
```

Arguments

name	string required. The name of the enhancement job.
remote_host_id	integer required. The ID of the remote host.
credential_id	integer required. The ID of the remote host credential.
source_schema_and_table	string required. The source database schema and table.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled.

- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`multipart_key` array optional. The source table primary key.

`limiting_sql` string optional. The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

`target_schema` string optional. The output table schema.

`target_table` string optional. The output table name.

`country` string optional. The country of the addresses to be geocoded; either 'us' or 'ca'.

`provider` string optional. The geocoding provider; one of postgis and geocoder_ca.

`output_address` boolean optional. Whether to output the parsed address. Only guaranteed for the 'postgis' provider.

`parent_id` integer optional. Parent ID that triggers this enhancement.

Value

A list containing the following elements:

`id` integer, The ID for the enhancement.

`name` string, The name of the enhancement job.

`type` string, The type of the enhancement (e.g CASS-NCOA)

`createdAt` string, The time this enhancement was created.

`updatedAt` string, The time the enhancement was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.

- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`remoteHostId`
: integer, The ID of the remote host.

`credentialId`
: integer, The ID of the remote host credential.

`sourceSchemaAndTable`
: string, The source database schema and table.

`multipartKey`
: array, The source table primary key.

`limitingSQL`
: string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

`targetSchema`
: string, The output table schema.

`targetTable`
: string, The output table name.

`country`
: string, The country of the addresses to be geocoded; either 'us' or 'ca'.

`provider`
: string, The geocoding provider; one of postgis and geocoder_ca.

`outputAddress`
: boolean, Whether to output the parsed address. Only guaranteed for the 'postgis' provider.

`archived`
: string, The archival status of the requested item(s).

`parentId`
: integer, Parent ID that triggers this enhancement.

`enhancements_post_geocode_cancel`
Cancel a run

Description

Cancel a run

Usage

```
enhancements_post_geocode_cancel(id)
```

Arguments

`id`
: integer required. The ID of the job.

Value

A list containing the following elements:

`id`
: integer, The ID of the run.

`state`
: string, The state of the run, one of 'queued', 'running' or 'cancelled'.

`isCancelRequested`
: boolean, True if run cancel requested, else false.

enhancements_post_geocode_runs
Start a run

Description

Start a run

Usage

```
enhancements_post_geocode_runs(id)
```

Arguments

id	integer required. The ID of the Geocode job.

Value

A list containing the following elements:

id	integer, The ID of the run.
geocodeId	integer, The ID of the Geocode job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

enhancements_post_identity_resolution
Create an Identity Resolution Enhancement

Description

Create an Identity Resolution Enhancement

Usage

```
enhancements_post_identity_resolution(
  name,
  sources,
  schedule = NULL,
  notifications = NULL,
  threshold = NULL,
  match_target_id = NULL,
  enforced_links = NULL,
  customer_graph = NULL,
  golden_table = NULL,
  link_scores = NULL,
  legacy_id = NULL
)
```

Arguments

`name`	string required. The name of the enhancement job.
`sources`	array required. An array containing the following fields: • name string, A user-specified name for the source. • description string, A description of the source. • databaseName string, The name of the source database. • schemaName string, The name of the source schema. • tableName string, The name of the source table. • fieldMapping object, A mapping of PII fields to columns in this table. Valid keys are primary_key, first_name, middle_name, last_name, gender, phone, email, birth_date, birth_year, birth_month, birth_day, house_number, street, unit, full_address, city, state, state_code, zip, lat, lon, and name_suffix
`schedule`	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
`notifications`	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.

- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`threshold` number optional. A value that determines the extent to which similar records get assigned the same resolved ID. Must be within 0.5 and 1, inclusive. Defaults to 0.8 if unspecified.Higher values may result in fewer cases where records about different individuals erroneously receive the same resolved ID, but also more more cases where records about the same individual receive different resolved IDs.

`match_target_id` integer optional. The ID of the Civis Data (Custom) match target. See /match_targets for IDs.

`enforced_links` array optional. An array containing the following fields:

- source1 string, Name of the first source. Must be defined in Sources list.
- source1JoinCol string, Column from the first source to join on.
- source2 string, Name of the second source. Must be defined in Sources list
- source2JoinCol string, Column from the second source to join on.

`customer_graph` list optional. A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`golden_table` list optional. A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.
- fields array,

`link_scores` list optional. A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`legacy_id` integer optional. ID of this pipeline in the legacy IDR service application.

Value

A list containing the following elements:

`id` integer, The ID for the enhancement.

`name` string, The name of the enhancement job.

`type` string, The type of the enhancement (e.g CASS-NCOA)

`createdAt` string, The time this enhancement was created.

`updatedAt` string, The time the enhancement was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`threshold` number, A value that determines the extent to which similar records get assigned the same resolved ID. Must be within 0.5 and 1, inclusive. Defaults to 0.8 if unspecified.Higher values may result in fewer cases where records about different individuals erroneously receive the same resolved ID, but also more more cases where records about the same individual receive different resolved IDs.

`sources` array, An array containing the following fields:

- name string, A user-specified name for the source.
- description string, A description of the source.
- databaseName string, The name of the source database.
- schemaName string, The name of the source schema.
- tableName string, The name of the source table.
- fieldMapping object, A mapping of PII fields to columns in this table. Valid keys are primary_key, first_name, middle_name, last_name, gender, phone, email, birth_date, birth_year, birth_month, birth_day, house_number, street, unit, full_address, city, state, state_code, zip, lat, lon, and name_suffix

`matchTargetId` integer, The ID of the Civis Data (Custom) match target. See /match_targets for IDs.

`enforcedLinks` array, An array containing the following fields:

- source1 string, Name of the first source. Must be defined in Sources list.
- source1JoinCol string, Column from the first source to join on.
- source2 string, Name of the second source. Must be defined in Sources list
- source2JoinCol string, Column from the second source to join on.

`customerGraph` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`goldenTable` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.
- fields array,

`linkScores` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`legacyId` integer, ID of this pipeline in the legacy IDR service application.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.
- config string, How the Identity Resolution job was configured for this run.
- sampleRecordsQuery string, A SQL query to produce a sample of records to inspect.
- expandClusterQuery string, A customizable query to view PII associated with resolved ids.
- runMetrics list . A list containing the following elements:
  - numRecords integer, The number of input records for this run.
  - uniqueIds integer, The number of distinct unique IDs in the input records for this run.
  - uniqueDeduplicatedIds integer, The number of resolved IDs associated with more than one unique ID in the input.
  - maxClusterSize integer, The number of records in the largest cluster of resolved IDs.
  - avgClusterSize number, The average number of records with the same resolved ID.
  - clusterSizeFrequencies object, A mapping from numbers of records with the same resolved ID (i.e., sizes of clusters) to numbers of such clusters. For example, if there were 10 clusters with 2 records each, 2 would be a key in the mapping, and 10 would be its value.
- errorSection string, If there was a failure, this will denote which section of the Identity Resolution job failed. One of: data_preparation, compute_setup or data_processing.

`enhancements_post_identity_resolution_cancel`
Cancel a run

Description

Cancel a run

Usage

```
enhancements_post_identity_resolution_cancel(id)
```

Arguments

id	integer required. The ID of the job.

Value

A list containing the following elements:

id	integer, The ID of the run.
state	string, The state of the run, one of 'queued', 'running' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.

---

enhancements_post_identity_resolution_clone
Clone this Identity Resolution Enhancement

Description

Clone this Identity Resolution Enhancement

Usage

```
enhancements_post_identity_resolution_clone(
  id,
  clone_schedule = NULL,
  clone_triggers = NULL,
  clone_notifications = NULL
)
```

Arguments

id	integer required. The ID for the enhancement.
clone_schedule	boolean optional. If true, also copy the schedule to the new enhancement.
clone_triggers	boolean optional. If true, also copy the triggers to the new enhancement.
clone_notifications	boolean optional. If true, also copy the notifications to the new enhancement.

Value

A list containing the following elements:

`id` integer, The ID for the enhancement.

`name` string, The name of the enhancement job.

`type` string, The type of the enhancement (e.g CASS-NCOA)

`createdAt` string, The time this enhancement was created.

`updatedAt` string, The time the enhancement was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`threshold` number, A value that determines the extent to which similar records get assigned the same resolved ID. Must be within 0.5 and 1, inclusive. Defaults to 0.8 if unspecified.Higher values may result in fewer cases where records about different individuals erroneously receive the same resolved ID, but also more more cases where records about the same individual receive different resolved IDs.

`sources` array, An array containing the following fields:

- name string, A user-specified name for the source.
- description string, A description of the source.
- databaseName string, The name of the source database.
- schemaName string, The name of the source schema.
- tableName string, The name of the source table.
- fieldMapping object, A mapping of PII fields to columns in this table. Valid keys are primary_key, first_name, middle_name, last_name, gender, phone, email, birth_date, birth_year, birth_month, birth_day, house_number, street, unit, full_address, city, state, state_code, zip, lat, lon, and name_suffix

`matchTargetId` integer, The ID of the Civis Data (Custom) match target. See /match_targets for IDs.

`enforcedLinks` array, An array containing the following fields:

- source1 string, Name of the first source. Must be defined in Sources list.
- source1JoinCol string, Column from the first source to join on.
- source2 string, Name of the second source. Must be defined in Sources list
- source2JoinCol string, Column from the second source to join on.

`customerGraph` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`goldenTable` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.
- fields array,

`linkScores` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`legacyId` integer, ID of this pipeline in the legacy IDR service application.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.
- config string, How the Identity Resolution job was configured for this run.
- sampleRecordsQuery string, A SQL query to produce a sample of records to inspect.
- expandClusterQuery string, A customizable query to view PII associated with resolved ids.
- runMetrics list . A list containing the following elements:
  - numRecords integer, The number of input records for this run.
  - uniqueIds integer, The number of distinct unique IDs in the input records for this run.
  - uniqueDeduplicatedIds integer, The number of resolved IDs associated with more than one unique ID in the input.
  - maxClusterSize integer, The number of records in the largest cluster of resolved IDs.
  - avgClusterSize number, The average number of records with the same resolved ID.
  - clusterSizeFrequencies object, A mapping from numbers of records with the same resolved ID (i.e., sizes of clusters) to numbers of such clusters. For example, if there were 10 clusters with 2 records each, 2 would be a key in the mapping, and 10 would be its value.
- errorSection string, If there was a failure, this will denote which section of the Identity Resolution job failed. One of: data_preparation, compute_setup or data_processing.

`enhancements_post_identity_resolution_runs`
Start a run

Description

Start a run

Usage

```
enhancements_post_identity_resolution_runs(id)
```

Arguments

id	integer required. The ID of the Identity Resolution job.

Value

A list containing the following elements:

id	integer, The ID of the run.
identityResolutionId	integer, The ID of the Identity Resolution job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
config	string, How the Identity Resolution job was configured for this run.
sampleRecordsQuery	string, A SQL query to produce a sample of records to inspect.
expandClusterQuery	string, A customizable query to view PII associated with resolved ids.
runMetrics	list, A list containing the following elements: • numRecords integer, The number of input records for this run. • uniqueIds integer, The number of distinct unique IDs in the input records for this run. • uniqueDeduplicatedIds integer, The number of resolved IDs associated with more than one unique ID in the input. • maxClusterSize integer, The number of records in the largest cluster of resolved IDs. • avgClusterSize number, The average number of records with the same resolved ID. • clusterSizeFrequencies object, A mapping from numbers of records with the same resolved ID (i.e., sizes of clusters) to numbers of such clusters. For example, if there were 10 clusters with 2 records each, 2 would be a key in the mapping, and 10 would be its value.
errorSection	string, If there was a failure, this will denote which section of the Identity Resolution job failed. One of: data_preparation, compute_setup or data_processing.

enhancements_put_cass_ncoa *Replace all attributes of this CASS/NCOA Enhancement*

Description

Replace all attributes of this CASS/NCOA Enhancement

Usage

```
enhancements_put_cass_ncoa(
  id,
  name,
  source,
  schedule = NULL,
  notifications = NULL,
  destination = NULL,
  column_mapping = NULL,
  use_default_column_mapping = NULL,
  perform_ncoa = NULL,
  ncoa_credential_id = NULL,
  output_level = NULL,
  limiting_sql = NULL,
  batch_size = NULL,
  parent_id = NULL
)
```

Arguments

id	integer required. The ID for the enhancement.
name	string required. The name of the enhancement job.
source	list required. A list containing the following elements: • databaseTable list . A list containing the following elements: – schema string, The schema name of the source table. – table string, The name of the source table. – remoteHostId integer, The ID of the database host for the table. – credentialId integer, The id of the credentials to be used when performing the enhancement. – multipartKey array, The source table primary key.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on.

- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`destination` list optional. A list containing the following elements:

- databaseTable list . A list containing the following elements:
  - schema string, The schema name for the output data.
  - table string, The table name for the output data.

`column_mapping` list optional. A list containing the following elements:

- address1 string, The first address line.
- address2 string, The second address line.
- city string, The city of an address.
- state string, The state of an address.
- zip string, The zip code of an address.
- name string, The full name of the resident at this address. If needed, separate multiple columns with '+', e.g. 'first_name+last_name'
- company string, The name of the company located at this address.

`use_default_column_mapping`
boolean optional. Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

`perform_ncoa` boolean optional. Whether to update addresses for records matching the National Change of Address (NCOA) database.

`ncoa_credential_id`
integer optional. Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

output_level	string optional. The set of fields persisted by a CASS or NCOA enhancement.For CASS enhancements, one of 'cass' or 'all.'For NCOA enhancements, one of 'cass', 'ncoa' , 'coalesced' or 'all'.By default, all fields will be returned.
limiting_sql	string optional. The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
batch_size	integer optional. The maximum number of records processed at a time. Note that this parameter is not available to all users.
parent_id	integer optional. Parent ID that triggers this enhancement.

Value

A list containing the following elements:

id	integer, The ID for the enhancement.
name	string, The name of the enhancement job.
type	string, The type of the enhancement (e.g CASS-NCOA)
createdAt	string, The time this enhancement was created.
updatedAt	string, The time the enhancement was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the enhancement's last run
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list, A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully. • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."

- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`source` list, A list containing the following elements:

- databaseTable list . A list containing the following elements:
  - schema string, The schema name of the source table.
  - table string, The name of the source table.
  - remoteHostId integer, The ID of the database host for the table.
  - credentialId integer, The id of the credentials to be used when performing the enhancement.
  - multipartKey array, The source table primary key.

`destination` list, A list containing the following elements:

- databaseTable list . A list containing the following elements:
  - schema string, The schema name for the output data.
  - table string, The table name for the output data.

`columnMapping` list, A list containing the following elements:

- address1 string, The first address line.
- address2 string, The second address line.
- city string, The city of an address.
- state string, The state of an address.
- zip string, The zip code of an address.
- name string, The full name of the resident at this address. If needed, separate multiple columns with '+', e.g. 'first_name+last_name'
- company string, The name of the company located at this address.

`useDefaultColumnMapping`
boolean, Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

performNcoa	boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.
ncoaCredentialId	integer, Credential to use when performing NCOA updates. Required if 'performNcoa' is true.
outputLevel	string, The set of fields persisted by a CASS or NCOA enhancement.For CASS enhancements, one of 'cass' or 'all.'For NCOA enhancements, one of 'cass', 'ncoa' , 'coalesced' or 'all'.By default, all fields will be returned.
limitingSQL	string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
batchSize	integer, The maximum number of records processed at a time. Note that this parameter is not available to all users.
archived	string, The archival status of the requested item(s).
parentId	integer, Parent ID that triggers this enhancement.

`enhancements_put_cass_ncoa_archive`

Update the archive status of this object

Description

Update the archive status of this object

Usage

```
enhancements_put_cass_ncoa_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ID for the enhancement.
name	string, The name of the enhancement job.
type	string, The type of the enhancement (e.g CASS-NCOA)
createdAt	string, The time this enhancement was created.
updatedAt	string, The time the enhancement was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name.

- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`source` list, A list containing the following elements:

- databaseTable list . A list containing the following elements:

  - schema string, The schema name of the source table.
  - table string, The name of the source table.
  - remoteHostId integer, The ID of the database host for the table.
  - credentialId integer, The id of the credentials to be used when performing the enhancement.
  - multipartKey array, The source table primary key.

`destination` list, A list containing the following elements:

- databaseTable list . A list containing the following elements:
  - schema string, The schema name for the output data.
  - table string, The table name for the output data.

`columnMapping` list, A list containing the following elements:

- address1 string, The first address line.
- address2 string, The second address line.
- city string, The city of an address.
- state string, The state of an address.
- zip string, The zip code of an address.
- name string, The full name of the resident at this address. If needed, separate multiple columns with '+', e.g. 'first_name+last_name'
- company string, The name of the company located at this address.

`useDefaultColumnMapping`
boolean, Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

`performNcoa` boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.

`ncoaCredentialId`
integer, Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

`outputLevel` string, The set of fields persisted by a CASS or NCOA enhancement.For CASS enhancements, one of 'cass' or 'all.'For NCOA enhancements, one of 'cass', 'ncoa' , 'coalesced' or 'all'.By default, all fields will be returned.

`limitingSQL` string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

`batchSize` integer, The maximum number of records processed at a time. Note that this parameter is not available to all users.

`archived` string, The archival status of the requested item(s).

`parentId` integer, Parent ID that triggers this enhancement.

enhancements_put_cass_ncoa_projects

Add a CASS/NCOA Enhancement to a project

Description

Add a CASS/NCOA Enhancement to a project

Usage

```
enhancements_put_cass_ncoa_projects(id, project_id)
```

Arguments

id	integer required. The ID of the CASS/NCOA Enhancement.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

enhancements_put_cass_ncoa_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
enhancements_put_cass_ncoa_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
`totalGroupShares`	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`enhancements_put_cass_ncoa_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
enhancements_put_cass_ncoa_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_ids`	array required. An array of one or more user IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
`totalGroupShares`	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`enhancements_put_cass_ncoa_transfer` *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
enhancements_put_cass_ncoa_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. ID of target user
include_dependencies	boolean required. Whether or not to give manage permissions on all dependencies
email_body	string optional. Custom body text for e-mail sent on transfer.
send_email	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies — array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

enhancements_put_civis_data_match

Replace all attributes of this Civis Data Match Enhancement

Description

Replace all attributes of this Civis Data Match Enhancement

Usage

```
enhancements_put_civis_data_match(
  id,
  name,
  input_field_mapping,
  input_table,
  match_target_id,
  output_table,
  schedule = NULL,
```

```
  notifications = NULL,
  max_matches = NULL,
  threshold = NULL,
  archived = NULL,
  parent_id = NULL
)
```

Arguments

id	integer required. The ID for the enhancement.
name	string required. The name of the enhancement job.
input_field_mapping	list required. The field (i.e., column) mapping for the input table. See https://api.civisanalytics.com/enhanc mapping for a list of valid field types and descriptions. Each field type should be mapped to a string specifying a column name in the input table. For field types that support multiple values (e.g., the "phone" field), a list of column names can be provided (e.g., {"phone": ["home_phone", "mobile_phone"], ...}).
input_table	list required. A list containing the following elements: • databaseName string, The Redshift database name for the table. • schema string, The schema name for the table. • table string, The table name.
match_target_id	integer required. The ID of the Civis Data match target. See /match_targets for IDs.
output_table	list required. A list containing the following elements: • databaseName string, The Redshift database name for the table. • schema string, The schema name for the table. • table string, The table name.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.

- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`max_matches` integer optional. The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

`threshold` number optional. The score threshold (between 0 and 1). Matches below this threshold will not be returned. The default value is 0.5.

`archived` boolean optional. Whether the Civis Data Match Job has been archived.

`parent_id` integer optional. Parent ID that triggers this enhancement.

Value

A list containing the following elements:

`id` integer, The ID for the enhancement.

`name` string, The name of the enhancement job.

`type` string, The type of the enhancement (e.g CASS-NCOA)

`createdAt` string, The time this enhancement was created.

`updatedAt` string, The time the enhancement was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.

- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`inputFieldMapping` list, The field (i.e., column) mapping for the input table. See https://api.civisanalytics.com/enhancements/f mapping for a list of valid field types and descriptions. Each field type should be mapped to a string specifying a column name in the input table. For field types that support multiple values (e.g., the "phone" field), a list of column names can be provided (e.g., {"phone": ["home_phone", "mobile_phone"], ...}).

`inputTable` list, A list containing the following elements:

- databaseName string, The Redshift database name for the table.
- schema string, The schema name for the table.
- table string, The table name.

`matchTargetId` integer, The ID of the Civis Data match target. See /match_targets for IDs.

`outputTable` list, A list containing the following elements:

- databaseName string, The Redshift database name for the table.
- schema string, The schema name for the table.

- table string, The table name.

`maxMatches` integer, The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

`threshold` number, The score threshold (between 0 and 1). Matches below this threshold will not be returned. The default value is 0.5.

`archived` boolean, Whether the Civis Data Match Job has been archived.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`parentId` integer, Parent ID that triggers this enhancement.

`enhancements_put_civis_data_match_archive`
Update the archive status of this object

Description

Update the archive status of this object

Usage

```
enhancements_put_civis_data_match_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id` integer, The ID for the enhancement.

`name` string, The name of the enhancement job.

`type` string, The type of the enhancement (e.g CASS-NCOA)

`createdAt` string, The time this enhancement was created.

`updatedAt` string, The time the enhancement was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.

- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`inputFieldMapping`
: list, The field (i.e., column) mapping for the input table. See https://api.civisanalytics.com/enhancements/f mapping for a list of valid field types and descriptions. Each field type should be mapped to a string specifying a column name in the input table. For field types that support multiple values (e.g., the "phone" field), a list of column names can be provided (e.g., {"phone": ["home_phone", "mobile_phone"], ...}).

`inputTable`
: list, A list containing the following elements:

- databaseName string, The Redshift database name for the table.
- schema string, The schema name for the table.
- table string, The table name.

`matchTargetId`
: integer, The ID of the Civis Data match target. See /match_targets for IDs.

`outputTable`
: list, A list containing the following elements:

- databaseName string, The Redshift database name for the table.
- schema string, The schema name for the table.
- table string, The table name.

`maxMatches`
: integer, The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

`threshold`
: number, The score threshold (between 0 and 1). Matches below this threshold will not be returned. The default value is 0.5.

`archived`
: boolean, Whether the Civis Data Match Job has been archived.

`lastRun`
: list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`parentId`
: integer, Parent ID that triggers this enhancement.

`enhancements_put_civis_data_match_projects`
Add a Civis Data Match Enhancement to a project

Description

Add a Civis Data Match Enhancement to a project

Usage

```
enhancements_put_civis_data_match_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Civis Data Match Enhancement.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

---

enhancements_put_civis_data_match_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
enhancements_put_civis_data_match_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares`
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`enhancements_put_civis_data_match_shares_users`
Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

```
enhancements_put_civis_data_match_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id` integer required. The ID of the resource that is shared.

`user_ids` array required. An array of one or more user IDs.

`permission_level`
string required. Options are: "read", "write", or "manage".

`share_email_body`
string optional. Custom body text for e-mail sent on a share.

`send_shared_email`
boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- `readers` list, A list containing the following elements:
  - users array,
  - groups array,
- `writers` list, A list containing the following elements:
  - users array,
  - groups array,
- `owners` list, A list containing the following elements:
  - users array,
  - groups array,
- `totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- `totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`enhancements_put_civis_data_match_transfer` *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
enhancements_put_civis_data_match_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies` array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`enhancements_put_geocode`
Replace all attributes of this Geocode Enhancement

Description

Replace all attributes of this Geocode Enhancement

Usage

```
enhancements_put_geocode(
  id,
  name,
  remote_host_id,
  credential_id,
  source_schema_and_table,
  schedule = NULL,
  notifications = NULL,
  multipart_key = NULL,
  limiting_sql = NULL,
  target_schema = NULL,
  target_table = NULL,
  country = NULL,
  provider = NULL,
  output_address = NULL,
  parent_id = NULL
)
```

Arguments

id	integer required. The ID for the enhancement.
name	string required. The name of the enhancement job.
remote_host_id	integer required. The ID of the remote host.
credential_id	integer required. The ID of the remote host credential.
source_schema_and_table	string required. The source database schema and table.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully. • successEmailFromName string, Name from which success emails are sent; defaults to "Civis." • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job. • failureEmailAddresses array, Addresses to notify by e-mail when the job fails. • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes. • successOn boolean, If success email notifications are on. Defaults to user's preferences. • failureOn boolean, If failure email notifications are on. Defaults to user's preferences.
multipart_key	array optional. The source table primary key.
limiting_sql	string optional. The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
target_schema	string optional. The output table schema.
target_table	string optional. The output table name.
country	string optional. The country of the addresses to be geocoded; either 'us' or 'ca'.
provider	string optional. The geocoding provider; one of postgis and geocoder_ca.

`output_address` boolean optional. Whether to output the parsed address. Only guaranteed for the 'postgis' provider.

`parent_id` integer optional. Parent ID that triggers this enhancement.

Value

A list containing the following elements:

`id` integer, The ID for the enhancement.

`name` string, The name of the enhancement job.

`type` string, The type of the enhancement (e.g CASS-NCOA)

`createdAt` string, The time this enhancement was created.

`updatedAt` string, The time the enhancement was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.

- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`remoteHostId` integer, The ID of the remote host.

`credentialId` integer, The ID of the remote host credential.

`sourceSchemaAndTable` string, The source database schema and table.

`multipartKey` array, The source table primary key.

`limitingSQL` string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

`targetSchema` string, The output table schema.

`targetTable` string, The output table name.

`country` string, The country of the addresses to be geocoded; either 'us' or 'ca'.

`provider` string, The geocoding provider; one of postgis and geocoder_ca.

`outputAddress` boolean, Whether to output the parsed address. Only guaranteed for the 'postgis' provider.

`archived` string, The archival status of the requested item(s).

`parentId` integer, Parent ID that triggers this enhancement.

`enhancements_put_geocode_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
enhancements_put_geocode_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ID for the enhancement.
name	string, The name of the enhancement job.
type	string, The type of the enhancement (e.g CASS-NCOA)
createdAt	string, The time this enhancement was created.
updatedAt	string, The time the enhancement was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the enhancement's last run
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list, A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully. • successEmailFromName string, Name from which success emails are sent; defaults to "Civis." • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job. • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.

- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`remoteHostId` integer, The ID of the remote host.

`credentialId` integer, The ID of the remote host credential.

`sourceSchemaAndTable` string, The source database schema and table.

`multipartKey` array, The source table primary key.

`limitingSQL` string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

`targetSchema` string, The output table schema.

`targetTable` string, The output table name.

`country` string, The country of the addresses to be geocoded; either 'us' or 'ca'.

`provider` string, The geocoding provider; one of postgis and geocoder_ca.

`outputAddress` boolean, Whether to output the parsed address. Only guaranteed for the 'postgis' provider.

`archived` string, The archival status of the requested item(s).

`parentId` integer, Parent ID that triggers this enhancement.

`enhancements_put_geocode_projects` *Add a Geocode Enhancement to a project*

Description

Add a Geocode Enhancement to a project

Usage

```
enhancements_put_geocode_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Geocode Enhancement.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

---

enhancements_put_geocode_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
enhancements_put_geocode_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares`
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`enhancements_put_geocode_shares_users`
Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

```
enhancements_put_geocode_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id` integer required. The ID of the resource that is shared.

`user_ids` array required. An array of one or more user IDs.

`permission_level`
string required. Options are: "read", "write", or "manage".

`share_email_body`
string optional. Custom body text for e-mail sent on a share.

`send_shared_email`
boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares`
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`enhancements_put_geocode_transfer`
Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
enhancements_put_geocode_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies` array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`enhancements_put_identity_resolution`
Replace all attributes of this Identity Resolution Enhancement

---

Description

Replace all attributes of this Identity Resolution Enhancement

Usage

```
enhancements_put_identity_resolution(
  id,
  name,
  sources,
  schedule = NULL,
  notifications = NULL,
  threshold = NULL,
  match_target_id = NULL,
  enforced_links = NULL,
  customer_graph = NULL,
  golden_table = NULL,
  link_scores = NULL
)
```

Arguments

`id`	integer required. The ID for the enhancement.
`name`	string required. The name of the enhancement job.
`sources`	array required. An array containing the following fields:

- name string, A user-specified name for the source.
- description string, A description of the source.
- databaseName string, The name of the source database.
- schemaName string, The name of the source schema.
- tableName string, The name of the source table.
- fieldMapping object, A mapping of PII fields to columns in this table. Valid keys are primary_key, first_name, middle_name, last_name, gender, phone, email, birth_date, birth_year, birth_month, birth_day, house_number, street, unit, full_address, city, state, state_code, zip, lat, lon, and name_suffix

`schedule` list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`threshold` number optional. A value that determines the extent to which similar records get assigned the same resolved ID. Must be within 0.5 and 1, inclusive. Defaults to 0.8 if unspecified.Higher values may result in fewer cases where records about different individuals erroneously receive the same resolved ID, but also more more cases where records about the same individual receive different resolved IDs.

`match_target_id`
: integer optional. The ID of the Civis Data (Custom) match target. See /match_targets for IDs.

`enforced_links`
: array optional. An array containing the following fields:
  - source1 string, Name of the first source. Must be defined in Sources list.
  - source1JoinCol string, Column from the first source to join on.
  - source2 string, Name of the second source. Must be defined in Sources list
  - source2JoinCol string, Column from the second source to join on.

`customer_graph`
: list optional. A list containing the following elements:
  - databaseName string, The name of the destination database.
  - schemaName string, The name of the destination schema.
  - tableName string, The name of the destination table.

`golden_table`
: list optional. A list containing the following elements:
  - databaseName string, The name of the destination database.
  - schemaName string, The name of the destination schema.
  - tableName string, The name of the destination table.
  - fields array,

`link_scores`
: list optional. A list containing the following elements:
  - databaseName string, The name of the destination database.
  - schemaName string, The name of the destination schema.
  - tableName string, The name of the destination table.

Value

A list containing the following elements:

`id`
: integer, The ID for the enhancement.

`name`
: string, The name of the enhancement job.

`type`
: string, The type of the enhancement (e.g CASS-NCOA)

`createdAt`
: string, The time this enhancement was created.

`updatedAt`
: string, The time the enhancement was last updated.

`author`
: list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

`state`
: string, The status of the enhancement's last run

`schedule`
: list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth

- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`threshold` number, A value that determines the extent to which similar records get assigned the same resolved ID. Must be within 0.5 and 1, inclusive. Defaults to 0.8 if unspecified.Higher values may result in fewer cases where records about different individuals erroneously receive the same resolved ID, but also more more cases where records about the same individual receive different resolved IDs.

`sources` array, An array containing the following fields:

- name string, A user-specified name for the source.
- description string, A description of the source.
- databaseName string, The name of the source database.
- schemaName string, The name of the source schema.

- tableName string, The name of the source table.
- fieldMapping object, A mapping of PII fields to columns in this table. Valid keys are primary_key, first_name, middle_name, last_name, gender, phone, email, birth_date, birth_year, birth_month, birth_day, house_number, street, unit, full_address, city, state, state_code, zip, lat, lon, and name_suffix

`matchTargetId` integer, The ID of the Civis Data (Custom) match target. See /match_targets for IDs.

`enforcedLinks` array, An array containing the following fields:

- source1 string, Name of the first source. Must be defined in Sources list.
- source1JoinCol string, Column from the first source to join on.
- source2 string, Name of the second source. Must be defined in Sources list
- source2JoinCol string, Column from the second source to join on.

`customerGraph` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`goldenTable` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.
- fields array,

`linkScores` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`legacyId` integer, ID of this pipeline in the legacy IDR service application.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.
- config string, How the Identity Resolution job was configured for this run.
- sampleRecordsQuery string, A SQL query to produce a sample of records to inspect.
- expandClusterQuery string, A customizable query to view PII associated with resolved ids.
- runMetrics list . A list containing the following elements:
  - numRecords integer, The number of input records for this run.
  - uniqueIds integer, The number of distinct unique IDs in the input records for this run.

  - uniqueDeduplicatedIds integer, The number of resolved IDs associated with more than one unique ID in the input.
  - maxClusterSize integer, The number of records in the largest cluster of resolved IDs.
  - avgClusterSize number, The average number of records with the same resolved ID.
  - clusterSizeFrequencies object, A mapping from numbers of records with the same resolved ID (i.e., sizes of clusters) to numbers of such clusters. For example, if there were 10 clusters with 2 records each, 2 would be a key in the mapping, and 10 would be its value.
- errorSection string, If there was a failure, this will denote which section of the Identity Resolution job failed. One of: data_preparation, compute_setup or data_processing.

`enhancements_put_identity_resolution_archive`
Update the archive status of this object

Description

Update the archive status of this object

Usage

```
enhancements_put_identity_resolution_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id`	integer, The ID for the enhancement.
`name`	string, The name of the enhancement job.
`type`	string, The type of the enhancement (e.g CASS-NCOA)
`createdAt`	string, The time this enhancement was created.
`updatedAt`	string, The time the enhancement was last updated.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username.

- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the enhancement's last run

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`threshold` number, A value that determines the extent to which similar records get assigned the same resolved ID. Must be within 0.5 and 1, inclusive. Defaults to 0.8 if unspecified.Higher values may result in fewer cases where records about different individuals erroneously receive the same resolved ID, but also more more cases where records about the same individual receive different resolved IDs.

`sources` array, An array containing the following fields:

- name string, A user-specified name for the source.
- description string, A description of the source.
- databaseName string, The name of the source database.
- schemaName string, The name of the source schema.
- tableName string, The name of the source table.
- fieldMapping object, A mapping of PII fields to columns in this table. Valid keys are primary_key, first_name, middle_name, last_name, gender, phone, email, birth_date, birth_year, birth_month, birth_day, house_number, street, unit, full_address, city, state, state_code, zip, lat, lon, and name_suffix

`matchTargetId` integer, The ID of the Civis Data (Custom) match target. See /match_targets for IDs.

`enforcedLinks` array, An array containing the following fields:

- source1 string, Name of the first source. Must be defined in Sources list.
- source1JoinCol string, Column from the first source to join on.
- source2 string, Name of the second source. Must be defined in Sources list
- source2JoinCol string, Column from the second source to join on.

`customerGraph` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`goldenTable` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.
- fields array,

`linkScores` list, A list containing the following elements:

- databaseName string, The name of the destination database.
- schemaName string, The name of the destination schema.
- tableName string, The name of the destination table.

`legacyId` integer, ID of this pipeline in the legacy IDR service application.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.
- config string, How the Identity Resolution job was configured for this run.
- sampleRecordsQuery string, A SQL query to produce a sample of records to inspect.

- expandClusterQuery string, A customizable query to view PII associated with resolved ids.
- runMetrics list . A list containing the following elements:
  - numRecords integer, The number of input records for this run.
  - uniqueIds integer, The number of distinct unique IDs in the input records for this run.
  - uniqueDeduplicatedIds integer, The number of resolved IDs associated with more than one unique ID in the input.
  - maxClusterSize integer, The number of records in the largest cluster of resolved IDs.
  - avgClusterSize number, The average number of records with the same resolved ID.
  - clusterSizeFrequencies object, A mapping from numbers of records with the same resolved ID (i.e., sizes of clusters) to numbers of such clusters. For example, if there were 10 clusters with 2 records each, 2 would be a key in the mapping, and 10 would be its value.
- errorSection string, If there was a failure, this will denote which section of the Identity Resolution job failed. One of: data_preparation, compute_setup or data_processing.

`enhancements_put_identity_resolution_projects`
Add an Identity Resolution Enhancement to a project

Description

Add an Identity Resolution Enhancement to a project

Usage

```
enhancements_put_identity_resolution_projects(id, project_id)
```

Arguments

`id`	integer required. The ID of the Identity Resolution Enhancement.
`project_id`	integer required. The ID of the project.

Value

An empty HTTP response

`enhancements_put_identity_resolution_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
enhancements_put_identity_resolution_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`enhancements_put_identity_resolution_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
enhancements_put_identity_resolution_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_ids`	array required. An array of one or more user IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`enhancements_put_identity_resolution_transfer`
Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
enhancements_put_identity_resolution_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies`
: array, An array containing the following fields:
  - objectType string, Dependent object type
  - fcoType string, Human readable dependent object type
  - id integer, Dependent object ID
  - name string, Dependent object name, or nil if the requesting user cannot read this object

- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`exports_delete_files_csv_runs` *Cancel a run*

Description

Cancel a run

Usage

```
exports_delete_files_csv_runs(id, run_id)
```

Arguments

`id`	integer required. The ID of the CSV Export job.
`run_id`	integer required. The ID of the run.

Value

An empty HTTP response

`exports_get_files_csv` *Get a CSV Export*

Description

Get a CSV Export

Usage

```
exports_get_files_csv(id)
```

Arguments

`id`	integer required.

Value

A list containing the following elements:

`id` integer, The ID of this Csv Export job.

`name` string, The name of this Csv Export job.

`source` list, A list containing the following elements:

- sql string, The SQL query for this Csv Export job
- remoteHostId integer, The ID of the destination database host.
- credentialId integer, The ID of the credentials for the destination database.

`destination` list, A list containing the following elements:

- filenamePrefix string, The prefix of the name of the file returned to the user.
- storagePath list . A list containing the following elements:
  - filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"
  - storageHostId integer, The ID of the destination storage host.
  - credentialId integer, The ID of the credentials for the destination storage host.
  - existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path.One of: fail, append, overwrite. Default: fail. If "append" is specified,the new file will always be added to the provided path. If "overwrite" is specifiedall existing files at the provided path will be deleted and the new file will be added.By default, or if "fail" is specified, the export will fail if a file exists at the provided path.

`includeHeader` boolean, A boolean value indicating whether or not the header should be included. Defaults to true.

`compression` string, The compression of the output file. Valid arguments are "gzip" and "none". Defaults to "gzip".

`columnDelimiter` string, The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`hidden` boolean, A boolean value indicating whether or not this request should be hidden. Defaults to false.

`forceMultifile` boolean, Whether or not the csv should be split into multiple files. Default: false

`maxFileSize` integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

exports_get_files_csv_runs *Check status of a run*

Description

Check status of a run

Usage

```
exports_get_files_csv_runs(id, run_id)
```

Arguments

id	integer required. The ID of the CSV Export job.
run_id	integer required. The ID of the run.

Value

A list containing the following elements:

id	integer,
state	string,
createdAt	string, The time that the run was queued.
startedAt	string, The time that the run started.
finishedAt	string, The time that the run completed.
error	string, The error message for this run, if present.
outputCachedOn	string, The time that the output was originally exported, if a cache entry was used by the run.

exports_list *List*

Description

List

Usage

```
exports_list(
  type = NULL,
  status = NULL,
  author = NULL,
  hidden = NULL,
  archived = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

type	string optional. If specified, return exports of these types. It accepts a comma-separated list, possible values are 'database' and 'gdoc'.
status	string optional. If specified, returns export with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'failed', 'succeeded', 'idle', 'scheduled'.
author	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
archived	string optional. The archival status of the requested item(s).
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at, last_run.updated_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID for this export.
name	string, The name of this export.
type	string, The type of export.
createdAt	string, The creation time for this export.
updatedAt	string, The last modification time for this export.
state	string,
lastRun	list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`exports_list_files_csv_runs`
List runs for the given CSV Export job

Description

List runs for the given CSV Export job

Usage

```
exports_list_files_csv_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

`id`	integer required. The ID of the CSV Export job.
`limit`	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
`page_num`	integer optional. Page number of the results to return. Defaults to the first page, 1.
`order`	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
`order_dir`	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer,
state	string,
createdAt	string, The time that the run was queued.
startedAt	string, The time that the run started.
finishedAt	string, The time that the run completed.
error	string, The error message for this run, if present.

`exports_list_files_csv_runs_logs` *Get the logs for a run*

Description

Get the logs for a run

Usage

```
exports_list_files_csv_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the CSV Export job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

exports_list_files_csv_runs_outputs
List the outputs for a run

Description

List the outputs for a run

Usage

```
exports_list_files_csv_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the csv_export.
run_id	integer required. The ID of the run.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

`exports_patch_files_csv` *Update some attributes of this CSV Export*

Description

Update some attributes of this CSV Export

Usage

```
exports_patch_files_csv(
  id,
  name = NULL,
  source = NULL,
  destination = NULL,
  include_header = NULL,
  compression = NULL,
  column_delimiter = NULL,
  hidden = NULL,
  force_multifile = NULL,
  max_file_size = NULL
)
```

Arguments

`id`	integer required. The ID of this Csv Export job.
`name`	string optional. The name of this Csv Export job.
`source`	list optional. A list containing the following elements: • sql string, The SQL query for this Csv Export job • remoteHostId integer, The ID of the destination database host. • credentialId integer, The ID of the credentials for the destination database.
`destination`	list optional. A list containing the following elements: • filenamePrefix string, The prefix of the name of the file returned to the user. • storagePath list . A list containing the following elements: – filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/" – storageHostId integer, The ID of the destination storage host. – credentialId integer, The ID of the credentials for the destination storage host. – existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path.One of: fail, append, overwrite. Default: fail. If "append" is specified,the new file will always be added to the provided path. If "overwrite" is specifiedall existing files at the provided path will be deleted and the new file will be

added.By default, or if "fail" is specified, the export will fail if a file exists at the provided path.

`include_header` boolean optional. A boolean value indicating whether or not the header should be included. Defaults to true.

`compression` string optional. The compression of the output file. Valid arguments are "gzip" and "none". Defaults to "gzip".

`column_delimiter` string optional. The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`hidden` boolean optional. A boolean value indicating whether or not this request should be hidden. Defaults to false.

`force_multifile` boolean optional. Whether or not the csv should be split into multiple files. Default: false

`max_file_size` integer optional. The max file size, in MB, created files will be. Only available when force_multifile is true.

Value

A list containing the following elements:

`id` integer, The ID of this Csv Export job.

`name` string, The name of this Csv Export job.

`source` list, A list containing the following elements:

- sql string, The SQL query for this Csv Export job
- remoteHostId integer, The ID of the destination database host.
- credentialId integer, The ID of the credentials for the destination database.

`destination` list, A list containing the following elements:

- filenamePrefix string, The prefix of the name of the file returned to the user.
- storagePath list . A list containing the following elements:
  - filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"
  - storageHostId integer, The ID of the destination storage host.
  - credentialId integer, The ID of the credentials for the destination storage host.
  - existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path.One of: fail, append, overwrite. Default: fail. If "append" is specified,the new file will always be added to the provided path. If "overwrite" is specifiedall existing files at the provided path will be deleted and the new file will be added.By default, or if "fail" is specified, the export will fail if a file exists at the provided path.

`includeHeader` boolean, A boolean value indicating whether or not the header should be included. Defaults to true.

compression	string, The compression of the output file. Valid arguments are "gzip" and "none". Defaults to "gzip".
columnDelimiter	string, The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".
hidden	boolean, A boolean value indicating whether or not this request should be hidden. Defaults to false.
forceMultifile	boolean, Whether or not the csv should be split into multiple files. Default: false
maxFileSize	integer, The max file size, in MB, created files will be. Only available when force_multifile is true.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".

`exports_post_files_csv` *Create a CSV Export*

Description

Create a CSV Export

Usage

```
exports_post_files_csv(
  source,
  destination,
  name = NULL,
  include_header = NULL,
  compression = NULL,
  column_delimiter = NULL,
  hidden = NULL,
  force_multifile = NULL,
  max_file_size = NULL
)
```

Arguments

source	list required. A list containing the following elements: • sql string, The SQL query for this Csv Export job • remoteHostId integer, The ID of the destination database host. • credentialId integer, The ID of the credentials for the destination database.
destination	list required. A list containing the following elements: • filenamePrefix string, The prefix of the name of the file returned to the user. • storagePath list . A list containing the following elements:

  - filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"
  - storageHostId integer, The ID of the destination storage host.
  - credentialId integer, The ID of the credentials for the destination storage host.
  - existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path.One of: fail, append, overwrite. Default: fail. If "append" is specified,the new file will always be added to the provided path. If "overwrite" is specifiedall existing files at the provided path will be deleted and the new file will be added.By default, or if "fail" is specified, the export will fail if a file exists at the provided path.

`name` string optional. The name of this Csv Export job.

`include_header` boolean optional. A boolean value indicating whether or not the header should be included. Defaults to true.

`compression` string optional. The compression of the output file. Valid arguments are "gzip" and "none". Defaults to "gzip".

`column_delimiter` string optional. The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`hidden` boolean optional. A boolean value indicating whether or not this request should be hidden. Defaults to false.

`force_multifile` boolean optional. Whether or not the csv should be split into multiple files. Default: false

`max_file_size` integer optional. The max file size, in MB, created files will be. Only available when force_multifile is true.

Value

A list containing the following elements:

`id` integer, The ID of this Csv Export job.

`name` string, The name of this Csv Export job.

`source` list, A list containing the following elements:

- sql string, The SQL query for this Csv Export job
- remoteHostId integer, The ID of the destination database host.
- credentialId integer, The ID of the credentials for the destination database.

`destination` list, A list containing the following elements:

- filenamePrefix string, The prefix of the name of the file returned to the user.
- storagePath list . A list containing the following elements:
  - filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"

- storageHostId integer, The ID of the destination storage host.
- credentialId integer, The ID of the credentials for the destination storage host.
- existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path.One of: fail, append, overwrite. Default: fail. If "append" is specified,the new file will always be added to the provided path. If "overwrite" is specifiedall existing files at the provided path will be deleted and the new file will be added.By default, or if "fail" is specified, the export will fail if a file exists at the provided path.

`includeHeader` boolean, A boolean value indicating whether or not the header should be included. Defaults to true.

`compression` string, The compression of the output file. Valid arguments are "gzip" and "none". Defaults to "gzip".

`columnDelimiter` string, The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`hidden` boolean, A boolean value indicating whether or not this request should be hidden. Defaults to false.

`forceMultifile` boolean, Whether or not the csv should be split into multiple files. Default: false

`maxFileSize` integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`exports_post_files_csv_runs` *Start a run*

Description

Start a run

Usage

```
exports_post_files_csv_runs(id)
```

Arguments

`id` integer required. The ID of the CSV Export job.

Value

A list containing the following elements:

id	integer,
state	string,
createdAt	string, The time that the run was queued.
startedAt	string, The time that the run started.
finishedAt	string, The time that the run completed.
error	string, The error message for this run, if present.
outputCachedOn	string, The time that the output was originally exported, if a cache entry was used by the run.

---

exports_put_files_csv *Replace all attributes of this CSV Export*

Description

Replace all attributes of this CSV Export

Usage

```
exports_put_files_csv(
  id,
  source,
  destination,
  name = NULL,
  include_header = NULL,
  compression = NULL,
  column_delimiter = NULL,
  hidden = NULL,
  force_multifile = NULL,
  max_file_size = NULL
)
```

Arguments

id	integer required. The ID of this Csv Export job.
source	list required. A list containing the following elements: • sql string, The SQL query for this Csv Export job • remoteHostId integer, The ID of the destination database host. • credentialId integer, The ID of the credentials for the destination database.
destination	list required. A list containing the following elements: • filenamePrefix string, The prefix of the name of the file returned to the user.

- storagePath list . A list containing the following elements:
  - filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"
  - storageHostId integer, The ID of the destination storage host.
  - credentialId integer, The ID of the credentials for the destination storage host.
  - existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path.One of: fail, append, overwrite. Default: fail. If "append" is specified,the new file will always be added to the provided path. If "overwrite" is specifiedall existing files at the provided path will be deleted and the new file will be added.By default, or if "fail" is specified, the export will fail if a file exists at the provided path.

`name` string optional. The name of this Csv Export job.

`include_header` boolean optional. A boolean value indicating whether or not the header should be included. Defaults to true.

`compression` string optional. The compression of the output file. Valid arguments are "gzip" and "none". Defaults to "gzip".

`column_delimiter` string optional. The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`hidden` boolean optional. A boolean value indicating whether or not this request should be hidden. Defaults to false.

`force_multifile` boolean optional. Whether or not the csv should be split into multiple files. Default: false

`max_file_size` integer optional. The max file size, in MB, created files will be. Only available when force_multifile is true.

Value

A list containing the following elements:

`id` integer, The ID of this Csv Export job.

`name` string, The name of this Csv Export job.

`source` list, A list containing the following elements:

- sql string, The SQL query for this Csv Export job
- remoteHostId integer, The ID of the destination database host.
- credentialId integer, The ID of the credentials for the destination database.

`destination` list, A list containing the following elements:

- filenamePrefix string, The prefix of the name of the file returned to the user.
- storagePath list . A list containing the following elements:

- filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"
- storageHostId integer, The ID of the destination storage host.
- credentialId integer, The ID of the credentials for the destination storage host.
- existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path.One of: fail, append, overwrite. Default: fail. If "append" is specified,the new file will always be added to the provided path. If "overwrite" is specifiedall existing files at the provided path will be deleted and the new file will be added.By default, or if "fail" is specified, the export will fail if a file exists at the provided path.

`includeHeader` boolean, A boolean value indicating whether or not the header should be included. Defaults to true.

`compression` string, The compression of the output file. Valid arguments are "gzip" and "none". Defaults to "gzip".

`columnDelimiter` string, The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`hidden` boolean, A boolean value indicating whether or not this request should be hidden. Defaults to false.

`forceMultifile` boolean, Whether or not the csv should be split into multiple files. Default: false

`maxFileSize` integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`exports_put_files_csv_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
exports_put_files_csv_archive(id, status)
```

Arguments

`id` integer required. The ID of the object.

`status` boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id` integer, The ID of this Csv Export job.

`name` string, The name of this Csv Export job.

`source` list, A list containing the following elements:

- sql string, The SQL query for this Csv Export job
- remoteHostId integer, The ID of the destination database host.
- credentialId integer, The ID of the credentials for the destination database.

`destination` list, A list containing the following elements:

- filenamePrefix string, The prefix of the name of the file returned to the user.
- storagePath list . A list containing the following elements:
  - filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"
  - storageHostId integer, The ID of the destination storage host.
  - credentialId integer, The ID of the credentials for the destination storage host.
  - existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path.One of: fail, append, overwrite. Default: fail. If "append" is specified,the new file will always be added to the provided path. If "overwrite" is specifiedall existing files at the provided path will be deleted and the new file will be added.By default, or if "fail" is specified, the export will fail if a file exists at the provided path.

`includeHeader` boolean, A boolean value indicating whether or not the header should be included. Defaults to true.

`compression` string, The compression of the output file. Valid arguments are "gzip" and "none". Defaults to "gzip".

`columnDelimiter` string, The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`hidden` boolean, A boolean value indicating whether or not this request should be hidden. Defaults to false.

`forceMultifile` boolean, Whether or not the csv should be split into multiple files. Default: false

`maxFileSize` integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

fetch_all *Retrieve all results from a paginated endpoint*

Description

Retrieve all results from a paginated endpoint

Usage

```
fetch_all(fn, ...)
```

Arguments

fn	The API function to be called.
...	Arguments passed to fn.

Value

A list with the concatenated results of each page of fn.

See Also

Other pagination: fetch_until()

Examples

```
## Not run:
columns <- fetch_all(tables_list_columns, id = 123)
column_names <- columns %>% purrr::map_chr("name")

## End(Not run)
```

fetch_and_generate_client *Fetches and generates the client in generated_client.R*

Description

Fetches and generates the client in generated_client.R

Usage

```
fetch_and_generate_client()
```

Details

Skips autogeneration on windows with R < 3.4.0 and if R_CLIENT_DEV == "TRUE". A valid CIVIS_API_KEY must be set.

`fetch_logs` *Fetch job logs from the Civis Platform*

Description

Fetch job logs from the Civis Platform

Usage

```
fetch_logs(object, limit = 100, ...)
```

Arguments

`object`	A `civis_api`, `civis_error`, or `civis_ml` object.
`limit`	The number of log lines to fetch, the default is 100.
`...`	Extra parameters passed to methods, currently unused.

Value

A list of log messages with class `civis_logs`.

Examples

```
## Not run:
m <- civis_ml_fetch_existing(123)
fetch_logs(m)

import <- write_civis(iris, "scratch.mytest")
fetch_logs(import)

query_run <- query_civis(123)
fetch_logs(query_run)

e <- tryCatch(read_civis("asdf;"), error = function(e) e)
fetch_logs(e)

## End(Not run)
```

fetch_oos_scores *Retrieve out of sample/fold predictions*

Description

Retrieve out of sample/fold predictions

Usage

```
fetch_oos_scores(model, ...)
```

Arguments

model	A civis_ml model.
...	Parameters passed to read.csv.

Details

Returns the out of sample (or out of fold) predictions for each observation. For a multiclass model, a data frame is returned with one column of predictions for each class.

Value

A data.frame with out of sample/fold predictions for each row of the training data, and containing an additional column with a primary key.

See Also

civis_ml

fetch_output_file_ids *Return output from a civis_script.*

Description

Return output from a civis_script.

Usage

```
fetch_output_file_ids(x, regex = NULL)

fetch_output(x, regex = NULL)
```

Arguments

x	civis_script
regex	string, regex used to match the run output name.

Details

If the script has no outputs, the results are a list of length is 0.

Value

A named list of run output file ids with names matching `regex`.

Functions

- `fetch_output()`: Return output of `scripts_list_*_runs_outputs` matching `regex`.

See Also

Other script_utils: `civis_script()`, `run_civis()`, `run_template()`

Other script_utils: `civis_script()`, `run_civis()`, `run_template()`

Examples

```
## Not run:
out <- fetch_output(civis_script(1234))

# Filter output with regex, then read into memory:
ids <- fetch_output_file_ids(civis_script(1234), regex = '.csv')
vals <- lapply(ids, read_civis, using = read.csv)

## End(Not run)
```

`fetch_predictions` *Retrieve predictions from a CivisML prediction job*

Description

Retrieve predictions from a CivisML prediction job

Usage

```
fetch_predictions(x, ...)
```

Arguments

`x`	`civis_ml_prediction` object from `predict.civis_ml`
`...`	arguments passed to `read.csv`

Details

Predictions can also be downloaded as a `csv` directly using `download_civis` (see examples).

Value

A data.frame of predictions containing an additional column with a primary key. For a multiclass model, a data frame is returned with one column of predictions for each class.

Examples

```
## Not run:
m <- civis_ml("path/to/file.csv", model_type = "sparse_logistic",
  dependent_variable = "Species")
pred_job <- predict(m, newdata = "path/to/newdata.csv")
yhat <- fetch_predictions(pred_job)

# download instead:
download_civis(pred_job$model_info$output_file_ids, path = "my_predictions.csv")


## End(Not run)
```

---

fetch_until *Retrieve some results from a paginated endpoint*

Description

fetch_until will retrieve paginated results until a condition is met. This is useful when searching for a particular value or record.

Usage

```
fetch_until(fn, .until, ...)
```

Arguments

fn	The API function to be called.
.until	A function which returns a boolean value. .until will be called with each item from the API response. When .until returns TRUE iteration will stop and fetch_until will return all responses accumulated so far.
...	Arguments passed to fn.

Value

A list with the concatenated results of each page of fn.

See Also

Other pagination: fetch_all()

Examples

```
## Not run:
columns <- fetch_until(tables_list_columns,
                       .until = function(x) x == "voterbase_id")

## End(Not run)
```

`files_delete_preprocess_csv` *Archive a Preprocess CSV (deprecated, use archiving endpoints instead)*

Description

Archive a Preprocess CSV (deprecated, use archiving endpoints instead)

Usage

```
files_delete_preprocess_csv(id)
```

Arguments

`id`	integer required.

Value

An empty HTTP response

`files_delete_projects` *Remove a File from a project*

Description

Remove a File from a project

Usage

```
files_delete_projects(id, project_id)
```

Arguments

`id`	integer required. The ID of the File.
`project_id`	integer required. The ID of the project.

Value

An empty HTTP response

files_delete_shares_groups *Revoke the permissions a group has on this object*

Description

Revoke the permissions a group has on this object

Usage

```
files_delete_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

files_delete_shares_users *Revoke the permissions a user has on this object*

Description

Revoke the permissions a user has on this object

Usage

```
files_delete_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

`files_get` *Get details about a file*

Description

Get details about a file

Usage

```
files_get(id, link_expires_at = NULL, inline = NULL)
```

Arguments

`id`	integer required. The ID of the file.
`link_expires_at`	string optional. The date and time the download link will expire. Must be a time between now and 36 hours from now. Defaults to 30 minutes from now.
`inline`	boolean optional. If true, will return a url that can be displayed inline in HTML

Value

A list containing the following elements:

`id`	integer, The ID of the file.
`name`	string, The file name.
`createdAt`	string, The date and time the file was created.
`fileSize`	integer, The file size.
`expiresAt`	string, The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.
`description`	string, The user-defined description of the file.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`downloadUrl`	string, A JSON string containing information about the URL of the file.
`fileUrl`	string, The URL that may be used to download the file.
`detectedInfo`	list, A list containing the following elements: • includeHeader boolean, A boolean value indicating whether or not the first row of the file is a header row. • columnDelimiter string, The column delimiter for the file. One of "comma", "tab", or "pipe".

- compression string, The type of compression of the file. One of "gzip", or "none".
- tableColumns array, An array of hashes corresponding to the columns in the file. Each hash should have keys for column "name" and "sql_type"

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`files_get_preprocess_csv` *Get a Preprocess CSV*

Description

Get a Preprocess CSV

Usage

```
files_get_preprocess_csv(id)
```

Arguments

`id`	integer required.

Value

A list containing the following elements:

`id`	integer, The ID of the job created.
`fileId`	integer, The ID of the file.
`inPlace`	boolean, If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.
`detectTableColumns`	boolean, If true, detect the table columns in the file including the sql types. If false, skip table column detection.Defaults to false.
`forceCharacterSetConversion`	boolean, If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).
`includeHeader`	boolean, A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.
`columnDelimiter`	string, The column delimiter for the file. One of "comma", "tab", or "pipe". If not provided, the column delimiter will be auto-detected.
`hidden`	boolean, The hidden status of the item.

files_list_dependencies *List dependent objects for this object*

Description

List dependent objects for this object

Usage

```
files_list_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

files_list_projects *List the projects a File belongs to*

Description

List the projects a File belongs to

Usage

```
files_list_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the File.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

---

files_list_shares *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
files_list_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares`
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

files_patch *Update details about a file*

Description

Update details about a file

Usage

```
files_patch(id, name = NULL, expires_at = NULL, description = NULL)
```

Arguments

`id`	integer required. The ID of the file.
`name`	string optional. The file name. The extension must match the previous extension.
`expires_at`	string optional. The date and time the file will expire.
`description`	string optional. The user-defined description of the file.

Value

A list containing the following elements:

`id`	integer, The ID of the file.
`name`	string, The file name.
`createdAt`	string, The date and time the file was created.
`fileSize`	integer, The file size.
`expiresAt`	string, The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.
`description`	string, The user-defined description of the file.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`downloadUrl`	string, A JSON string containing information about the URL of the file.
`fileUrl`	string, The URL that may be used to download the file.
`detectedInfo`	list, A list containing the following elements: • includeHeader boolean, A boolean value indicating whether or not the first row of the file is a header row. • columnDelimiter string, The column delimiter for the file. One of "comma", "tab", or "pipe". • compression string, The type of compression of the file. One of "gzip", or "none". • tableColumns array, An array of hashes corresponding to the columns in the file. Each hash should have keys for column "name" and "sql_type"
`myPermissionLevel`	string, Your permission level on the object. One of "read", "write", or "manage".

---

files_patch_preprocess_csv

Update some attributes of this Preprocess CSV

Description

Update some attributes of this Preprocess CSV

Usage

```
files_patch_preprocess_csv(
  id,
  file_id = NULL,
  in_place = NULL,
  detect_table_columns = NULL,
  force_character_set_conversion = NULL,
  include_header = NULL,
  column_delimiter = NULL
)
```

Arguments

id	integer required. The ID of the job created.
file_id	integer optional. The ID of the file.
in_place	boolean optional. If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.
detect_table_columns	boolean optional. If true, detect the table columns in the file including the sql types. If false, skip table column detection.Defaults to false.
force_character_set_conversion	boolean optional. If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).
include_header	boolean optional. A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.
column_delimiter	string optional. The column delimiter for the file. One of "comma", "tab", or "pipe". If not provided, the column delimiter will be auto-detected.

Value

A list containing the following elements:

id	integer, The ID of the job created.
fileId	integer, The ID of the file.
inPlace	boolean, If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.
detectTableColumns	boolean, If true, detect the table columns in the file including the sql types. If false, skip table column detection.Defaults to false.
forceCharacterSetConversion	boolean, If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).

includeHeader	boolean, A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.
columnDelimiter	string, The column delimiter for the file. One of "comma", "tab", or "pipe". If not provided, the column delimiter will be auto-detected.
hidden	boolean, The hidden status of the item.

`files_post` *Initiate an upload of a file into the platform*

Description

Initiate an upload of a file into the platform

Usage

```
files_post(name, expires_at = NULL, description = NULL)
```

Arguments

name	string required. The file name.
expires_at	string optional. The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.
description	string optional. The user-defined description of the file.

Value

A list containing the following elements:

id	integer, The ID of the file.
name	string, The file name.
createdAt	string, The date and time the file was created.
fileSize	integer, The file size.
expiresAt	string, The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.
description	string, The user-defined description of the file.
uploadUrl	string, The URL that may be used to upload a file. To use the upload URL, initiate a POST request to the given URL with the file you wish to import as the "file" form field.
uploadFields	list, A hash containing the form fields to be included with the POST request.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".

`files_post_multipart` *Initiate a multipart upload*

Description

Initiate a multipart upload

Usage

```
files_post_multipart(name, num_parts, expires_at = NULL, description = NULL)
```

Arguments

`name`	string required. The file name.
`num_parts`	integer required. The number of parts in which the file will be uploaded. This parameter determines the number of presigned URLs that are returned.
`expires_at`	string optional. The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.
`description`	string optional. The user-defined description of the file.

Value

A list containing the following elements:

`id`	integer, The ID of the file.
`name`	string, The file name.
`createdAt`	string, The date and time the file was created.
`fileSize`	integer, The file size.
`expiresAt`	string, The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.
`description`	string, The user-defined description of the file.
`uploadUrls`	array, An array of URLs that may be used to upload file parts. Use separate PUT requests to complete the part uploads. Links expire after 12 hours.

files_post_multipart_complete *Complete a multipart upload*

Description

Complete a multipart upload

Usage

```
files_post_multipart_complete(id)
```

Arguments

id	integer required. The ID of the file.

Value

An empty HTTP response

files_post_preprocess_csv *Create a Preprocess CSV*

Description

Create a Preprocess CSV

Usage

```
files_post_preprocess_csv(
  file_id,
  in_place = NULL,
  detect_table_columns = NULL,
  force_character_set_conversion = NULL,
  include_header = NULL,
  column_delimiter = NULL,
  hidden = NULL
)
```

Arguments

file_id	integer required. The ID of the file.
in_place	boolean optional. If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.
detect_table_columns	boolean optional. If true, detect the table columns in the file including the sql types. If false, skip table column detection.Defaults to false.
force_character_set_conversion	boolean optional. If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).
include_header	boolean optional. A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.
column_delimiter	string optional. The column delimiter for the file. One of "comma", "tab", or "pipe". If not provided, the column delimiter will be auto-detected.
hidden	boolean optional. The hidden status of the item.

Value

A list containing the following elements:

id	integer, The ID of the job created.
fileId	integer, The ID of the file.
inPlace	boolean, If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.
detectTableColumns	boolean, If true, detect the table columns in the file including the sql types. If false, skip table column detection.Defaults to false.
forceCharacterSetConversion	boolean, If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).
includeHeader	boolean, A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.
columnDelimiter	string, The column delimiter for the file. One of "comma", "tab", or "pipe". If not provided, the column delimiter will be auto-detected.
hidden	boolean, The hidden status of the item.

files_put *Update details about a file*

Description

Update details about a file

Usage

```
files_put(id, name, expires_at, description = NULL)
```

Arguments

id	integer required. The ID of the file.
name	string required. The file name. The extension must match the previous extension.
expires_at	string required. The date and time the file will expire.
description	string optional. The user-defined description of the file.

Value

A list containing the following elements:

id	integer, The ID of the file.
name	string, The file name.
createdAt	string, The date and time the file was created.
fileSize	integer, The file size.
expiresAt	string, The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.
description	string, The user-defined description of the file.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
downloadUrl	string, A JSON string containing information about the URL of the file.
fileUrl	string, The URL that may be used to download the file.
detectedInfo	list, A list containing the following elements: • includeHeader boolean, A boolean value indicating whether or not the first row of the file is a header row. • columnDelimiter string, The column delimiter for the file. One of "comma", "tab", or "pipe".

- compression string, The type of compression of the file. One of "gzip", or "none".
- tableColumns array, An array of hashes corresponding to the columns in the file. Each hash should have keys for column "name" and "sql_type"

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`files_put_preprocess_csv`
Replace all attributes of this Preprocess CSV

Description

Replace all attributes of this Preprocess CSV

Usage

```
files_put_preprocess_csv(
  id,
  file_id,
  in_place = NULL,
  detect_table_columns = NULL,
  force_character_set_conversion = NULL,
  include_header = NULL,
  column_delimiter = NULL
)
```

Arguments

`id`	integer required. The ID of the job created.
`file_id`	integer required. The ID of the file.
`in_place`	boolean optional. If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.
`detect_table_columns`	boolean optional. If true, detect the table columns in the file including the sql types. If false, skip table column detection.Defaults to false.
`force_character_set_conversion`	boolean optional. If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).
`include_header`	boolean optional. A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.
`column_delimiter`	string optional. The column delimiter for the file. One of "comma", "tab", or "pipe". If not provided, the column delimiter will be auto-detected.

Value

A list containing the following elements:

id	integer, The ID of the job created.
fileId	integer, The ID of the file.
inPlace	boolean, If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.
detectTableColumns	boolean, If true, detect the table columns in the file including the sql types. If false, skip table column detection.Defaults to false.
forceCharacterSetConversion	boolean, If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).
includeHeader	boolean, A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.
columnDelimiter	string, The column delimiter for the file. One of "comma", "tab", or "pipe". If not provided, the column delimiter will be auto-detected.
hidden	boolean, The hidden status of the item.

---

files_put_preprocess_csv_archive

Update the archive status of this object

Description

Update the archive status of this object

Usage

```
files_put_preprocess_csv_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ID of the job created.
fileId	integer, The ID of the file.
inPlace	boolean, If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.
detectTableColumns	boolean, If true, detect the table columns in the file including the sql types. If false, skip table column detection.Defaults to false.
forceCharacterSetConversion	boolean, If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).
includeHeader	boolean, A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.
columnDelimiter	string, The column delimiter for the file. One of "comma", "tab", or "pipe". If not provided, the column delimiter will be auto-detected.
hidden	boolean, The hidden status of the item.

---

files_put_projects *Add a File to a project*

Description

Add a File to a project

Usage

```
files_put_projects(id, project_id)
```

Arguments

id	integer required. The ID of the File.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

files_put_shares_groups *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
files_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`files_put_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
files_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_ids`	array required. An array of one or more user IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,

- groups array,

totalUserShares
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`files_put_transfer` *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
files_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies
: array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object

- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`generate_client` *Generate a client*

Description

Generate a client

Usage

```
generate_client(spec)
```

Arguments

`spec`	usually from `get_spec`

Value

A string containing one documented function for each verb at each endpoint.

`get_database_id` *Return the database id for a given database name*

Description

Return the database id for a given database name

Usage

```
get_database_id(database_name)
```

Arguments

`database_name`	string, Name of database.

Value

database_id integer, id of the matching database.

get_error *Get error data from civis_errors*

Description

Get error data from civis_errors

Usage

```
get_error(x)
```

Arguments

x	The error, usually from tryCatch

Value

A list containing

f	The function
args	A list of named arguments f was called with
error	The error message from platform (if any)
status	The status

Examples

```
## Not run:
 q_id <- queries_post(db_id, query, n_rows, cred_id)[["id"]]
 r <- tryCatch(await(queries_get, id = q_id), error = function(e) e)
 print(r)
 get_error(r)

 r <- try(await(queries_get, id = q_id))
 get_error(r)

## End(Not run)
```

get_feature_importance *Get model feature importance*

Description

Get model feature importance

Usage

```
get_feature_importance(model)
```

Arguments

model	Object from civis_ml

Value

A matrix of features and their importance, ranked in descending order of importance

get_job_type_version *Derive the job type and version from the given alias.*

Description

Derive the job type and version from the given alias.

Usage

```
get_job_type_version(alias)
```

Arguments

alias	A one-length character vector of the CivisML alias

Value

A list containing the job_type and version

get_metric *Get model metrics*

Description

Get model metrics

Usage

```
get_metric(model, name = NULL)
```

Arguments

model	Object from civis_ml
name	Name of the metric. If NULL, all metrics are returned. Possible metrics are listed in details.

Details

The list of possible metrics provided by CivisML is:

Classification:

"accuracy", "confusion_matrix", "p_correct", "pop_incidence_true", "pop_incidence_pred", "roc_auc", "log_loss", "brier_score", "roc_curve", "calibration_curve", "deciles", "score_histogram", "training_histogram", "oos_score_table".

Regression:

"mad", "rmse", "r_squared", "score_histogram", "training_histogram", "y_yhat_plot", "y_yhat_outlier_rows", "oos_score_table"

Value

The metric given by name, or all metrics.

get_script_fun *Get a script function matching a job type.*

Description

Get a script function matching a job type.

Usage

```
get_script_fun(job, verb = c("list", "post"), fun_type = c("outputs", "logs"))
```

Arguments

job	output of jobs_get
verb	one of "list" or "post"
fun_type	one of "logs" or "outputs"

Details

container and custom scripts both have the same job type, but can be distinguished by a non-null fromTemplateId.

Value

The correct output or log fetching function (e.g. scripts_list_containers_runs_logs) based on the job type.

---

get_status *Get the status from results of* await

Description

Get the status from results of await

Usage

```
get_status(response)
```

Arguments

response	the results from await

See Also

await await_all

get_table_id *Return the table id for a given table*

Description

Return the table id for a given table

Usage

```
get_table_id(table_name, database = NULL)
```

Arguments

table_name	string, Name of table and schema "schema.tablename"
database	string, Name of database where data frame is to be uploaded. If no database is specified, uses options(civis.default_db)

Value

table_id integer, id of the matching table.

See Also

Other tables: refresh_table(), transfer_table()

get_template_ids_all_versions *Get template IDs for all accessible CivisML versions.*

Description

Get template IDs for all accessible CivisML versions.

Usage

```
get_template_ids_all_versions()
```

Value

A data frame containing the template id, CivisML version, and job type.

get_train_template_id *Get training template IDs for the specified CivisML version*

Description

Get training template IDs for the specified CivisML version

Usage

```
get_train_template_id(civisml_version = "prod")
```

Arguments

civisml_version	A one-length character vector of the CivisML version

Value

The template ID for training

git_repos_delete *Remove the bookmark on a git repository*

Description

Remove the bookmark on a git repository

Usage

```
git_repos_delete(id)
```

Arguments

id	integer required. The ID for this git repository.

Value

An empty HTTP response

git_repos_get *Get a bookmarked git repository*

Description

Get a bookmarked git repository

Usage

```
git_repos_get(id)
```

Arguments

id	integer required. The ID for this git repository.

Value

A list containing the following elements:

id	integer, The ID for this git repository.
repoUrl	string, The URL for this git repository.
createdAt	string,
updatedAt	string,

git_repos_list *List bookmarked git repositories*

Description

List bookmarked git repositories

Usage

```
git_repos_list(limit = NULL, page_num = NULL, order = NULL, order_dir = NULL)
```

Arguments

limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to repo_url. Must be one of: repo_url, created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id	integer, The ID for this git repository.
repoUrl	string, The URL for this git repository.
createdAt	string,
updatedAt	string,

---

git_repos_list_refs *Get all branches and tags of a bookmarked git repository*

Description

Get all branches and tags of a bookmarked git repository

Usage

```
git_repos_list_refs(id)
```

Arguments

id	integer required. The ID for this git repository.

Value

A list containing the following elements:

branches	array, List of branch names of this git repository.
tags	array, List of tag names of this git repository.

---

git_repos_post *Bookmark a git repository*

Description

Bookmark a git repository

Usage

```
git_repos_post(repo_url)
```

Arguments

repo_url	string required. The URL for this git repository.

Value

A list containing the following elements:

id	integer, The ID for this git repository.
repoUrl	string, The URL for this git repository.
createdAt	string,
updatedAt	string,

---

groups_delete *Delete a Group (deprecated)*

Description

Delete a Group (deprecated)

Usage

```
groups_delete(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

---

groups_delete_members *Remove a user from a group*

Description

Remove a user from a group

Usage

```
groups_delete_members(id, user_id)
```

Arguments

id	integer required. The ID of the group.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

groups_delete_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
groups_delete_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

---

groups_delete_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
groups_delete_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

groups_get *Get a Group*

Description

Get a Group

Usage

```
groups_get(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID of this group.
name	string, This group's name.
createdAt	string, The date and time when this group was created.
updatedAt	string, The date and time when this group was last updated.
description	string, The description of the group.
slug	string, The slug for this group.
organizationId	integer, The ID of the organization this group belongs to.
organizationName	string, The name of the organization this group belongs to.
memberCount	integer, The number of active members in this group.
totalMemberCount	integer, The total number of members in this group.
defaultOtpRequiredForLogin	boolean, The two factor authentication requirement for this group.
roleIds	array, An array of ids of all the roles this group has.
defaultTimeZone	string, The default time zone of this group.
defaultJobsLabel	string, The default partition label for jobs of this group.
defaultNotebooksLabel	string, The default partition label for notebooks of this group.
defaultServicesLabel	string, The default partition label for services of this group.
lastUpdatedById	integer, The ID of the user who last updated this group.

createdById integer, The ID of the user who created this group.

members array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.
- email string, This user's email address.
- primaryGroupId integer, The ID of the primary group of this user.
- active boolean, Whether this user account is active or deactivated.

`groups_list` *List Groups*

Description

List Groups

Usage

```
groups_list(
  query = NULL,
  permission = NULL,
  include_members = NULL,
  organization_id = NULL,
  user_ids = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

query	string optional. If specified, it will filter the groups returned.
permission	string optional. A permissions string, one of "read", "write", or "manage". Lists only groups for which the current user has that permission.
include_members	boolean optional. Show members of the group.
organization_id	integer optional. The organization by which to filter groups.
user_ids	array optional. A list of user IDs to filter groups by.Groups will be returned if any of the users is a member
limit	integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.

page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to name. Must be one of: name, created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id	integer, The ID of this group.
name	string, This group's name.
createdAt	string, The date and time when this group was created.
updatedAt	string, The date and time when this group was last updated.
description	string, The description of the group.
slug	string, The slug for this group.
organizationId	integer, The ID of the organization this group belongs to.
organizationName	string, The name of the organization this group belongs to.
memberCount	integer, The number of active members in this group.
totalMemberCount	integer, The total number of members in this group.
lastUpdatedById	integer, The ID of the user who last updated this group.
createdById	integer, The ID of the user who created this group.
members	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.

---

groups_list_child_groups

Get child groups of this group

Description

Get child groups of this group

Usage

```
groups_list_child_groups(id)
```

Arguments

id	integer required. The ID of this group.

Value

A list containing the following elements:

manageable	array, An array containing the following fields: • id integer, • name string,
writeable	array, An array containing the following fields: • id integer, • name string,
readable	array, An array containing the following fields: • id integer, • name string,

---

groups_list_shares *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
groups_list_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array,

- groups array,

owners list, A list containing the following elements:

- users array,
- groups array,

totalUserShares
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`groups_patch` *Update some attributes of this Group*

Description

Update some attributes of this Group

Usage

```
groups_patch(
  id,
  name = NULL,
  description = NULL,
  slug = NULL,
  organization_id = NULL,
  default_otp_required_for_login = NULL,
  role_ids = NULL,
  default_time_zone = NULL,
  default_jobs_label = NULL,
  default_notebooks_label = NULL,
  default_services_label = NULL
)
```

Arguments

id
: integer required. The ID of this group.

name
: string optional. This group's name.

description
: string optional. The description of the group.

slug
: string optional. The slug for this group.

organization_id
: integer optional. The ID of the organization this group belongs to.

default_otp_required_for_login
: boolean optional. The two factor authentication requirement for this group.

`role_ids` array optional. An array of ids of all the roles this group has.

`default_time_zone`
string optional. The default time zone of this group.

`default_jobs_label`
string optional. The default partition label for jobs of this group.

`default_notebooks_label`
string optional. The default partition label for notebooks of this group.

`default_services_label`
string optional. The default partition label for services of this group.

Value

A list containing the following elements:

`id` integer, The ID of this group.

`name` string, This group's name.

`createdAt` string, The date and time when this group was created.

`updatedAt` string, The date and time when this group was last updated.

`description` string, The description of the group.

`slug` string, The slug for this group.

`organizationId` integer, The ID of the organization this group belongs to.

`organizationName`
string, The name of the organization this group belongs to.

`memberCount` integer, The number of active members in this group.

`totalMemberCount`
integer, The total number of members in this group.

`defaultOtpRequiredForLogin`
boolean, The two factor authentication requirement for this group.

`roleIds` array, An array of ids of all the roles this group has.

`defaultTimeZone`
string, The default time zone of this group.

`defaultJobsLabel`
string, The default partition label for jobs of this group.

`defaultNotebooksLabel`
string, The default partition label for notebooks of this group.

`defaultServicesLabel`
string, The default partition label for services of this group.

`lastUpdatedById`
integer, The ID of the user who last updated this group.

`createdById` integer, The ID of the user who created this group.

`members` array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.

- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.
- email string, This user's email address.
- primaryGroupId integer, The ID of the primary group of this user.
- active boolean, Whether this user account is active or deactivated.

`groups_post` *Create a Group*

Description

Create a Group

Usage

```
groups_post(
  name,
  description = NULL,
  slug = NULL,
  organization_id = NULL,
  default_otp_required_for_login = NULL,
  role_ids = NULL,
  default_time_zone = NULL,
  default_jobs_label = NULL,
  default_notebooks_label = NULL,
  default_services_label = NULL
)
```

Arguments

`name`	string required. This group's name.
`description`	string optional. The description of the group.
`slug`	string optional. The slug for this group.
`organization_id`	integer optional. The ID of the organization this group belongs to.
`default_otp_required_for_login`	boolean optional. The two factor authentication requirement for this group.
`role_ids`	array optional. An array of ids of all the roles this group has.
`default_time_zone`	string optional. The default time zone of this group.
`default_jobs_label`	string optional. The default partition label for jobs of this group.
`default_notebooks_label`	string optional. The default partition label for notebooks of this group.
`default_services_label`	string optional. The default partition label for services of this group.

Value

A list containing the following elements:

`id` integer, The ID of this group.

`name` string, This group's name.

`createdAt` string, The date and time when this group was created.

`updatedAt` string, The date and time when this group was last updated.

`description` string, The description of the group.

`slug` string, The slug for this group.

`organizationId` integer, The ID of the organization this group belongs to.

`organizationName` string, The name of the organization this group belongs to.

`memberCount` integer, The number of active members in this group.

`totalMemberCount` integer, The total number of members in this group.

`defaultOtpRequiredForLogin` boolean, The two factor authentication requirement for this group.

`roleIds` array, An array of ids of all the roles this group has.

`defaultTimeZone` string, The default time zone of this group.

`defaultJobsLabel` string, The default partition label for jobs of this group.

`defaultNotebooksLabel` string, The default partition label for notebooks of this group.

`defaultServicesLabel` string, The default partition label for services of this group.

`lastUpdatedById` integer, The ID of the user who last updated this group.

`createdById` integer, The ID of the user who created this group.

`members` array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.
- email string, This user's email address.
- primaryGroupId integer, The ID of the primary group of this user.
- active boolean, Whether this user account is active or deactivated.

groups_put *Replace all attributes of this Group*

Description

Replace all attributes of this Group

Usage

```
groups_put(
  id,
  name,
  description = NULL,
  slug = NULL,
  organization_id = NULL,
  default_otp_required_for_login = NULL,
  role_ids = NULL,
  default_time_zone = NULL,
  default_jobs_label = NULL,
  default_notebooks_label = NULL,
  default_services_label = NULL
)
```

Arguments

id	integer required. The ID of this group.
name	string required. This group's name.
description	string optional. The description of the group.
slug	string optional. The slug for this group.
organization_id	integer optional. The ID of the organization this group belongs to.
default_otp_required_for_login	boolean optional. The two factor authentication requirement for this group.
role_ids	array optional. An array of ids of all the roles this group has.
default_time_zone	string optional. The default time zone of this group.
default_jobs_label	string optional. The default partition label for jobs of this group.
default_notebooks_label	string optional. The default partition label for notebooks of this group.
default_services_label	string optional. The default partition label for services of this group.

Value

A list containing the following elements:

`id` integer, The ID of this group.

`name` string, This group's name.

`createdAt` string, The date and time when this group was created.

`updatedAt` string, The date and time when this group was last updated.

`description` string, The description of the group.

`slug` string, The slug for this group.

`organizationId` integer, The ID of the organization this group belongs to.

`organizationName` string, The name of the organization this group belongs to.

`memberCount` integer, The number of active members in this group.

`totalMemberCount` integer, The total number of members in this group.

`defaultOtpRequiredForLogin` boolean, The two factor authentication requirement for this group.

`roleIds` array, An array of ids of all the roles this group has.

`defaultTimeZone` string, The default time zone of this group.

`defaultJobsLabel` string, The default partition label for jobs of this group.

`defaultNotebooksLabel` string, The default partition label for notebooks of this group.

`defaultServicesLabel` string, The default partition label for services of this group.

`lastUpdatedById` integer, The ID of the user who last updated this group.

`createdById` integer, The ID of the user who created this group.

`members` array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.
- email string, This user's email address.
- primaryGroupId integer, The ID of the primary group of this user.
- active boolean, Whether this user account is active or deactivated.

`groups_put_members` *Add a user to a group*

Description

Add a user to a group

Usage

```
groups_put_members(id, user_id)
```

Arguments

`id`	integer required. The ID of the group.
`user_id`	integer required. The ID of the user.

Value

A list containing the following elements:

`id`	integer, The ID of this group.
`name`	string, This group's name.
`createdAt`	string, The date and time when this group was created.
`updatedAt`	string, The date and time when this group was last updated.
`description`	string, The description of the group.
`slug`	string, The slug for this group.
`organizationId`	integer, The ID of the organization this group belongs to.
`organizationName`	string, The name of the organization this group belongs to.
`memberCount`	integer, The number of active members in this group.
`totalMemberCount`	integer, The total number of members in this group.
`defaultOtpRequiredForLogin`	boolean, The two factor authentication requirement for this group.
`roleIds`	array, An array of ids of all the roles this group has.
`defaultTimeZone`	string, The default time zone of this group.
`defaultJobsLabel`	string, The default partition label for jobs of this group.
`defaultNotebooksLabel`	string, The default partition label for notebooks of this group.
`defaultServicesLabel`	string, The default partition label for services of this group.

`lastUpdatedById`
: integer, The ID of the user who last updated this group.

`createdById`
: integer, The ID of the user who created this group.

`members`
: array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.
- email string, This user's email address.
- primaryGroupId integer, The ID of the primary group of this user.
- active boolean, Whether this user account is active or deactivated.

`groups_put_shares_groups`
Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
groups_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`
: integer required. The ID of the resource that is shared.

`group_ids`
: array required. An array of one or more group IDs.

`permission_level`
: string required. Options are: "read", "write", or "manage".

`share_email_body`
: string optional. Custom body text for e-mail sent on a share.

`send_shared_email`
: boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- `readers` list, A list containing the following elements:
  - users array,
  - groups array,
- `writers` list, A list containing the following elements:
  - users array,
  - groups array,
- `owners` list, A list containing the following elements:
  - users array,
  - groups array,
- `totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- `totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`groups_put_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
groups_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

- `id` integer required. The ID of the resource that is shared.
- `user_ids` array required. An array of one or more user IDs.
- `permission_level` string required. Options are: "read", "write", or "manage".
- `share_email_body` string optional. Custom body text for e-mail sent on a share.
- `send_shared_email` boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares`
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`hist.civis_ml` *Plot a histogram of the predicted scores or probabilities for a CivisML object*

Description

Plot a histogram of the predicted scores or probabilities for a CivisML object

Usage

```
## S3 method for class 'civis_ml'
hist(x, name = NULL, ...)
```

Arguments

`x`	civis_ml object
`name`	name of the class (for multiclass output)
`...`	unused.

Value

A `ggplot2` plot object is returned invisibly.

imports_delete_files_csv

Archive a CSV Import (deprecated, use archiving endpoints instead)

Description

Archive a CSV Import (deprecated, use archiving endpoints instead)

Usage

```
imports_delete_files_csv(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

imports_delete_files_csv_runs

Cancel a run

Description

Cancel a run

Usage

```
imports_delete_files_csv_runs(id, run_id)
```

Arguments

id	integer required. The ID of the CSV Import job.
run_id	integer required. The ID of the run.

Value

An empty HTTP response

`imports_delete_files_runs` *Cancel a run*

Description

Cancel a run

Usage

```
imports_delete_files_runs(id, run_id)
```

Arguments

`id`	integer required. The ID of the Import job.
`run_id`	integer required. The ID of the run.

Value

An empty HTTP response

`imports_delete_projects` *Remove an Import from a project*

Description

Remove an Import from a project

Usage

```
imports_delete_projects(id, project_id)
```

Arguments

`id`	integer required. The ID of the Import.
`project_id`	integer required. The ID of the project.

Value

An empty HTTP response

`imports_delete_shares_groups` *Revoke the permissions a group has on this object*

Description

Revoke the permissions a group has on this object

Usage

```
imports_delete_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

`imports_delete_shares_users` *Revoke the permissions a user has on this object*

Description

Revoke the permissions a user has on this object

Usage

```
imports_delete_shares_users(id, user_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. The ID of the user.

Value

An empty HTTP response

imports_delete_syncs *Archive a sync (deprecated, use the /archive endpoint instead)*

Description

Archive a sync (deprecated, use the /archive endpoint instead)

Usage

```
imports_delete_syncs(id, sync_id)
```

Arguments

id	integer required. The ID of the import to fetch.
sync_id	integer required. The ID of the sync to fetch.

Value

An empty HTTP response

imports_get *Get details about an import*

Description

Get details about an import

Usage

```
imports_get(id)
```

Arguments

id	integer required. The ID for the import.

Value

A list containing the following elements:

name	string, The name of the import.
syncType	string, The type of sync to perform; one of Dbsync, AutoImport, GdocImport, and GdocExport.
source	list, A list containing the following elements: • remoteHostId integer, • credentialId integer,

- additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
- name string,

`destination` list, A list containing the following elements:

- remoteHostId integer,
- credentialId integer,
- additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
- name string,

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`parentId` integer, Parent id to trigger this import from

`id` integer, The ID for the import.

`isOutbound` boolean,

`jobType` string, The job type of this import.

`syncs` array, An array containing the following fields:

- id integer,
- source object,
- destination object,
- advancedOptions object,

`state` string,

`createdAt` string,

`updatedAt` string,

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this import.

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

imports_get_batches *Get details about a batch import*

Description

Get details about a batch import

Usage

```
imports_get_batches(id)
```

Arguments

id	integer required. The ID for the import.

Value

A list containing the following elements:

id	integer, The ID for the import.
schema	string, The destination schema name. This schema must already exist in Redshift.
table	string, The destination table name, without the schema prefix. This table must already exist in Redshift.
remoteHostId	integer, The ID of the destination database host.
state	string, The state of the run; one of "queued", "running", "succeeded", "failed", or "cancelled".
startedAt	string, The time the last run started at.
finishedAt	string, The time the last run completed.
error	string, The error returned by the run, if any.
hidden	boolean, The hidden status of the item.

imports_get_files_csv *Get a CSV Import*

Description

Get a CSV Import

Usage

```
imports_get_files_csv(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for the import.
name	string, The name of the import.
source	list, A list containing the following elements: • fileIds array, The file ID(s) to import, if importing Civis file(s). • storagePath list . A list containing the following elements: – storageHostId integer, The ID of the source storage host. – credentialId integer, The ID of the credentials for the source storage host. – filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).
destination	list, A list containing the following elements: • schema string, The destination schema name. • table string, The destination table name. • remoteHostId integer, The ID of the destination database host. • credentialId integer, The ID of the credentials for the destination database. • primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table.These columns must not contain NULL values. If the import mode is "upsert", this field is required;see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information. • lastModifiedKeys array, A list of the columns indicating a record has been updated.If the destination table does not exist, and the import mode is "upsert", this field is required.
firstRowIsHeader	boolean, A boolean value indicating whether or not the first row of the source file is a header row.
columnDelimiter	string, The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".
escaped	boolean, A boolean value indicating whether or not the source file has quotes escaped with a backslash.Defaults to false.
compression	string, The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".
existingTableRows	string, The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert".Defaults to "fail".

maxErrors integer, The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

tableColumns array, An array containing the following fields:

- name string, The column name.
- sqlType string, The SQL type of the column.

loosenTypes boolean, If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

execution string, In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

redshiftDestinationOptions
list, A list containing the following elements:

- diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
- distkey string, Distkey for this table in Redshift
- sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

hidden boolean, The hidden status of the item.

myPermissionLevel
string, Your permission level on the object. One of "read", "write", or "manage".

---

imports_get_files_csv_runs

Check status of a run

Description

Check status of a run

Usage

```
imports_get_files_csv_runs(id, run_id)
```

Arguments

id integer required. The ID of the CSV Import job.

run_id integer required. The ID of the run.

Value

A list containing the following elements:

id	integer, The ID of the run.
csvImportId	integer, The ID of the CSV Import job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
inputs	list, The inputs of the import at the time of the run.

---

imports_get_files_runs

Check status of a run

Description

Check status of a run

Usage

```
imports_get_files_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Import job.
run_id	integer required. The ID of the run.

Value

A list containing the following elements:

id	integer, The ID of the run.
importId	integer, The ID of the Import job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.

finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
inputs	list, The inputs of the import at the time of the run.

`imports_list` *List Imports*

Description

List Imports

Usage

```
imports_list(
  type = NULL,
  destination = NULL,
  source = NULL,
  status = NULL,
  author = NULL,
  hidden = NULL,
  archived = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

type	string optional. If specified, return imports of these types. It accepts a comma-separated list, possible values are Dbsync, AutoImport, GdocImport, and GdocExport.
destination	string optional. If specified, returns imports with one of these destinations. It accepts a comma-separated list of remote host ids.
source	string optional. If specified, returns imports with one of these sources. It accepts a comma-separated list of remote host ids. 'Dbsync' must be specified for 'type'.
status	string optional. If specified, returns imports with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'failed', 'succeeded', 'idle', 'scheduled'.
author	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
archived	string optional. The archival status of the requested item(s).

limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at, last_run.updated_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

name	string, The name of the import.
syncType	string, The type of sync to perform; one of Dbsync, AutoImport, GdocImport, and GdocExport.
source	list, A list containing the following elements: • remoteHostId integer, • credentialId integer, • additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id. • name string,
destination	list, A list containing the following elements: • remoteHostId integer, • credentialId integer, • additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id. • name string,
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
id	integer, The ID for the import.
isOutbound	boolean,

jobType	string, The job type of this import.
state	string,
createdAt	string,
updatedAt	string,
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
timeZone	string, The time zone of this import.
archived	string, The archival status of the requested item(s).

`imports_list_batches` *List batch imports*

Description

List batch imports

Usage

```
imports_list_batches(
  hidden = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID for the import.
schema	string, The destination schema name. This schema must already exist in Redshift.
table	string, The destination table name, without the schema prefix. This table must already exist in Redshift.
remoteHostId	integer, The ID of the destination database host.
state	string, The state of the run; one of "queued", "running", "succeeded", "failed", or "cancelled".
startedAt	string, The time the last run started at.
finishedAt	string, The time the last run completed.
error	string, The error returned by the run, if any.

---

imports_list_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
imports_list_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

---

imports_list_files_csv_runs

List runs for the given CSV Import job

Description

List runs for the given CSV Import job

Usage

```
imports_list_files_csv_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the CSV Import job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
csvImportId	integer, The ID of the CSV Import job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

---

imports_list_files_csv_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

```
imports_list_files_csv_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the CSV Import job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

imports_list_files_runs *List runs for the given Import job*

Description

List runs for the given Import job

Usage

```
imports_list_files_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the Import job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
importId	integer, The ID of the Import job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

imports_list_files_runs_logs *Get the logs for a run*

Description

Get the logs for a run

Usage

```
imports_list_files_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the Import job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

imports_list_projects *List the projects an Import belongs to*

Description

List the projects an Import belongs to

Usage

```
imports_list_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the Import.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

---

imports_list_runs *Get the run history of this import*

Description

Get the run history of this import

Usage

```
imports_list_runs(id)
```

Arguments

id	integer required.

Value

An array containing the following fields:

id	integer,
state	string,
createdAt	string, The time that the run was queued.
startedAt	string, The time that the run started.
finishedAt	string, The time that the run completed.
error	string, The error message for this run, if present.

---

imports_list_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

```
imports_list_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the import job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

`imports_list_shares` *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
imports_list_shares(id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`imports_patch_files_csv` *Update some attributes of this CSV Import*

Description

Update some attributes of this CSV Import

Usage

```
imports_patch_files_csv(
  id,
  name = NULL,
  source = NULL,
  destination = NULL,
  first_row_is_header = NULL,
  column_delimiter = NULL,
  escaped = NULL,
  compression = NULL,
  existing_table_rows = NULL,
  max_errors = NULL,
  table_columns = NULL,
  loosen_types = NULL,
  execution = NULL,
  redshift_destination_options = NULL
)
```

Arguments

`id`	integer required. The ID for the import.
`name`	string optional. The name of the import.
`source`	list optional. A list containing the following elements: • fileIds array, The file ID(s) to import, if importing Civis file(s). • storagePath list . A list containing the following elements: – storageHostId integer, The ID of the source storage host. – credentialId integer, The ID of the credentials for the source storage host. – filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).
`destination`	list optional. A list containing the following elements: • schema string, The destination schema name. • table string, The destination table name.

- remoteHostId integer, The ID of the destination database host.
- credentialId integer, The ID of the credentials for the destination database.
- primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table.These columns must not contain NULL values. If the import mode is "upsert", this field is required;see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
- lastModifiedKeys array, A list of the columns indicating a record has been updated.If the destination table does not exist, and the import mode is "upsert", this field is required.

`first_row_is_header`
: boolean optional. A boolean value indicating whether or not the first row of the source file is a header row.

`column_delimiter`
: string optional. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`escaped`
: boolean optional. A boolean value indicating whether or not the source file has quotes escaped with a backslash.Defaults to false.

`compression`
: string optional. The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".

`existing_table_rows`
: string optional. The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert".Defaults to "fail".

`max_errors`
: integer optional. The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

`table_columns`
: array optional. An array containing the following fields:

  - name string, The column name.
  - sqlType string, The SQL type of the column.

`loosen_types`
: boolean optional. If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

`execution`
: string optional. In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

`redshift_destination_options`
: list optional. A list containing the following elements:

  - diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
  - distkey string, Distkey for this table in Redshift
  - sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

Value

A list containing the following elements:

`id` integer, The ID for the import.

`name` string, The name of the import.

`source` list, A list containing the following elements:

- fileIds array, The file ID(s) to import, if importing Civis file(s).
- storagePath list . A list containing the following elements:
  - storageHostId integer, The ID of the source storage host.
  - credentialId integer, The ID of the credentials for the source storage host.
  - filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).

`destination` list, A list containing the following elements:

- schema string, The destination schema name.
- table string, The destination table name.
- remoteHostId integer, The ID of the destination database host.
- credentialId integer, The ID of the credentials for the destination database.
- primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table.These columns must not contain NULL values. If the import mode is "upsert", this field is required;see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
- lastModifiedKeys array, A list of the columns indicating a record has been updated.If the destination table does not exist, and the import mode is "upsert", this field is required.

`firstRowIsHeader` boolean, A boolean value indicating whether or not the first row of the source file is a header row.

`columnDelimiter` string, The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`escaped` boolean, A boolean value indicating whether or not the source file has quotes escaped with a backslash.Defaults to false.

`compression` string, The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".

`existingTableRows` string, The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert".Defaults to "fail".

`maxErrors` integer, The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

`tableColumns` array, An array containing the following fields:

- name string, The column name.
- sqlType string, The SQL type of the column.

`loosenTypes` boolean, If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

`execution` string, In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

`redshiftDestinationOptions`
list, A list containing the following elements:

- diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
- distkey string, Distkey for this table in Redshift
- sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

`hidden` boolean, The hidden status of the item.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`imports_post` *Create a new import configuration*

Description

Create a new import configuration

Usage

```
imports_post(
  name,
  sync_type,
  is_outbound,
  source = NULL,
  destination = NULL,
  schedule = NULL,
  notifications = NULL,
  parent_id = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  hidden = NULL
)
```

Arguments

name	string required. The name of the import.
sync_type	string required. The type of sync to perform; one of Dbsync, AutoImport, GdocImport, and GdocExport.
is_outbound	boolean required.
source	list optional. A list containing the following elements: • remoteHostId integer, • credentialId integer, • additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
destination	list optional. A list containing the following elements: • remoteHostId integer, • credentialId integer, • additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully. • successEmailFromName string, Name from which success emails are sent; defaults to "Civis." • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job. • failureEmailAddresses array, Addresses to notify by e-mail when the job fails. • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.

- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`parent_id` integer optional. Parent id to trigger this import from

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this import.

`hidden` boolean optional. The hidden status of the item.

Value

A list containing the following elements:

`name` string, The name of the import.

`syncType` string, The type of sync to perform; one of Dbsync, AutoImport, GdocImport, and GdocExport.

`source` list, A list containing the following elements:

- remoteHostId integer,
- credentialId integer,
- additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
- name string,

`destination` list, A list containing the following elements:

- remoteHostId integer,
- credentialId integer,
- additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
- name string,

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion

- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`parentId` integer, Parent id to trigger this import from

`id` integer, The ID for the import.

`isOutbound` boolean,

`jobType` string, The job type of this import.

`syncs` array, An array containing the following fields:

- id integer,
- source object,
- destination object,
- advancedOptions object,

`state` string,

`createdAt` string,

`updatedAt` string,

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.

- online boolean, Whether this user is online.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this import.

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`imports_post_batches` *Upload multiple files to Civis*

Description

Upload multiple files to Civis

Usage

```
imports_post_batches(
  file_ids,
  schema,
  table,
  remote_host_id,
  credential_id,
  column_delimiter = NULL,
  first_row_is_header = NULL,
  compression = NULL,
  hidden = NULL
)
```

Arguments

`file_ids`	array required. The file IDs for the import.
`schema`	string required. The destination schema name. This schema must already exist in Redshift.
`table`	string required. The destination table name, without the schema prefix. This table must already exist in Redshift.
`remote_host_id`	integer required. The ID of the destination database host.

`credential_id` integer required. The ID of the credentials to be used when performing the database import.

`column_delimiter`
string optional. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". If unspecified, defaults to "comma".

`first_row_is_header`
boolean optional. A boolean value indicating whether or not the first row is a header row. If unspecified, defaults to false.

`compression` string optional. The type of compression. Valid arguments are "gzip", "zip", and "none". If unspecified, defaults to "gzip".

`hidden` boolean optional. The hidden status of the item.

Value

A list containing the following elements:

`id` integer, The ID for the import.

`schema` string, The destination schema name. This schema must already exist in Redshift.

`table` string, The destination table name, without the schema prefix. This table must already exist in Redshift.

`remoteHostId` integer, The ID of the destination database host.

`state` string, The state of the run; one of "queued", "running", "succeeded", "failed", or "cancelled".

`startedAt` string, The time the last run started at.

`finishedAt` string, The time the last run completed.

`error` string, The error returned by the run, if any.

`hidden` boolean, The hidden status of the item.

`imports_post_cancel` *Cancel a run*

Description

Cancel a run

Usage

```
imports_post_cancel(id)
```

Arguments

`id` integer required. The ID of the job.

Value

A list containing the following elements:

id	integer, The ID of the run.
state	string, The state of the run, one of 'queued', 'running' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.

`imports_post_files` *Initate an import of a tabular file into the platform*

Description

Initate an import of a tabular file into the platform

Usage

```
imports_post_files(
  schema,
  name,
  remote_host_id,
  credential_id,
  max_errors = NULL,
  existing_table_rows = NULL,
  diststyle = NULL,
  distkey = NULL,
  sortkey1 = NULL,
  sortkey2 = NULL,
  column_delimiter = NULL,
  first_row_is_header = NULL,
  multipart = NULL,
  escaped = NULL,
  hidden = NULL
)
```

Arguments

schema	string required. The schema of the destination table.
name	string required. The name of the destination table.
remote_host_id	integer required. The id of the destination database host.
credential_id	integer required. The id of the credentials to be used when performing the database import.
max_errors	integer optional. The maximum number of rows with errors to remove from the import before failing.

existing_table_rows
: string optional. The behaviour if a table with the requested name already exists. One of "fail", "truncate", "append", or "drop".Defaults to "fail".

diststyle
: string optional. The diststyle to use for the table. One of "even", "all", or "key".

distkey
: string optional. The column to use as the distkey for the table.

sortkey1
: string optional. The column to use as the sort key for the table.

sortkey2
: string optional. The second column in a compound sortkey for the table.

column_delimiter
: string optional. The column delimiter of the file. If column_delimiter is null or omitted, it will be auto-detected. Valid arguments are "comma", "tab", and "pipe".

first_row_is_header
: boolean optional. A boolean value indicating whether or not the first row is a header row. If first_row_is_header is null or omitted, it will be auto-detected.

multipart
: boolean optional. If true, the upload URI will require a 'multipart/form-data' POST request. Defaults to false.

escaped
: boolean optional. If true, escape quotes with a backslash; otherwise, escape quotes by double-quoting. Defaults to false.

hidden
: boolean optional. The hidden status of the item.

Value

A list containing the following elements:

id
: integer, The id of the import.

uploadUri
: string, The URI which may be used to upload a tabular file for import. You must use this URI to upload the file you wish imported and then inform the Civis API when your upload is complete using the URI given by the runUri field of this response.

runUri
: string, The URI to POST to once the file upload is complete. After uploading the file using the URI given in the uploadUri attribute of the response, POST to this URI to initiate the import of your uploaded file into the platform.

uploadFields
: list, If multipart was set to true, these fields should be included in the multipart upload.

---

imports_post_files_csv

Create a CSV Import

Description

Create a CSV Import

Usage

```
imports_post_files_csv(
  source,
  destination,
  first_row_is_header,
  name = NULL,
  column_delimiter = NULL,
  escaped = NULL,
  compression = NULL,
  existing_table_rows = NULL,
  max_errors = NULL,
  table_columns = NULL,
  loosen_types = NULL,
  execution = NULL,
  redshift_destination_options = NULL,
  hidden = NULL
)
```

Arguments

`source` list required. A list containing the following elements:

- fileIds array, The file ID(s) to import, if importing Civis file(s).
- storagePath list . A list containing the following elements:
  - storageHostId integer, The ID of the source storage host.
  - credentialId integer, The ID of the credentials for the source storage host.
  - filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).

`destination` list required. A list containing the following elements:

- schema string, The destination schema name.
- table string, The destination table name.
- remoteHostId integer, The ID of the destination database host.
- credentialId integer, The ID of the credentials for the destination database.
- primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table.These columns must not contain NULL values. If the import mode is "upsert", this field is required;see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
- lastModifiedKeys array, A list of the columns indicating a record has been updated.If the destination table does not exist, and the import mode is "upsert", this field is required.

`first_row_is_header` boolean required. A boolean value indicating whether or not the first row of the source file is a header row.

`name` string optional. The name of the import.

`column_delimiter`
string optional. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`escaped` boolean optional. A boolean value indicating whether or not the source file has quotes escaped with a backslash.Defaults to false.

`compression` string optional. The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".

`existing_table_rows`
string optional. The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert".Defaults to "fail".

`max_errors` integer optional. The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

`table_columns` array optional. An array containing the following fields:

- name string, The column name.
- sqlType string, The SQL type of the column.

`loosen_types` boolean optional. If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

`execution` string optional. In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

`redshift_destination_options`
list optional. A list containing the following elements:

- diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
- distkey string, Distkey for this table in Redshift
- sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

`hidden` boolean optional. The hidden status of the item.

Value

A list containing the following elements:

`id` integer, The ID for the import.

`name` string, The name of the import.

`source` list, A list containing the following elements:

- fileIds array, The file ID(s) to import, if importing Civis file(s).
- storagePath list . A list containing the following elements:

- storageHostId integer, The ID of the source storage host.
- credentialId integer, The ID of the credentials for the source storage host.
- filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).

`destination` list, A list containing the following elements:

- schema string, The destination schema name.
- table string, The destination table name.
- remoteHostId integer, The ID of the destination database host.
- credentialId integer, The ID of the credentials for the destination database.
- primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table.These columns must not contain NULL values. If the import mode is "upsert", this field is required;see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
- lastModifiedKeys array, A list of the columns indicating a record has been updated.If the destination table does not exist, and the import mode is "upsert", this field is required.

`firstRowIsHeader` boolean, A boolean value indicating whether or not the first row of the source file is a header row.

`columnDelimiter` string, The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`escaped` boolean, A boolean value indicating whether or not the source file has quotes escaped with a backslash.Defaults to false.

`compression` string, The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".

`existingTableRows` string, The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert".Defaults to "fail".

`maxErrors` integer, The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

`tableColumns` array, An array containing the following fields:

- name string, The column name.
- sqlType string, The SQL type of the column.

`loosenTypes` boolean, If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

`execution` string, In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at

which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

`redshiftDestinationOptions`
list, A list containing the following elements:

- diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
- distkey string, Distkey for this table in Redshift
- sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

`hidden` boolean, The hidden status of the item.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`imports_post_files_csv_runs` *Start a run*

Description

Start a run

Usage

```
imports_post_files_csv_runs(id)
```

Arguments

`id`	integer required. The ID of the CSV Import job.

Value

A list containing the following elements:

`id`	integer, The ID of the run.
`csvImportId`	integer, The ID of the CSV Import job.
`state`	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
`isCancelRequested`	boolean, True if run cancel requested, else false.
`createdAt`	string, The time the run was created.
`startedAt`	string, The time the run started at.
`finishedAt`	string, The time the run completed.
`error`	string, The error, if any, returned by the run.
`inputs`	list, The inputs of the import at the time of the run.

imports_post_files_runs *Start a run*

Description

Start a run

Usage

```
imports_post_files_runs(id)
```

Arguments

id	integer required. The ID of the Import job.

Value

A list containing the following elements:

id	integer, The ID of the run.
importId	integer, The ID of the Import job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
inputs	list, The inputs of the import at the time of the run.

imports_post_runs *Run an import*

Description

Run an import

Usage

```
imports_post_runs(id)
```

Arguments

id	integer required. The ID of the import to run.

Value

A list containing the following elements:

runId	integer, The ID of the new run triggered.

---

imports_post_syncs *Create a sync*

Description

Create a sync

Usage

```
imports_post_syncs(id, source, destination, advanced_options = NULL)
```

Arguments

id	integer required.
source	list required. A list containing the following elements: • path string, The path of the dataset to sync from; for a database source, schema.tablename. If you are doing a Google Sheet export, this can be blank. This is a legacy parameter, it is recommended you use one of the following: databaseTable, file, googleWorksheet • databaseTable list . A list containing the following elements: – schema string, The database schema name. – table string, The database table name. – useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used. • file list . • googleWorksheet list . A list containing the following elements: – spreadsheet string, The spreadsheet document name. – spreadsheetId string, The spreadsheet document id. – worksheet string, The worksheet tab name. – worksheetId integer, The worksheet tab id. • salesforce list . A list containing the following elements: – objectName string, This parameter is deprecated
destination	list required. A list containing the following elements:

- path string, The schema.tablename to sync to. If you are doing a Google Sheet export, this is the spreadsheet and sheet name separated by a period. i.e. if you have a spreadsheet named "MySpreadsheet" and a sheet called "Sheet1" this field would be "MySpreadsheet.Sheet1". This is a legacy parameter, it is recommended you use one of the following: databaseTable, googleWorksheet
- databaseTable list . A list containing the following elements:
  - schema string, The database schema name.
  - table string, The database table name.
  - useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
- googleWorksheet list . A list containing the following elements:
  - spreadsheet string, The spreadsheet document name.
  - spreadsheetId string, The spreadsheet document id.
  - worksheet string, The worksheet tab name.
  - worksheetId integer, The worksheet tab id.

`advanced_options`

list optional. A list containing the following elements:

- maxErrors integer, For Google Doc and Auto Imports. The maximum number of errors that can occur without the job failing.
- existingTableRows string, For Google Doc and Auto Imports. The behavior if a table with the requested name already exists. One of "fail", "truncate", "append", or "drop". Defaults to "fail".
- firstRowIsHeader boolean, For Google Doc and Auto Imports. A boolean value indicating whether or not the first row is a header row.
- diststyle string, For Auto Imports. The diststyle to use for a Redshift table.
- distkey string, For Auto Imports. The distkey to use for a Redshift table.
- sortkey1 string, For Auto Imports. The first sortkey to use for a Redshift table.
- sortkey2 string, For Auto Imports. The second sortkey to use for a Redshift table.
- columnDelimiter string, For Auto Imports. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". If column_delimiter is null or omitted, it will be auto-detected.
- columnOverrides object, For Auto Imports. Hash used for overriding auto-detected names and types, with keys being the index of the column being overridden.
- escaped boolean, For Auto Imports. If true, escape quotes with a backslash; otherwise, escape quotes by double-quoting. Defaults to false.
- identityColumn string, For DB Syncs. The column or columns to use as primary key for incremental syncs. Should be a unique identifier. If blank, primary key columns will be auto-detected. If more than one identity column is specified, an identity column must be specified for each table. We recommend the primary key be a sequential data type such as an integer, double, timestamp, date, or float. If using a primary key that is a string data

type, we recommend having a last_modified_column to ensure all data is synced to the destination table.

- lastModifiedColumn string, For DB Syncs. The column to use to detect recently modified data for incremental syncs. Defaults to "Auto-Detect", which will use the first column it finds containing either "modif" or "update" in the name. When specified, only rows where last_modified_column in the source >= last_modified_column in the destination will be synced.
- rowChunkSize integer, For DB Syncs. If specified, will split the sync into chunks of this size.
- wipeDestinationTable boolean, For DB Syncs. If true, will perform a full table refresh.
- truncateLongLines boolean, For DB Syncs to Redshift. When true, truncates column data to fit the column specification.
- invalidCharReplacement string, For DB Syncs to Redshift. If specified, will replace each invalid UTF-8 character with this character. Must be a single ASCII character.
- verifyTableRowCounts boolean, For DB Syncs. When true, an error will be raised if the destination table does not have the same number of rows as the source table after the sync.
- partitionColumnName string, This parameter is deprecated
- partitionSchemaName string, This parameter is deprecated
- partitionTableName string, This parameter is deprecated
- partitionTablePartitionColumnMinName string, This parameter is deprecated
- partitionTablePartitionColumnMaxName string, This parameter is deprecated
- mysqlCatalogMatchesSchema boolean, This attribute is no longer available; defaults to true but cannot be used.
- chunkingMethod string, This parameter is deprecated
- exportAction string, For Google Doc Exports. The kind of export action you want to have the export execute. Set to "newsprsht" if you want a new worksheet inside a new spreadsheet. Set to "newwksht" if you want a new worksheet inside an existing spreadsheet. Set to "updatewksht" if you want to overwrite an existing worksheet inside an existing spreadsheet. Set to "appendwksht" if you want to append to the end of an existing worksheet inside an existing spreadsheet. Default is set to "newsprsht"
- sqlQuery string, For Google Doc Exports. The SQL query for the export.
- contactLists string, This parameter is deprecated
- soqlQuery string, This parameter is deprecated
- includeDeletedRecords boolean, This parameter is deprecated

Value

A list containing the following elements:

`id` integer,

`source` list, A list containing the following elements:

- id integer, The ID of the table or file, if available.
- path string, The path of the dataset to sync from; for a database source, schema.tablename. If you are doing a Google Sheet export, this can be blank. This is a legacy parameter, it is recommended you use one of the following: databaseTable, file, googleWorksheet
- databaseTable list . A list containing the following elements:
  - schema string, The database schema name.
  - table string, The database table name.
  - useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
- file list . A list containing the following elements:
  - id integer, The file id.
- googleWorksheet list . A list containing the following elements:
  - spreadsheet string, The spreadsheet document name.
  - spreadsheetId string, The spreadsheet document id.
  - worksheet string, The worksheet tab name.
  - worksheetId integer, The worksheet tab id.
- salesforce list . A list containing the following elements:
  - objectName string, This parameter is deprecated

`destination` list, A list containing the following elements:

- path string, The schema.tablename to sync to. If you are doing a Google Sheet export, this is the spreadsheet and sheet name separated by a period. i.e. if you have a spreadsheet named "MySpreadsheet" and a sheet called "Sheet1" this field would be "MySpreadsheet.Sheet1". This is a legacy parameter, it is recommended you use one of the following: databaseTable, googleWorksheet
- databaseTable list . A list containing the following elements:
  - schema string, The database schema name.
  - table string, The database table name.
  - useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
- googleWorksheet list . A list containing the following elements:
  - spreadsheet string, The spreadsheet document name.
  - spreadsheetId string, The spreadsheet document id.
  - worksheet string, The worksheet tab name.
  - worksheetId integer, The worksheet tab id.

`advancedOptions`

list, A list containing the following elements:

- maxErrors integer, For Google Doc and Auto Imports. The maximum number of errors that can occur without the job failing.
- existingTableRows string, For Google Doc and Auto Imports. The behavior if a table with the requested name already exists. One of "fail", "truncate", "append", or "drop". Defaults to "fail".

- firstRowIsHeader boolean, For Google Doc and Auto Imports. A boolean value indicating whether or not the first row is a header row.
- diststyle string, For Auto Imports. The diststyle to use for a Redshift table.
- distkey string, For Auto Imports. The distkey to use for a Redshift table.
- sortkey1 string, For Auto Imports. The first sortkey to use for a Redshift table.
- sortkey2 string, For Auto Imports. The second sortkey to use for a Redshift table.
- columnDelimiter string, For Auto Imports. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". If column_delimiter is null or omitted, it will be auto-detected.
- columnOverrides object, For Auto Imports. Hash used for overriding auto-detected names and types, with keys being the index of the column being overridden.
- escaped boolean, For Auto Imports. If true, escape quotes with a backslash; otherwise, escape quotes by double-quoting. Defaults to false.
- identityColumn string, For DB Syncs. The column or columns to use as primary key for incremental syncs. Should be a unique identifier. If blank, primary key columns will be auto-detected. If more than one identity column is specified, an identity column must be specified for each table. We recommend the primary key be a sequential data type such as an integer, double, timestamp, date, or float. If using a primary key that is a string data type, we recommend having a last_modified_column to ensure all data is synced to the destination table.
- lastModifiedColumn string, For DB Syncs. The column to use to detect recently modified data for incremental syncs. Defaults to "Auto-Detect", which will use the first column it finds containing either "modif" or "update" in the name. When specified, only rows where last_modified_column in the source >= last_modified_column in the destination will be synced.
- rowChunkSize integer, For DB Syncs. If specified, will split the sync into chunks of this size.
- wipeDestinationTable boolean, For DB Syncs. If true, will perform a full table refresh.
- truncateLongLines boolean, For DB Syncs to Redshift. When true, truncates column data to fit the column specification.
- invalidCharReplacement string, For DB Syncs to Redshift. If specified, will replace each invalid UTF-8 character with this character. Must be a single ASCII character.
- verifyTableRowCounts boolean, For DB Syncs. When true, an error will be raised if the destination table does not have the same number of rows as the source table after the sync.
- partitionColumnName string, This parameter is deprecated
- partitionSchemaName string, This parameter is deprecated
- partitionTableName string, This parameter is deprecated
- partitionTablePartitionColumnMinName string, This parameter is deprecated

- partitionTablePartitionColumnMaxName string, This parameter is deprecated
- mysqlCatalogMatchesSchema boolean, This attribute is no longer available; defaults to true but cannot be used.
- chunkingMethod string, This parameter is deprecated
- exportAction string, For Google Doc Exports. The kind of export action you want to have the export execute. Set to "newsprsht" if you want a new worksheet inside a new spreadsheet. Set to "newwksht" if you want a new worksheet inside an existing spreadsheet. Set to "updatewksht" if you want to overwrite an existing worksheet inside an existing spreadsheet. Set to "appendwksht" if you want to append to the end of an existing worksheet inside an existing spreadsheet. Default is set to "newsprsht"
- sqlQuery string, For Google Doc Exports. The SQL query for the export.
- contactLists string, This parameter is deprecated
- soqlQuery string, This parameter is deprecated
- includeDeletedRecords boolean, This parameter is deprecated

---

`imports_put` *Update an import*

Description

Update an import

Usage

```
imports_put(
  id,
  name,
  sync_type,
  is_outbound,
  source = NULL,
  destination = NULL,
  schedule = NULL,
  notifications = NULL,
  parent_id = NULL,
  next_run_at = NULL,
  time_zone = NULL
)
```

Arguments

`id`	integer required. The ID for the import.
`name`	string required. The name of the import.
`sync_type`	string required. The type of sync to perform; one of Dbsync, AutoImport, GdocImport, and GdocExport.

`is_outbound` boolean required.

`source` list optional. A list containing the following elements:

- remoteHostId integer,
- credentialId integer,
- additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.

`destination` list optional. A list containing the following elements:

- remoteHostId integer,
- credentialId integer,
- additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.

`schedule` list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

parent_id	integer optional. Parent id to trigger this import from
next_run_at	string optional. The time of the next scheduled run.
time_zone	string optional. The time zone of this import.

Value

A list containing the following elements:

name	string, The name of the import.
syncType	string, The type of sync to perform; one of Dbsync, AutoImport, GdocImport, and GdocExport.
source	list, A list containing the following elements: • remoteHostId integer, • credentialId integer, • additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id. • name string,
destination	list, A list containing the following elements: • remoteHostId integer, • credentialId integer, • additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id. • name string,
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list, A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.

- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`parentId` integer, Parent id to trigger this import from

`id` integer, The ID for the import.

`isOutbound` boolean,

`jobType` string, The job type of this import.

`syncs` array, An array containing the following fields:

- id integer,
- source object,
- destination object,
- advancedOptions object,

`state` string,

`createdAt` string,

`updatedAt` string,

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.

- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this import.

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`imports_put_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
imports_put_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`name` string, The name of the import.

`syncType` string, The type of sync to perform; one of Dbsync, AutoImport, GdocImport, and GdocExport.

`source` list, A list containing the following elements:

- remoteHostId integer,
- credentialId integer,
- additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
- name string,

`destination` list, A list containing the following elements:

- remoteHostId integer,
- credentialId integer,

- additionalCredentials array, Array that holds additional credentials used for specific imports. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
- name string,

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`parentId` integer, Parent id to trigger this import from

`id` integer, The ID for the import.

`isOutbound` boolean,

`jobType` string, The job type of this import.

`syncs` array, An array containing the following fields:

- id integer,
- source object,
- destination object,
- advancedOptions object,

state	string,
createdAt	string,
updatedAt	string,
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
runningAs	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
nextRunAt	string, The time of the next scheduled run.
timeZone	string, The time zone of this import.
hidden	boolean, The hidden status of the item.
archived	string, The archival status of the requested item(s).
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".

`imports_put_files_csv` *Replace all attributes of this CSV Import*

Description

Replace all attributes of this CSV Import

Usage

```
imports_put_files_csv(
  id,
  source,
  destination,
  first_row_is_header,
  name = NULL,
  column_delimiter = NULL,
  escaped = NULL,
  compression = NULL,
  existing_table_rows = NULL,
  max_errors = NULL,
  table_columns = NULL,
  loosen_types = NULL,
  execution = NULL,
  redshift_destination_options = NULL
)
```

Arguments

id	integer required. The ID for the import.
source	list required. A list containing the following elements: • fileIds array, The file ID(s) to import, if importing Civis file(s). • storagePath list . A list containing the following elements: – storageHostId integer, The ID of the source storage host. – credentialId integer, The ID of the credentials for the source storage host. – filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).
destination	list required. A list containing the following elements: • schema string, The destination schema name. • table string, The destination table name. • remoteHostId integer, The ID of the destination database host. • credentialId integer, The ID of the credentials for the destination database. • primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table.These columns must not contain NULL values. If the import mode is "upsert", this field is required;see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information. • lastModifiedKeys array, A list of the columns indicating a record has been updated.If the destination table does not exist, and the import mode is "upsert", this field is required.

`first_row_is_header`
: boolean required. A boolean value indicating whether or not the first row of the source file is a header row.

`name`
: string optional. The name of the import.

`column_delimiter`
: string optional. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`escaped`
: boolean optional. A boolean value indicating whether or not the source file has quotes escaped with a backslash.Defaults to false.

`compression`
: string optional. The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".

`existing_table_rows`
: string optional. The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert".Defaults to "fail".

`max_errors`
: integer optional. The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

`table_columns`
: array optional. An array containing the following fields:
  - name string, The column name.
  - sqlType string, The SQL type of the column.

`loosen_types`
: boolean optional. If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

`execution`
: string optional. In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

`redshift_destination_options`
: list optional. A list containing the following elements:
  - diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
  - distkey string, Distkey for this table in Redshift
  - sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

Value

A list containing the following elements:

`id`
: integer, The ID for the import.

`name`
: string, The name of the import.

`source`
: list, A list containing the following elements:
  - fileIds array, The file ID(s) to import, if importing Civis file(s).

- storagePath list . A list containing the following elements:
  - storageHostId integer, The ID of the source storage host.
  - credentialId integer, The ID of the credentials for the source storage host.
  - filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).

`destination` list, A list containing the following elements:

- schema string, The destination schema name.
- table string, The destination table name.
- remoteHostId integer, The ID of the destination database host.
- credentialId integer, The ID of the credentials for the destination database.
- primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table.These columns must not contain NULL values. If the import mode is "upsert", this field is required;see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
- lastModifiedKeys array, A list of the columns indicating a record has been updated.If the destination table does not exist, and the import mode is "upsert", this field is required.

`firstRowIsHeader` boolean, A boolean value indicating whether or not the first row of the source file is a header row.

`columnDelimiter` string, The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`escaped` boolean, A boolean value indicating whether or not the source file has quotes escaped with a backslash.Defaults to false.

`compression` string, The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".

`existingTableRows` string, The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert".Defaults to "fail".

`maxErrors` integer, The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

`tableColumns` array, An array containing the following fields:

- name string, The column name.
- sqlType string, The SQL type of the column.

`loosenTypes` boolean, If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

`execution` string, In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate",

the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

`redshiftDestinationOptions`
: list, A list containing the following elements:
  - diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
  - distkey string, Distkey for this table in Redshift
  - sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

`hidden`
: boolean, The hidden status of the item.

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`imports_put_files_csv_archive`
Update the archive status of this object

Description

Update the archive status of this object

Usage

```
imports_put_files_csv_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id`
: integer, The ID for the import.

`name`
: string, The name of the import.

`source`
: list, A list containing the following elements:
  - fileIds array, The file ID(s) to import, if importing Civis file(s).
  - storagePath list . A list containing the following elements:
    - storageHostId integer, The ID of the source storage host.
    - credentialId integer, The ID of the credentials for the source storage host.

- filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).

`destination` list, A list containing the following elements:

- schema string, The destination schema name.
- table string, The destination table name.
- remoteHostId integer, The ID of the destination database host.
- credentialId integer, The ID of the credentials for the destination database.
- primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table.These columns must not contain NULL values. If the import mode is "upsert", this field is required;see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
- lastModifiedKeys array, A list of the columns indicating a record has been updated.If the destination table does not exist, and the import mode is "upsert", this field is required.

`firstRowIsHeader` boolean, A boolean value indicating whether or not the first row of the source file is a header row.

`columnDelimiter` string, The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

`escaped` boolean, A boolean value indicating whether or not the source file has quotes escaped with a backslash.Defaults to false.

`compression` string, The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".

`existingTableRows` string, The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert".Defaults to "fail".

`maxErrors` integer, The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

`tableColumns` array, An array containing the following fields:

- name string, The column name.
- sqlType string, The SQL type of the column.

`loosenTypes` boolean, If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

`execution` string, In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

redshiftDestinationOptions
: list, A list containing the following elements:
  - diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
  - distkey string, Distkey for this table in Redshift
  - sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

hidden
: boolean, The hidden status of the item.

myPermissionLevel
: string, Your permission level on the object. One of "read", "write", or "manage".

---

imports_put_projects *Add an Import to a project*

Description

Add an Import to a project

Usage

```
imports_put_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Import.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

---

imports_put_shares_groups *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
imports_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
`totalGroupShares`	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`imports_put_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
imports_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_ids`	array required. An array of one or more user IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`imports_put_syncs` *Update a sync*

Description

Update a sync

Usage

```
imports_put_syncs(id, sync_id, source, destination, advanced_options = NULL)
```

Arguments

id
: integer required. The ID of the import to fetch.

sync_id
: integer required. The ID of the sync to fetch.

source
: list required. A list containing the following elements:

- path string, The path of the dataset to sync from; for a database source, schema.tablename. If you are doing a Google Sheet export, this can be blank. This is a legacy parameter, it is recommended you use one of the following: databaseTable, file, googleWorksheet
- databaseTable list . A list containing the following elements:
  - schema string, The database schema name.
  - table string, The database table name.
  - useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
- file list .
- googleWorksheet list . A list containing the following elements:
  - spreadsheet string, The spreadsheet document name.
  - spreadsheetId string, The spreadsheet document id.
  - worksheet string, The worksheet tab name.
  - worksheetId integer, The worksheet tab id.
- salesforce list . A list containing the following elements:
  - objectName string, This parameter is deprecated

destination
: list required. A list containing the following elements:

- path string, The schema.tablename to sync to. If you are doing a Google Sheet export, this is the spreadsheet and sheet name separated by a period. i.e. if you have a spreadsheet named "MySpreadsheet" and a sheet called "Sheet1" this field would be "MySpreadsheet.Sheet1". This is a legacy parameter, it is recommended you use one of the following: databaseTable, googleWorksheet

- databaseTable list . A list containing the following elements:
  - schema string, The database schema name.
  - table string, The database table name.
  - useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
- googleWorksheet list . A list containing the following elements:
  - spreadsheet string, The spreadsheet document name.
  - spreadsheetId string, The spreadsheet document id.
  - worksheet string, The worksheet tab name.
  - worksheetId integer, The worksheet tab id.

`advanced_options`

list optional. A list containing the following elements:

- maxErrors integer, For Google Doc and Auto Imports. The maximum number of errors that can occur without the job failing.
- existingTableRows string, For Google Doc and Auto Imports. The behavior if a table with the requested name already exists. One of "fail", "truncate", "append", or "drop". Defaults to "fail".
- firstRowIsHeader boolean, For Google Doc and Auto Imports. A boolean value indicating whether or not the first row is a header row.
- diststyle string, For Auto Imports. The diststyle to use for a Redshift table.
- distkey string, For Auto Imports. The distkey to use for a Redshift table.
- sortkey1 string, For Auto Imports. The first sortkey to use for a Redshift table.
- sortkey2 string, For Auto Imports. The second sortkey to use for a Redshift table.
- columnDelimiter string, For Auto Imports. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". If column_delimiter is null or omitted, it will be auto-detected.
- columnOverrides object, For Auto Imports. Hash used for overriding auto-detected names and types, with keys being the index of the column being overridden.
- escaped boolean, For Auto Imports. If true, escape quotes with a backslash; otherwise, escape quotes by double-quoting. Defaults to false.
- identityColumn string, For DB Syncs. The column or columns to use as primary key for incremental syncs. Should be a unique identifier. If blank, primary key columns will be auto-detected. If more than one identity column is specified, an identity column must be specified for each table. We recommend the primary key be a sequential data type such as an integer, double, timestamp, date, or float. If using a primary key that is a string data type, we recommend having a last_modified_column to ensure all data is synced to the destination table.
- lastModifiedColumn string, For DB Syncs. The column to use to detect recently modified data for incremental syncs. Defaults to "Auto-Detect", which will use the first column it finds containing either "modif" or "update" in the name. When specified, only rows where last_modified_column in the source >= last_modified_column in the destination will be synced.

- rowChunkSize integer, For DB Syncs. If specified, will split the sync into chunks of this size.
- wipeDestinationTable boolean, For DB Syncs. If true, will perform a full table refresh.
- truncateLongLines boolean, For DB Syncs to Redshift. When true, truncates column data to fit the column specification.
- invalidCharReplacement string, For DB Syncs to Redshift. If specified, will replace each invalid UTF-8 character with this character. Must be a single ASCII character.
- verifyTableRowCounts boolean, For DB Syncs. When true, an error will be raised if the destination table does not have the same number of rows as the source table after the sync.
- partitionColumnName string, This parameter is deprecated
- partitionSchemaName string, This parameter is deprecated
- partitionTableName string, This parameter is deprecated
- partitionTablePartitionColumnMinName string, This parameter is deprecated
- partitionTablePartitionColumnMaxName string, This parameter is deprecated
- mysqlCatalogMatchesSchema boolean, This attribute is no longer available; defaults to true but cannot be used.
- chunkingMethod string, This parameter is deprecated
- exportAction string, For Google Doc Exports. The kind of export action you want to have the export execute. Set to "newsprsht" if you want a new worksheet inside a new spreadsheet. Set to "newwksht" if you want a new worksheet inside an existing spreadsheet. Set to "updatewksht" if you want to overwrite an existing worksheet inside an existing spreadsheet. Set to "appendwksht" if you want to append to the end of an existing worksheet inside an existing spreadsheet. Default is set to "newsprsht"
- sqlQuery string, For Google Doc Exports. The SQL query for the export.
- contactLists string, This parameter is deprecated
- soqlQuery string, This parameter is deprecated
- includeDeletedRecords boolean, This parameter is deprecated

Value

A list containing the following elements:

`id` integer,

`source` list, A list containing the following elements:

- id integer, The ID of the table or file, if available.
- path string, The path of the dataset to sync from; for a database source, schema.tablename. If you are doing a Google Sheet export, this can be blank. This is a legacy parameter, it is recommended you use one of the following: databaseTable, file, googleWorksheet
- databaseTable list . A list containing the following elements:

  - schema string, The database schema name.
  - table string, The database table name.
  - useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
- file list . A list containing the following elements:
  - id integer, The file id.
- googleWorksheet list . A list containing the following elements:
  - spreadsheet string, The spreadsheet document name.
  - spreadsheetId string, The spreadsheet document id.
  - worksheet string, The worksheet tab name.
  - worksheetId integer, The worksheet tab id.
- salesforce list . A list containing the following elements:
  - objectName string, This parameter is deprecated

`destination` list, A list containing the following elements:

- path string, The schema.tablename to sync to. If you are doing a Google Sheet export, this is the spreadsheet and sheet name separated by a period. i.e. if you have a spreadsheet named "MySpreadsheet" and a sheet called "Sheet1" this field would be "MySpreadsheet.Sheet1". This is a legacy parameter, it is recommended you use one of the following: databaseTable, googleWorksheet
- databaseTable list . A list containing the following elements:
  - schema string, The database schema name.
  - table string, The database table name.
  - useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
- googleWorksheet list . A list containing the following elements:
  - spreadsheet string, The spreadsheet document name.
  - spreadsheetId string, The spreadsheet document id.
  - worksheet string, The worksheet tab name.
  - worksheetId integer, The worksheet tab id.

`advancedOptions`
list, A list containing the following elements:

- maxErrors integer, For Google Doc and Auto Imports. The maximum number of errors that can occur without the job failing.
- existingTableRows string, For Google Doc and Auto Imports. The behavior if a table with the requested name already exists. One of "fail", "truncate", "append", or "drop". Defaults to "fail".
- firstRowIsHeader boolean, For Google Doc and Auto Imports. A boolean value indicating whether or not the first row is a header row.
- diststyle string, For Auto Imports. The diststyle to use for a Redshift table.
- distkey string, For Auto Imports. The distkey to use for a Redshift table.
- sortkey1 string, For Auto Imports. The first sortkey to use for a Redshift table.

- sortkey2 string, For Auto Imports. The second sortkey to use for a Redshift table.
- columnDelimiter string, For Auto Imports. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". If column_delimiter is null or omitted, it will be auto-detected.
- columnOverrides object, For Auto Imports. Hash used for overriding auto-detected names and types, with keys being the index of the column being overridden.
- escaped boolean, For Auto Imports. If true, escape quotes with a backslash; otherwise, escape quotes by double-quoting. Defaults to false.
- identityColumn string, For DB Syncs. The column or columns to use as primary key for incremental syncs. Should be a unique identifier. If blank, primary key columns will be auto-detected. If more than one identity column is specified, an identity column must be specified for each table. We recommend the primary key be a sequential data type such as an integer, double, timestamp, date, or float. If using a primary key that is a string data type, we recommend having a last_modified_column to ensure all data is synced to the destination table.
- lastModifiedColumn string, For DB Syncs. The column to use to detect recently modified data for incremental syncs. Defaults to "Auto-Detect", which will use the first column it finds containing either "modif" or "update" in the name. When specified, only rows where last_modified_column in the source >= last_modified_column in the destination will be synced.
- rowChunkSize integer, For DB Syncs. If specified, will split the sync into chunks of this size.
- wipeDestinationTable boolean, For DB Syncs. If true, will perform a full table refresh.
- truncateLongLines boolean, For DB Syncs to Redshift. When true, truncates column data to fit the column specification.
- invalidCharReplacement string, For DB Syncs to Redshift. If specified, will replace each invalid UTF-8 character with this character. Must be a single ASCII character.
- verifyTableRowCounts boolean, For DB Syncs. When true, an error will be raised if the destination table does not have the same number of rows as the source table after the sync.
- partitionColumnName string, This parameter is deprecated
- partitionSchemaName string, This parameter is deprecated
- partitionTableName string, This parameter is deprecated
- partitionTablePartitionColumnMinName string, This parameter is deprecated
- partitionTablePartitionColumnMaxName string, This parameter is deprecated
- mysqlCatalogMatchesSchema boolean, This attribute is no longer available; defaults to true but cannot be used.
- chunkingMethod string, This parameter is deprecated

- exportAction string, For Google Doc Exports. The kind of export action you want to have the export execute. Set to "newsprsht" if you want a new worksheet inside a new spreadsheet. Set to "newwksht" if you want a new worksheet inside an existing spreadsheet. Set to "updatewksht" if you want to overwrite an existing worksheet inside an existing spreadsheet. Set to "appendwksht" if you want to append to the end of an existing worksheet inside an existing spreadsheet. Default is set to "newsprsht"
- sqlQuery string, For Google Doc Exports. The SQL query for the export.
- contactLists string, This parameter is deprecated
- soqlQuery string, This parameter is deprecated
- includeDeletedRecords boolean, This parameter is deprecated

---

`imports_put_syncs_archive` *Update the archive status of this sync*

Description

Update the archive status of this sync

Usage

```
imports_put_syncs_archive(id, sync_id, status = NULL)
```

Arguments

`id`	integer required. The ID of the import to fetch.
`sync_id`	integer required. The ID of the sync to fetch.
`status`	boolean optional. The desired archived status of the sync.

Value

A list containing the following elements:

`id` integer,

`source` list, A list containing the following elements:

- id integer, The ID of the table or file, if available.
- path string, The path of the dataset to sync from; for a database source, schema.tablename. If you are doing a Google Sheet export, this can be blank. This is a legacy parameter, it is recommended you use one of the following: databaseTable, file, googleWorksheet
- databaseTable list . A list containing the following elements:
  - schema string, The database schema name.
  - table string, The database table name.

  - useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
- file list . A list containing the following elements:
  - id integer, The file id.
- googleWorksheet list . A list containing the following elements:
  - spreadsheet string, The spreadsheet document name.
  - spreadsheetId string, The spreadsheet document id.
  - worksheet string, The worksheet tab name.
  - worksheetId integer, The worksheet tab id.
- salesforce list . A list containing the following elements:
  - objectName string, This parameter is deprecated

`destination` list, A list containing the following elements:

- path string, The schema.tablename to sync to. If you are doing a Google Sheet export, this is the spreadsheet and sheet name separated by a period. i.e. if you have a spreadsheet named "MySpreadsheet" and a sheet called "Sheet1" this field would be "MySpreadsheet.Sheet1". This is a legacy parameter, it is recommended you use one of the following: databaseTable, googleWorksheet
- databaseTable list . A list containing the following elements:
  - schema string, The database schema name.
  - table string, The database table name.
  - useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
- googleWorksheet list . A list containing the following elements:
  - spreadsheet string, The spreadsheet document name.
  - spreadsheetId string, The spreadsheet document id.
  - worksheet string, The worksheet tab name.
  - worksheetId integer, The worksheet tab id.

`advancedOptions`
list, A list containing the following elements:

- maxErrors integer, For Google Doc and Auto Imports. The maximum number of errors that can occur without the job failing.
- existingTableRows string, For Google Doc and Auto Imports. The behavior if a table with the requested name already exists. One of "fail", "truncate", "append", or "drop". Defaults to "fail".
- firstRowIsHeader boolean, For Google Doc and Auto Imports. A boolean value indicating whether or not the first row is a header row.
- diststyle string, For Auto Imports. The diststyle to use for a Redshift table.
- distkey string, For Auto Imports. The distkey to use for a Redshift table.
- sortkey1 string, For Auto Imports. The first sortkey to use for a Redshift table.
- sortkey2 string, For Auto Imports. The second sortkey to use for a Redshift table.

- columnDelimiter string, For Auto Imports. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". If column_delimiter is null or omitted, it will be auto-detected.
- columnOverrides object, For Auto Imports. Hash used for overriding auto-detected names and types, with keys being the index of the column being overridden.
- escaped boolean, For Auto Imports. If true, escape quotes with a backslash; otherwise, escape quotes by double-quoting. Defaults to false.
- identityColumn string, For DB Syncs. The column or columns to use as primary key for incremental syncs. Should be a unique identifier. If blank, primary key columns will be auto-detected. If more than one identity column is specified, an identity column must be specified for each table. We recommend the primary key be a sequential data type such as an integer, double, timestamp, date, or float. If using a primary key that is a string data type, we recommend having a last_modified_column to ensure all data is synced to the destination table.
- lastModifiedColumn string, For DB Syncs. The column to use to detect recently modified data for incremental syncs. Defaults to "Auto-Detect", which will use the first column it finds containing either "modif" or "update" in the name. When specified, only rows where last_modified_column in the source >= last_modified_column in the destination will be synced.
- rowChunkSize integer, For DB Syncs. If specified, will split the sync into chunks of this size.
- wipeDestinationTable boolean, For DB Syncs. If true, will perform a full table refresh.
- truncateLongLines boolean, For DB Syncs to Redshift. When true, truncates column data to fit the column specification.
- invalidCharReplacement string, For DB Syncs to Redshift. If specified, will replace each invalid UTF-8 character with this character. Must be a single ASCII character.
- verifyTableRowCounts boolean, For DB Syncs. When true, an error will be raised if the destination table does not have the same number of rows as the source table after the sync.
- partitionColumnName string, This parameter is deprecated
- partitionSchemaName string, This parameter is deprecated
- partitionTableName string, This parameter is deprecated
- partitionTablePartitionColumnMinName string, This parameter is deprecated
- partitionTablePartitionColumnMaxName string, This parameter is deprecated
- mysqlCatalogMatchesSchema boolean, This attribute is no longer available; defaults to true but cannot be used.
- chunkingMethod string, This parameter is deprecated
- exportAction string, For Google Doc Exports. The kind of export action you want to have the export execute. Set to "newsprsht" if you want a new worksheet inside a new spreadsheet. Set to "newwksht" if you want a new

worksheet inside an existing spreadsheet. Set to "updatewksht" if you want to overwrite an existing worksheet inside an existing spreadsheet. Set to "appendwksht" if you want to append to the end of an existing worksheet inside an existing spreadsheet. Default is set to "newsprsht"

- sqlQuery string, For Google Doc Exports. The SQL query for the export.
- contactLists string, This parameter is deprecated
- soqlQuery string, This parameter is deprecated
- includeDeletedRecords boolean, This parameter is deprecated

---

`imports_put_transfer` *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
imports_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies` array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID

- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`jobs_delete_projects` *Remove a Job from a project*

Description

Remove a Job from a project

Usage

```
jobs_delete_projects(id, project_id)
```

Arguments

`id`	integer required. The ID of the Job.
`project_id`	integer required. The ID of the project.

Value

An empty HTTP response

`jobs_delete_runs` *Cancel a run*

Description

Cancel a run

Usage

```
jobs_delete_runs(id, run_id)
```

Arguments

`id`	integer required. The ID of the Job.
`run_id`	integer required. The ID of the Run.

Value

An empty HTTP response

jobs_delete_shares_groups *Revoke the permissions a group has on this object*

Description

Revoke the permissions a group has on this object

Usage

```
jobs_delete_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

jobs_delete_shares_users *Revoke the permissions a user has on this object*

Description

Revoke the permissions a user has on this object

Usage

```
jobs_delete_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

jobs_get *Show basic job info*

Description

Show basic job info

Usage

```
jobs_get(id)
```

Arguments

id	integer required. The ID for this job.

Value

A list containing the following elements:

id	integer,
name	string,
type	string,
fromTemplateId	integer,
state	string, Whether the job is idle, queued, running, cancelled, or failed.
createdAt	string,
updatedAt	string,
runs	array, An array containing the following fields: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
hidden	boolean, The hidden status of the item.
archived	string, The archival status of the requested item(s).

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`successEmailSubject` string,

`successEmailBody` string,

`runningAsUser` string,

`runByUser` string,

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`jobs_get_runs` *Check status of a job*

Description

Check status of a job

Usage

```
jobs_get_runs(id, run_id)
```

Arguments

`id` integer required. The ID of the Job.

`run_id` integer required. The ID of the Run.

Value

A list containing the following elements:

id	integer,
state	string,
createdAt	string, The time that the run was queued.
startedAt	string, The time that the run started.
finishedAt	string, The time that the run completed.
error	string, The error message for this run, if present.

---

jobs_list *List Jobs*

Description

List Jobs

Usage

```
jobs_list(
  state = NULL,
  type = NULL,
  q = NULL,
  permission = NULL,
  scheduled = NULL,
  hidden = NULL,
  archived = NULL,
  author = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

state	string optional. The job's state. One or more of queued, running, succeeded, failed, and cancelled. Specify multiple values as a comma-separated list (e.g., "A,B").
type	string optional. The job's type. Specify multiple values as a comma-separated list (e.g., "A,B").
q	string optional. Query string to search on the id, name, and job type.
permission	string optional. A permissions string, one of "read", "write", or "manage". Lists only jobs for which the current user has that permission.

scheduled	boolean optional. If the item is scheduled.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
archived	string optional. The archival status of the requested item(s).
author	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer,
name	string,
type	string,
fromTemplateId	integer,
state	string, Whether the job is idle, queued, running, cancelled, or failed.
createdAt	string,
updatedAt	string,
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
archived	string, The archival status of the requested item(s).
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled.

- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`jobs_list_children` *Show nested tree of children that this job triggers*

Description

Show nested tree of children that this job triggers

Usage

```
jobs_list_children(id)
```

Arguments

`id`	integer required. The ID for this job.

Value

A list containing the following elements:

`id`	integer,
`name`	string,
`type`	string,
`fromTemplateId`	integer,
`state`	string,
`createdAt`	string,
`updatedAt`	string,
`runs`	array, An array containing the following fields: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
`lastRun`	list, A list containing the following elements: • id integer,

- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`children` array,

`jobs_list_dependencies`
List dependent objects for this object

Description

List dependent objects for this object

Usage

```
jobs_list_dependencies(id, user_id = NULL)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer optional. ID of target user

Value

An array containing the following fields:

`objectType`	string, Dependent object type
`fcoType`	string, Human readable dependent object type
`id`	integer, Dependent object ID
`name`	string, Dependent object name, or nil if the requesting user cannot read this object
`permissionLevel`	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
`description`	string, Additional information about the dependency, if relevant
`shareable`	boolean, Whether or not the requesting user can share this object.

`jobs_list_parents` *Show chain of parents as a list that this job triggers from*

Description

Show chain of parents as a list that this job triggers from

Usage

```
jobs_list_parents(id)
```

Arguments

`id`	integer required. The ID for this job.

Value

A list containing the following elements:

`id`	integer,
`name`	string,
`type`	string,
`fromTemplateId`	integer,
`state`	string, Whether the job is idle, queued, running, cancelled, or failed.
`createdAt`	string,
`updatedAt`	string,
`runs`	array, An array containing the following fields: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
`lastRun`	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
`hidden`	boolean, The hidden status of the item.
`archived`	string, The archival status of the requested item(s).

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`successEmailSubject` string,

`successEmailBody` string,

`runningAsUser` string,

`runByUser` string,

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

---

jobs_list_projects *List the projects a Job belongs to*

Description

List the projects a Job belongs to

Usage

```
jobs_list_projects(id, hidden = NULL)
```

Arguments

`id` integer required. The ID of the Job.

`hidden` boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

`id`	integer, The ID for this project.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The name of this project.
`description`	string, A description of the project.
`users`	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`autoShare`	boolean,
`createdAt`	string,
`updatedAt`	string,
`archived`	string, The archival status of the requested item(s).

`jobs_list_runs` *List runs for the given job*

Description

List runs for the given job

Usage

```
jobs_list_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID for this job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer,
state	string,
createdAt	string, The time that the run was queued.
startedAt	string, The time that the run started.
finishedAt	string, The time that the run completed.
error	string, The error message for this run, if present.

---

jobs_list_runs_logs *Get the logs for a run*

Description

Get the logs for a run

Usage

```
jobs_list_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the job job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

---

jobs_list_runs_outputs

List the outputs for a run

Description

List the outputs for a run

Usage

```
jobs_list_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the job.
run_id	integer required. The ID of the run.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

---

jobs_list_shares *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
jobs_list_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`jobs_list_workflows` *List the workflows a job belongs to*

Description

List the workflows a job belongs to

Usage

```
jobs_list_workflows(id, archived = NULL)
```

Arguments

`id`	integer required.
`archived`	string optional. The archival status of the requested item(s).

Value

An array containing the following fields:

`id`	integer, The ID for this workflow.
`name`	string, The name of this workflow.
`description`	string, A description of the workflow.
`valid`	boolean, The validity of the workflow definition.
`fileId`	string, The file id for the s3 file containing the workflow configuration.
`user`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`state`	string, The state of the workflow. State is "running" if any execution is running, otherwise reflects most recent execution state.
`schedule`	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

allowConcurrentExecutions
: boolean, Whether the workflow can execute when already running.

timeZone
: string, The time zone of this workflow.

nextExecutionAt
: string, The time of the next scheduled execution.

archived
: string, The archival status of the requested item(s).

createdAt
: string,

updatedAt
: string,

`jobs_post_runs` *Run a job*

Description

Run a job

Usage

```
jobs_post_runs(id)
```

Arguments

id	integer required. The ID for this job.

Value

A list containing the following elements:

id	integer,
state	string,
createdAt	string, The time that the run was queued.
startedAt	string, The time that the run started.
finishedAt	string, The time that the run completed.
error	string, The error message for this run, if present.

jobs_post_trigger_email *Generate and retrieve trigger email address*

Description

Generate and retrieve trigger email address

Usage

```
jobs_post_trigger_email(id)
```

Arguments

id	integer required. The ID for this job.

Value

A list containing the following elements:

triggerEmail	string, Email address which may be used to trigger this job to run.

jobs_put_archive *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
jobs_put_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id` integer,

`name` string,

`type` string,

`fromTemplateId` integer,

`state` string, Whether the job is idle, queued, running, cancelled, or failed.

`createdAt` string,

`updatedAt` string,

`runs` array, An array containing the following fields:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`successEmailSubject` string,

`successEmailBody` string,

`runningAsUser` string,

`runByUser` string,

schedule — list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

---

jobs_put_projects *Add a Job to a project*

Description

Add a Job to a project

Usage

```
jobs_put_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Job.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

---

jobs_put_shares_groups *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
jobs_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

jobs_put_shares_users *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
jobs_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

jobs_put_transfer *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
jobs_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. ID of target user
include_dependencies	boolean required. Whether or not to give manage permissions on all dependencies
email_body	string optional. Custom body text for e-mail sent on transfer.
send_email	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies — array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

json_values_delete_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
json_values_delete_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

json_values_delete_shares_users *Revoke the permissions a user has on this object*

Description

Revoke the permissions a user has on this object

Usage

```
json_values_delete_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

json_values_get *Get details about a JSON Value*

Description

Get details about a JSON Value

Usage

```
json_values_get(id)
```

Arguments

id	integer required. The ID of the JSON Value.

Value

A list containing the following elements:

id	integer, The ID of the JSON Value.
name	string, The name of the JSON Value.
value	string, The deserialized JSON value.

json_values_list_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
json_values_list_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

json_values_list_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
json_values_list_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

json_values_patch *Update some attributes of this JSON Value*

Description

Update some attributes of this JSON Value

Usage

```
json_values_patch(id, name = NULL, value_str = NULL)
```

Arguments

id	integer required. The ID of the JSON Value.
name	string optional. The name of the JSON Value.
value_str	string optional. The JSON value to store. Should be a serialized JSON string. Limited to 1000000 bytes.

Value

A list containing the following elements:

id	integer, The ID of the JSON Value.
name	string, The name of the JSON Value.
value	string, The deserialized JSON value.

---

json_values_post *Create a JSON Value*

Description

Create a JSON Value

Usage

```
json_values_post(value_str, name = NULL)
```

Arguments

value_str	string required. The JSON value to store. Should be a serialized JSON string. Limited to 1000000 bytes.
name	string optional. The name of the JSON Value.

Value

A list containing the following elements:

id	integer, The ID of the JSON Value.
name	string, The name of the JSON Value.
value	string, The deserialized JSON value.

---

json_values_put_shares_groups *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
json_values_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
`totalGroupShares`	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`json_values_put_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
json_values_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_ids`	array required. An array of one or more user IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`json_values_put_transfer` *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
json_values_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. ID of target user
include_dependencies	boolean required. Whether or not to give manage permissions on all dependencies
email_body	string optional. Custom body text for e-mail sent on transfer.
send_email	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies
: array, An array containing the following fields:
  - objectType string, Dependent object type
  - fcoType string, Human readable dependent object type
  - id integer, Dependent object ID
  - name string, Dependent object name, or nil if the requesting user cannot read this object
  - permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
  - description string, Additional information about the dependency, if relevant
  - shared boolean, Whether dependent object was successfully shared with target user

---

`match_targets_delete_shares_groups`
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
match_targets_delete_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

`match_targets_delete_shares_users`
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
match_targets_delete_shares_users(id, user_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. The ID of the user.

Value

An empty HTTP response

match_targets_get *Show Match Target info*

Description

Show Match Target info

Usage

```
match_targets_get(id)
```

Arguments

id	integer required. The ID of the match target

Value

A list containing the following elements:

id	integer, The ID of the match target
name	string, The name of the match target
targetFileName	string, The name of the target file
createdAt	string,
updatedAt	string,
archived	boolean, Whether the match target has been archived.

match_targets_list *List match targets*

Description

List match targets

Usage

```
match_targets_list()
```

Value

An array containing the following fields:

id	integer, The ID of the match target
name	string, The name of the match target
targetFileName	string, The name of the target file
createdAt	string,
updatedAt	string,
archived	boolean, Whether the match target has been archived.

match_targets_list_shares
List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
match_targets_list_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers list, A list containing the following elements:

- users array,
- groups array,

writers list, A list containing the following elements:

- users array,
- groups array,

owners list, A list containing the following elements:

- users array,
- groups array,

totalUserShares
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

match_targets_patch *Update a match target*

Description

Update a match target

Usage

```
match_targets_patch(id, name = NULL, target_file_name = NULL, archived = NULL)
```

Arguments

id	integer required. The ID of the match target
name	string optional. The name of the match target
target_file_name	string optional. The name of the target file
archived	boolean optional. Whether the match target has been archived.

Value

A list containing the following elements:

id	integer, The ID of the match target
name	string, The name of the match target
targetFileName	string, The name of the target file
createdAt	string,
updatedAt	string,
archived	boolean, Whether the match target has been archived.

match_targets_post *Create a new match target*

Description

Create a new match target

Usage

```
match_targets_post(name, target_file_name = NULL, archived = NULL)
```

Arguments

name	string required. The name of the match target
target_file_name	string optional. The name of the target file
archived	boolean optional. Whether the match target has been archived.

Value

A list containing the following elements:

id	integer, The ID of the match target
name	string, The name of the match target
targetFileName	string, The name of the target file
createdAt	string,
updatedAt	string,
archived	boolean, Whether the match target has been archived.

---

match_targets_put_archive

Update the archive status of this object

Description

Update the archive status of this object

Usage

```
match_targets_put_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ID of the match target
name	string, The name of the match target
targetFileName	string, The name of the target file
createdAt	string,
updatedAt	string,
archived	boolean, Whether the match target has been archived.

match_targets_put_shares_groups
Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
match_targets_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`match_targets_put_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
match_targets_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers
: list, A list containing the following elements:
- users array,
- groups array,

writers
: list, A list containing the following elements:
- users array,
- groups array,

owners
: list, A list containing the following elements:
- users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`media_delete_optimizations_runs` *Cancel a run*

Description

Cancel a run

Usage

```
media_delete_optimizations_runs(id, run_id)
```

Arguments

`id`	integer required. The ID of the Optimization job.
`run_id`	integer required. The ID of the run.

Value

An empty HTTP response

`media_delete_optimizations_shares_groups` *Revoke the permissions a group has on this object*

Description

Revoke the permissions a group has on this object

Usage

```
media_delete_optimizations_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

`media_delete_optimizations_shares_users`
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
media_delete_optimizations_shares_users(id, user_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. The ID of the user.

Value

An empty HTTP response

`media_delete_ratecards_shares_groups`
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
media_delete_ratecards_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

media_delete_ratecards_shares_users
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
media_delete_ratecards_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

media_delete_spot_orders_shares_groups
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
media_delete_spot_orders_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

`media_delete_spot_orders_shares_users`
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
media_delete_spot_orders_shares_users(id, user_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. The ID of the user.

Value

An empty HTTP response

`media_get_optimizations`
Show a single optimization

Description

Show a single optimization

Usage

```
media_get_optimizations(id)
```

Arguments

`id`	integer required. The optimization ID.

Value

A list containing the following elements:

`id`	integer, The optimization ID.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name.

- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of the optimization.

`createdAt` string,

`updatedAt` string,

`finishedAt` string, The end time of the last run.

`state` string, The state of the last run.

`lastRunId` integer, The ID of the last run.

`spotOrderId` integer, The ID for the spot order produced by the optimization.

`archived` string, The archival status of the requested item(s).

`reportLink` string, A link to the visual report for the optimization.

`spotOrderLink` string, A link to the json version of the spot order.

`fileLinks` array, Links to the csv and xml versions of the spot order.

`runs` array, An array containing the following fields:

- marketId integer, The market ID.
- startDate string, The start date for the media run.
- endDate string, The end date for the media run.
- forceCpm boolean, Whether to force optimization to use CPM data even if partition data is available.
- reachAlpha number, A tuning parameter used to adjust RF.
- syscodes array, The syscodes for the media run.
- rateCards array, The ratecards for the media run.
- constraints array, The constraints for the media run.

`programs` array, An array of programs that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_programs is not also set.

`networks` array, An array of networks that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_networks is not also set.

`excludePrograms` boolean, If Civis Media Optimizer should exclude the programs in the programs parameter.If this value is set to false, it will make the optimization limit itself to the programs supplied through the programs parameter.An error will be thrown if programs is not also set.

`excludeNetworks` boolean, If Civis Media Optimizer should exclude the networks in the networks parameter.If this value is set to false, it will make the optimization limit itself to the networks supplied through the networks.An error will be thrown if networks is not also set.

`timeSlotPercentages` list, The maximum amount of the budget spent on that particular day of the week, daypart, or specific time slot for broadcast and cable.

media_get_optimizations_runs *Check status of a run*

Description

Check status of a run

Usage

```
media_get_optimizations_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Optimization job.
run_id	integer required. The ID of the run.

Value

A list containing the following elements:

id	integer, The ID of the run.
optimizationId	integer, The ID of the Optimization job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

media_get_ratecards *Get a Ratecard*

Description

Get a Ratecard

Usage

```
media_get_ratecards(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ratecard ID.
filename	string, Name of the ratecard file.
startOn	string, First day to which the ratecard applies.
endOn	string, Last day to which the ratecard applies.
dmaNumber	integer, Number of the DMA associated with the ratecard.
archived	string, The archival status of the requested item(s).

---

media_get_spot_orders *Show a single spot order*

Description

Show a single spot order

Usage

```
media_get_spot_orders(id)
```

Arguments

id	integer required. The ID for the spot order.

Value

A list containing the following elements:

id	integer, The ID for the spot order.
archived	string, The archival status of the requested item(s).
csvS3Uri	string, S3 URI for the spot order CSV file.
jsonS3Uri	string, S3 URI for the spot order JSON file.
xmlArchiveS3Uri	string, S3 URI for the spot order XML archive.
lastTransformJobId	integer, ID of the spot order transformation job.

media_list_dmas *List all Designated Market Areas*

Description

List all Designated Market Areas

Usage

```
media_list_dmas(name = NULL, number = NULL)
```

Arguments

name	string optional. If specified, will be used to filter the DMAs returned. Substring matching is supported with "%" and "*" wildcards (e.g., "name=%region%" will return both "region1" and "my region").
number	integer optional. If specified, will be used to filter the DMAS by number.

Value

An array containing the following fields:

name	string, Name for the DMA region.
number	integer, Identifier number for a DMA.

media_list_optimizations *List all optimizations*

Description

List all optimizations

Usage

```
media_list_optimizations(
  archived = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

archived	string optional. The archival status of the requested item(s).
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, author, name.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The optimization ID.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of the optimization.
createdAt	string,
updatedAt	string,
finishedAt	string, The end time of the last run.
state	string, The state of the last run.
lastRunId	integer, The ID of the last run.
spotOrderId	integer, The ID for the spot order produced by the optimization.
archived	string, The archival status of the requested item(s).

---

media_list_optimizations_runs

List runs for the given Optimization job

Description

List runs for the given Optimization job

Usage

```
media_list_optimizations_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the Optimization job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
optimizationId	integer, The ID of the Optimization job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

---

media_list_optimizations_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

```
media_list_optimizations_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the Optimization job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

---

media_list_optimizations_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
media_list_optimizations_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares`
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`media_list_ratecards` *List all ratecards*

Description

List all ratecards

Usage

```
media_list_ratecards(archived = NULL, filename = NULL, dma_number = NULL)
```

Arguments

`archived`	string optional. The archival status of the requested item(s).
`filename`	string optional. If specified, will be used to filter the ratecards returned. Substring matching is supported with "%" and "*" wildcards (e.g., "filename=%ratecard%" will return both "ratecard 1" and "my ratecard").
`dma_number`	integer optional. If specified, will be used to filter the ratecards by DMA.

Value

An array containing the following fields:

id	integer, The ratecard ID.
filename	string, Name of the ratecard file.
startOn	string, First day to which the ratecard applies.
endOn	string, Last day to which the ratecard applies.
dmaNumber	integer, Number of the DMA associated with the ratecard.
archived	string, The archival status of the requested item(s).

---

media_list_ratecards_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
media_list_ratecards_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

media_list_spot_orders

List all spot orders

Description

List all spot orders

Usage

```
media_list_spot_orders(id = NULL, archived = NULL)
```

Arguments

id	integer optional. The ID for the spot order.
archived	string optional. The archival status of the requested item(s).

Value

An array containing the following fields:

id	integer, The ID for the spot order.
archived	string, The archival status of the requested item(s).

media_list_spot_orders_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
media_list_spot_orders_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares`
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`media_list_targets` *List all Media Targets*

Description

List all Media Targets

Usage

```
media_list_targets(name = NULL, identifier = NULL, data_source = NULL)
```

Arguments

`name`	string optional. The name of the target.
`identifier`	string optional. A unique identifier for this target.
`data_source`	string optional. The source of viewership data for this target.

Value

An array containing the following fields:

`name`	string, The name of the target.
`identifier`	string, A unique identifier for this target.
`dataSource`	string, The source of viewership data for this target.

`media_patch_optimizations` *Edit an existing optimization*

Description

Edit an existing optimization

Usage

```
media_patch_optimizations(
  id,
  name = NULL,
  runs = NULL,
  programs = NULL,
  networks = NULL,
  exclude_programs = NULL,
  exclude_networks = NULL,
  time_slot_percentages = NULL
)
```

Arguments

`id`	integer required. The optimization ID.
`name`	string optional. The name of the optimization.
`runs`	array optional. An array containing the following fields: • marketId integer, The market ID. • startDate string, The start date for the media run. • endDate string, The end date for the media run. • forceCpm boolean, Whether to force optimization to use CPM data even if partition data is available. • reachAlpha number, A tuning parameter used to adjust RF. • syscodes array, The syscodes for the media run. • rateCards array, The ratecards for the media run. • constraints array, The constraints for the media run.
`programs`	array optional. An array of programs that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_programs is not also set.
`networks`	array optional. An array of networks that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_networks is not also set.
`exclude_programs`	boolean optional. If Civis Media Optimizer should exclude the programs in the programs parameter.If this value is set to false, it will make the optimization limit itself to the programs supplied through the programs parameter.An error will be thrown if programs is not also set.

`exclude_networks`
: boolean optional. If Civis Media Optimizer should exclude the networks in the networks parameter.If this value is set to false, it will make the optimization limit itself to the networks supplied through the networks.An error will be thrown if networks is not also set.

`time_slot_percentages`
: list optional. The maximum amount of the budget spent on that particular day of the week, daypart, or specific time slot for broadcast and cable.

Value

A list containing the following elements:

`id` integer, The optimization ID.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of the optimization.

`createdAt` string,

`updatedAt` string,

`finishedAt` string, The end time of the last run.

`state` string, The state of the last run.

`lastRunId` integer, The ID of the last run.

`spotOrderId` integer, The ID for the spot order produced by the optimization.

`archived` string, The archival status of the requested item(s).

`reportLink` string, A link to the visual report for the optimization.

`spotOrderLink` string, A link to the json version of the spot order.

`fileLinks` array, Links to the csv and xml versions of the spot order.

`runs` array, An array containing the following fields:

- marketId integer, The market ID.
- startDate string, The start date for the media run.
- endDate string, The end date for the media run.
- forceCpm boolean, Whether to force optimization to use CPM data even if partition data is available.
- reachAlpha number, A tuning parameter used to adjust RF.
- syscodes array, The syscodes for the media run.
- rateCards array, The ratecards for the media run.
- constraints array, The constraints for the media run.

`programs` array, An array of programs that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_programs is not also set.

networks array, An array of networks that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_networks is not also set.

excludePrograms
boolean, If Civis Media Optimizer should exclude the programs in the programs parameter.If this value is set to false, it will make the optimization limit itself to the programs supplied through the programs parameter.An error will be thrown if programs is not also set.

excludeNetworks
boolean, If Civis Media Optimizer should exclude the networks in the networks parameter.If this value is set to false, it will make the optimization limit itself to the networks supplied through the networks.An error will be thrown if networks is not also set.

timeSlotPercentages
list, The maximum amount of the budget spent on that particular day of the week, daypart, or specific time slot for broadcast and cable.

`media_patch_ratecards` *Update some attributes of this Ratecard*

Description

Update some attributes of this Ratecard

Usage

```
media_patch_ratecards(
  id,
  filename = NULL,
  start_on = NULL,
  end_on = NULL,
  dma_number = NULL
)
```

Arguments

id	integer required. The ratecard ID.
filename	string optional. Name of the ratecard file.
start_on	string optional. First day to which the ratecard applies.
end_on	string optional. Last day to which the ratecard applies.
dma_number	integer optional. Number of the DMA associated with the ratecard.

Value

A list containing the following elements:

id	integer, The ratecard ID.
filename	string, Name of the ratecard file.
startOn	string, First day to which the ratecard applies.
endOn	string, Last day to which the ratecard applies.
dmaNumber	integer, Number of the DMA associated with the ratecard.
archived	string, The archival status of the requested item(s).

---

media_post_optimizations

Create a new optimization

Description

Create a new optimization

Usage

```
media_post_optimizations(
  runs,
  name = NULL,
  programs = NULL,
  networks = NULL,
  exclude_programs = NULL,
  exclude_networks = NULL,
  time_slot_percentages = NULL
)
```

Arguments

runs	array required. An array containing the following fields: • marketId integer, The market ID. • startDate string, The start date for the media run. • endDate string, The end date for the media run. • forceCpm boolean, Whether to force optimization to use CPM data even if partition data is available. • reachAlpha number, A tuning parameter used to adjust RF. • syscodes array, The syscodes for the media run. • rateCards array, The ratecards for the media run. • constraints array, The constraints for the media run.
name	string optional. The name of the optimization.

`programs` array optional. An array of programs that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_programs is not also set.

`networks` array optional. An array of networks that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_networks is not also set.

`exclude_programs` boolean optional. If Civis Media Optimizer should exclude the programs in the programs parameter.If this value is set to false, it will make the optimization limit itself to the programs supplied through the programs parameter.An error will be thrown if programs is not also set.

`exclude_networks` boolean optional. If Civis Media Optimizer should exclude the networks in the networks parameter.If this value is set to false, it will make the optimization limit itself to the networks supplied through the networks.An error will be thrown if networks is not also set.

`time_slot_percentages` list optional. The maximum amount of the budget spent on that particular day of the week, daypart, or specific time slot for broadcast and cable.

Value

A list containing the following elements:

`id` integer, The optimization ID.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of the optimization.

`createdAt` string,

`updatedAt` string,

`finishedAt` string, The end time of the last run.

`state` string, The state of the last run.

`lastRunId` integer, The ID of the last run.

`spotOrderId` integer, The ID for the spot order produced by the optimization.

`archived` string, The archival status of the requested item(s).

`reportLink` string, A link to the visual report for the optimization.

`spotOrderLink` string, A link to the json version of the spot order.

`fileLinks` array, Links to the csv and xml versions of the spot order.

`runs` array, An array containing the following fields:

- marketId integer, The market ID.
- startDate string, The start date for the media run.

- endDate string, The end date for the media run.
- forceCpm boolean, Whether to force optimization to use CPM data even if partition data is available.
- reachAlpha number, A tuning parameter used to adjust RF.
- syscodes array, The syscodes for the media run.
- rateCards array, The ratecards for the media run.
- constraints array, The constraints for the media run.

`programs` array, An array of programs that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_programs is not also set.

`networks` array, An array of networks that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_networks is not also set.

`excludePrograms`
boolean, If Civis Media Optimizer should exclude the programs in the programs parameter.If this value is set to false, it will make the optimization limit itself to the programs supplied through the programs parameter.An error will be thrown if programs is not also set.

`excludeNetworks`
boolean, If Civis Media Optimizer should exclude the networks in the networks parameter.If this value is set to false, it will make the optimization limit itself to the networks supplied through the networks.An error will be thrown if networks is not also set.

`timeSlotPercentages`
list, The maximum amount of the budget spent on that particular day of the week, daypart, or specific time slot for broadcast and cable.

`media_post_optimizations_clone`
Clone an existing optimization

Description

Clone an existing optimization

Usage

```
media_post_optimizations_clone(id)
```

Arguments

`id` integer required. The optimization ID.

Value

A list containing the following elements:

`id` integer, The optimization ID.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of the optimization.

`createdAt` string,

`updatedAt` string,

`finishedAt` string, The end time of the last run.

`state` string, The state of the last run.

`lastRunId` integer, The ID of the last run.

`spotOrderId` integer, The ID for the spot order produced by the optimization.

`archived` string, The archival status of the requested item(s).

`reportLink` string, A link to the visual report for the optimization.

`spotOrderLink` string, A link to the json version of the spot order.

`fileLinks` array, Links to the csv and xml versions of the spot order.

`runs` array, An array containing the following fields:

- marketId integer, The market ID.
- startDate string, The start date for the media run.
- endDate string, The end date for the media run.
- forceCpm boolean, Whether to force optimization to use CPM data even if partition data is available.
- reachAlpha number, A tuning parameter used to adjust RF.
- syscodes array, The syscodes for the media run.
- rateCards array, The ratecards for the media run.
- constraints array, The constraints for the media run.

`programs` array, An array of programs that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_programs is not also set.

`networks` array, An array of networks that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_networks is not also set.

`excludePrograms` boolean, If Civis Media Optimizer should exclude the programs in the programs parameter.If this value is set to false, it will make the optimization limit itself to the programs supplied through the programs parameter.An error will be thrown if programs is not also set.

excludeNetworks
: boolean, If Civis Media Optimizer should exclude the networks in the networks parameter.If this value is set to false, it will make the optimization limit itself to the networks supplied through the networks.An error will be thrown if networks is not also set.

timeSlotPercentages
: list, The maximum amount of the budget spent on that particular day of the week, daypart, or specific time slot for broadcast and cable.

`media_post_optimizations_runs` *Start a run*

Description

Start a run

Usage

```
media_post_optimizations_runs(id)
```

Arguments

id	integer required. The ID of the Optimization job.

Value

A list containing the following elements:

id	integer, The ID of the run.
optimizationId	integer, The ID of the Optimization job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

media_post_ratecards *Create a Ratecard*

Description

Create a Ratecard

Usage

```
media_post_ratecards(filename, start_on, end_on, dma_number)
```

Arguments

filename	string required. Name of the ratecard file.
start_on	string required. First day to which the ratecard applies.
end_on	string required. Last day to which the ratecard applies.
dma_number	integer required. Number of the DMA associated with the ratecard.

Value

A list containing the following elements:

id	integer, The ratecard ID.
filename	string, Name of the ratecard file.
startOn	string, First day to which the ratecard applies.
endOn	string, Last day to which the ratecard applies.
dmaNumber	integer, Number of the DMA associated with the ratecard.
archived	string, The archival status of the requested item(s).

media_post_spot_orders *Create a spot order*

Description

Create a spot order

Usage

```
media_post_spot_orders(body = NULL)
```

Arguments

body	string optional. CSV body of a spot order.

Value

A list containing the following elements:

id	integer, The ID for the spot order.
archived	string, The archival status of the requested item(s).
csvS3Uri	string, S3 URI for the spot order CSV file.
jsonS3Uri	string, S3 URI for the spot order JSON file.
xmlArchiveS3Uri	string, S3 URI for the spot order XML archive.
lastTransformJobId	integer, ID of the spot order transformation job.

---

media_put_optimizations_archive

Update the archive status of this object

Description

Update the archive status of this object

Usage

```
media_put_optimizations_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The optimization ID.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of the optimization.
createdAt	string,
updatedAt	string,

`finishedAt` string, The end time of the last run.

`state` string, The state of the last run.

`lastRunId` integer, The ID of the last run.

`spotOrderId` integer, The ID for the spot order produced by the optimization.

`archived` string, The archival status of the requested item(s).

`reportLink` string, A link to the visual report for the optimization.

`spotOrderLink` string, A link to the json version of the spot order.

`fileLinks` array, Links to the csv and xml versions of the spot order.

`runs` array, An array containing the following fields:

- marketId integer, The market ID.
- startDate string, The start date for the media run.
- endDate string, The end date for the media run.
- forceCpm boolean, Whether to force optimization to use CPM data even if partition data is available.
- reachAlpha number, A tuning parameter used to adjust RF.
- syscodes array, The syscodes for the media run.
- rateCards array, The ratecards for the media run.
- constraints array, The constraints for the media run.

`programs` array, An array of programs that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_programs is not also set.

`networks` array, An array of networks that the Civis Media Optimizer either exclude or limit to.An error will be thrown if exclude_networks is not also set.

`excludePrograms` boolean, If Civis Media Optimizer should exclude the programs in the programs parameter.If this value is set to false, it will make the optimization limit itself to the programs supplied through the programs parameter.An error will be thrown if programs is not also set.

`excludeNetworks` boolean, If Civis Media Optimizer should exclude the networks in the networks parameter.If this value is set to false, it will make the optimization limit itself to the networks supplied through the networks.An error will be thrown if networks is not also set.

`timeSlotPercentages` list, The maximum amount of the budget spent on that particular day of the week, daypart, or specific time slot for broadcast and cable.

`media_put_optimizations_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
media_put_optimizations_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`media_put_optimizations_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
media_put_optimizations_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_ids`	array required. An array of one or more user IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`media_put_ratecards` *Replace all attributes of this Ratecard*

Description

Replace all attributes of this Ratecard

Usage

```
media_put_ratecards(id, filename, start_on, end_on, dma_number)
```

Arguments

`id`	integer required. The ratecard ID.
`filename`	string required. Name of the ratecard file.
`start_on`	string required. First day to which the ratecard applies.
`end_on`	string required. Last day to which the ratecard applies.
`dma_number`	integer required. Number of the DMA associated with the ratecard.

Value

A list containing the following elements:

`id`	integer, The ratecard ID.
`filename`	string, Name of the ratecard file.
`startOn`	string, First day to which the ratecard applies.
`endOn`	string, Last day to which the ratecard applies.
`dmaNumber`	integer, Number of the DMA associated with the ratecard.
`archived`	string, The archival status of the requested item(s).

media_put_ratecards_archive

Update the archive status of this object

Description

Update the archive status of this object

Usage

```
media_put_ratecards_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ratecard ID.
filename	string, Name of the ratecard file.
startOn	string, First day to which the ratecard applies.
endOn	string, Last day to which the ratecard applies.
dmaNumber	integer, Number of the DMA associated with the ratecard.
archived	string, The archival status of the requested item(s).

media_put_ratecards_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
media_put_ratecards_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

media_put_ratecards_shares_users

Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

```
media_put_ratecards_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

media_put_spot_orders *Edit the specified spot order*

Description

Edit the specified spot order

Usage

```
media_put_spot_orders(id, body = NULL)
```

Arguments

id	integer required. The ID for the spot order.
body	string optional. CSV body of a spot order.

Value

A list containing the following elements:

id	integer, The ID for the spot order.
archived	string, The archival status of the requested item(s).
csvS3Uri	string, S3 URI for the spot order CSV file.
jsonS3Uri	string, S3 URI for the spot order JSON file.
xmlArchiveS3Uri	string, S3 URI for the spot order XML archive.
lastTransformJobId	integer, ID of the spot order transformation job.

---

media_put_spot_orders_archive

Update the archive status of this object

Description

Update the archive status of this object

Usage

```
media_put_spot_orders_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ID for the spot order.
archived	string, The archival status of the requested item(s).
csvS3Uri	string, S3 URI for the spot order CSV file.
jsonS3Uri	string, S3 URI for the spot order JSON file.
xmlArchiveS3Uri	string, S3 URI for the spot order XML archive.
lastTransformJobId	integer, ID of the spot order transformation job.

`media_put_spot_orders_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
media_put_spot_orders_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`media_put_spot_orders_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
media_put_spot_orders_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_ids`	array required. An array of one or more user IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`models_delete_builds` *Cancel a build*

Description

Cancel a build

Usage

```
models_delete_builds(id, build_id)
```

Arguments

`id`	integer required. The ID of the Model job.
`build_id`	integer required. The ID of the build.

Value

An empty HTTP response

`models_delete_projects`
Remove a Model from a project

Description

Remove a Model from a project

Usage

```
models_delete_projects(id, project_id)
```

Arguments

`id`	integer required. The ID of the Model.
`project_id`	integer required. The ID of the project.

Value

An empty HTTP response

`models_delete_shares_groups` *Revoke the permissions a group has on this object*

Description

Revoke the permissions a group has on this object

Usage

```
models_delete_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

`models_delete_shares_users` *Revoke the permissions a user has on this object*

Description

Revoke the permissions a user has on this object

Usage

```
models_delete_shares_users(id, user_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. The ID of the user.

Value

An empty HTTP response

`models_get` *Retrieve model configuration*

Description

Retrieve model configuration

Usage

```
models_get(id)
```

Arguments

`id`	integer required. The ID of the model.

Value

A list containing the following elements:

`id`	integer, The ID of the model.
`tableName`	string, The qualified name of the table containing the training set from which to build the model.
`databaseId`	integer, The ID of the database holding the training set table used to build the model.
`credentialId`	integer, The ID of the credential used to read the target table. Defaults to the user's default credential.
`modelName`	string, The name of the model.
`description`	string, A description of the model.
`interactionTerms`	boolean, Whether to search for interaction terms.
`boxCoxTransformation`	boolean, Whether to transform data so that it assumes a normal distribution. Valid only with continuous models.
`modelTypeId`	integer, The ID of the model's type.
`primaryKey`	string, The unique ID (primary key) of the training dataset.
`dependentVariable`	string, The dependent variable of the training dataset.
`dependentVariableOrder`	array, The order of dependent variables, especially useful for Ordinal Modeling.
`excludedColumns`	array, A list of columns which will be considered ineligible to be independent variables.
`limitingSQL`	string, A custom SQL WHERE clause used to filter the rows used to build the model. (e.g., "id > 105").

`activeBuildId` integer, The ID of the current active build, the build used to score predictions.

`crossValidationParameters`
list, Cross validation parameter grid for tree methods, e.g. {"n_estimators": [100, 200, 500], "learning_rate": [0.01, 0.1], "max_depth": [2, 3]}.

`numberOfFolds` integer, Number of folds for cross validation. Default value is 5.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`parentId` integer, The ID of the parent job that will trigger this model.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`timeZone` string, The time zone of this model.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`hidden` boolean, The hidden status of the item.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`createdAt` string, The time the model was created.

`updatedAt` string, The time the model was updated.

`currentBuildState`
string, The status of the current model build. One of "succeeded", "failed", "queued", or "running,"or "idle", if no build has been attempted.

`currentBuildException`
string, Exception message, if applicable, of the current model build.

`builds` array, An array containing the following fields:

- id integer, The ID of the model build.
- name string, The name of the model build.
- createdAt string, The time the model build was created.
- description string, A description of the model build.
- rootMeanSquaredError number, A key metric for continuous models. Nil for other model types.
- rSquaredError number, A key metric for continuous models. Nil for other model types.
- rocAuc number, A key metric for binary, multinomial, and ordinal models. Nil for other model types.

`predictions` array, An array containing the following fields:

- id integer, The ID of the model to which to apply the prediction.
- tableName string, The qualified name of the table on which to apply the predictive model.
- primaryKey array, The primary key or composite keys of the table being predicted.
- limitingSQL string, A SQL WHERE clause used to scope the rows to be predicted.
- outputTable string, The qualified name of the table to be created which will contain the model's predictions.
- schedule object,

- state string, The status of the prediction. One of: "succeeded", "failed", "queued", or "running,"or "idle", if no build has been attempted.

`lastOutputLocation`
string, The output JSON for the last build.

`archived` string, The archival status of the requested item(s).

`models_get_builds` *Check status of a build*

Description

Check status of a build

Usage

```
models_get_builds(id, build_id)
```

Arguments

`id`	integer required. The ID of the Model job.
`build_id`	integer required. The ID of the build.

Value

A list containing the following elements:

`id`	integer, The ID of the model build.
`state`	string, The state of the model build.one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
`error`	string, The error, if any, returned by the build.
`name`	string, The name of the model build.
`createdAt`	string, The time the model build was created.
`description`	string, A description of the model build.
`rootMeanSquaredError`	number, A key metric for continuous models. Nil for other model types.
`rSquaredError`	number, A key metric for continuous models. Nil for other model types.
`rocAuc`	number, A key metric for binary, multinomial, and ordinal models. Nil for other model types.
`transformationMetadata`	string, A string representing the full JSON output of the metadata for transformation of column names
`output`	string, A string representing the JSON output for the specified build. Only present when smaller than 10KB in size.
`outputLocation`	string, A URL representing the location of the full JSON output for the specified build.The URL link will be valid for 5 minutes.

`models_list` *List*

Description

List

Usage

```
models_list(
  model_name = NULL,
  training_table_name = NULL,
  dependent_variable = NULL,
  status = NULL,
  author = NULL,
  hidden = NULL,
  archived = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

`model_name`	string optional. If specified, will be used to filter the models returned. Substring matching is supported. (e.g., "modelName=model" will return both "model1" and "my model").
`training_table_name`	string optional. If specified, will be used to filter the models returned by the training dataset table name. Substring matching is supported. (e.g., "trainingTableName=table" will return both "table1" and "my_table").
`dependent_variable`	string optional. If specified, will be used to filter the models returned by the dependent variable column name. Substring matching is supported. (e.g., "dependentVariable=predictor" will return both "predictor" and "my predictor").
`status`	string optional. If specified, returns models with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'failed', 'succeeded', 'idle', 'scheduled'.
`author`	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.
`hidden`	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
`archived`	string optional. The archival status of the requested item(s).
`limit`	integer optional. Number of results to return. Defaults to its maximum of 50.

page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at, last_run.updated_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the model.
tableName	string, The qualified name of the table containing the training set from which to build the model.
databaseId	integer, The ID of the database holding the training set table used to build the model.
credentialId	integer, The ID of the credential used to read the target table. Defaults to the user's default credential.
modelName	string, The name of the model.
description	string, A description of the model.
interactionTerms	boolean, Whether to search for interaction terms.
boxCoxTransformation	boolean, Whether to transform data so that it assumes a normal distribution. Valid only with continuous models.
modelTypeId	integer, The ID of the model's type.
primaryKey	string, The unique ID (primary key) of the training dataset.
dependentVariable	string, The dependent variable of the training dataset.
dependentVariableOrder	array, The order of dependent variables, especially useful for Ordinal Modeling.
excludedColumns	array, A list of columns which will be considered ineligible to be independent variables.
limitingSQL	string, A custom SQL WHERE clause used to filter the rows used to build the model. (e.g., "id > 105").
crossValidationParameters	list, Cross validation parameter grid for tree methods, e.g. {"n_estimators": [100, 200, 500], "learning_rate": [0.01, 0.1], "max_depth": [2, 3]}.
numberOfFolds	integer, Number of folds for cross validation. Default value is 5.
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth

- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`parentId` integer, The ID of the parent job that will trigger this model.

`timeZone` string, The time zone of this model.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`createdAt` string, The time the model was created.

`updatedAt` string, The time the model was updated.

`currentBuildState` string, The status of the current model build. One of "succeeded", "failed", "queued", or "running,"or "idle", if no build has been attempted.

`currentBuildException` string, Exception message, if applicable, of the current model build.

`builds` array, An array containing the following fields:

- id integer, The ID of the model build.
- name string, The name of the model build.
- createdAt string, The time the model build was created.
- description string, A description of the model build.
- rootMeanSquaredError number, A key metric for continuous models. Nil for other model types.
- rSquaredError number, A key metric for continuous models. Nil for other model types.
- rocAuc number, A key metric for binary, multinomial, and ordinal models. Nil for other model types.

`predictions` array, An array containing the following fields:

- id integer, The ID of the model to which to apply the prediction.
- tableName string, The qualified name of the table on which to apply the predictive model.

- primaryKey array, The primary key or composite keys of the table being predicted.
- limitingSQL string, A SQL WHERE clause used to scope the rows to be predicted.
- outputTable string, The qualified name of the table to be created which will contain the model's predictions.
- state string, The status of the prediction. One of: "succeeded", "failed", "queued", or "running,"or "idle", if no build has been attempted.

`lastOutputLocation`
: string, The output JSON for the last build.

`archived`
: string, The archival status of the requested item(s).

`models_list_builds` *List builds for the given Model job*

Description

List builds for the given Model job

Usage

```
models_list_builds(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

`id`	integer required. The ID of the Model job.
`limit`	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
`page_num`	integer optional. Page number of the results to return. Defaults to the first page, 1.
`order`	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
`order_dir`	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the model build.
state	string, The state of the model build.one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
error	string, The error, if any, returned by the build.
name	string, The name of the model build.
createdAt	string, The time the model build was created.
description	string, A description of the model build.
rootMeanSquaredError	number, A key metric for continuous models. Nil for other model types.
rSquaredError	number, A key metric for continuous models. Nil for other model types.
rocAuc	number, A key metric for binary, multinomial, and ordinal models. Nil for other model types.
transformationMetadata	string, A string representing the full JSON output of the metadata for transformation of column names
output	string, A string representing the JSON output for the specified build. Only present when smaller than 10KB in size.
outputLocation	string, A URL representing the location of the full JSON output for the specified build.The URL link will be valid for 5 minutes.

`models_list_builds_logs`

Get the logs for a build

Description

Get the logs for a build

Usage

```
models_list_builds_logs(id, build_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the Model job.
build_id	integer required. The ID of the build.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

---

models_list_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
models_list_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

`models_list_projects` *List the projects a Model belongs to*

Description

List the projects a Model belongs to

Usage

```
models_list_projects(id, hidden = NULL)
```

Arguments

`id`	integer required. The ID of the Model.
`hidden`	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

`id`	integer, The ID for this project.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The name of this project.
`description`	string, A description of the project.
`users`	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`autoShare`	boolean,
`createdAt`	string,
`updatedAt`	string,
`archived`	string, The archival status of the requested item(s).

models_list_schedules *Show the model build schedule*

Description

Show the model build schedule

Usage

```
models_list_schedules(id)
```

Arguments

id	integer required. The ID of the model associated with this schedule.

Value

A list containing the following elements:

id	integer, The ID of the model associated with this schedule.
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

models_list_shares *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
models_list_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

models_list_types *List all available model types*

Description

List all available model types

Usage

```
models_list_types()
```

Value

An array containing the following fields:

`id` integer, The ID of the model type.

`algorithm` string, The name of the algorithm used to train the model.

`dvType` string, The type of dependent variable predicted by the model.

`fintAllowed` boolean, Whether this model type supports searching for interaction terms.

`models_put_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
models_put_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id`	integer, The ID of the model.
`tableName`	string, The qualified name of the table containing the training set from which to build the model.
`databaseId`	integer, The ID of the database holding the training set table used to build the model.
`credentialId`	integer, The ID of the credential used to read the target table. Defaults to the user's default credential.
`modelName`	string, The name of the model.
`description`	string, A description of the model.
`interactionTerms`	boolean, Whether to search for interaction terms.
`boxCoxTransformation`	boolean, Whether to transform data so that it assumes a normal distribution. Valid only with continuous models.
`modelTypeId`	integer, The ID of the model's type.
`primaryKey`	string, The unique ID (primary key) of the training dataset.
`dependentVariable`	string, The dependent variable of the training dataset.
`dependentVariableOrder`	array, The order of dependent variables, especially useful for Ordinal Modeling.
`excludedColumns`	array, A list of columns which will be considered ineligible to be independent variables.

`limitingSQL` string, A custom SQL WHERE clause used to filter the rows used to build the model. (e.g., "id > 105").

`activeBuildId` integer, The ID of the current active build, the build used to score predictions.

`crossValidationParameters`
list, Cross validation parameter grid for tree methods, e.g. {"n_estimators": [100, 200, 500], "learning_rate": [0.01, 0.1], "max_depth": [2, 3]}.

`numberOfFolds` integer, Number of folds for cross validation. Default value is 5.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`parentId` integer, The ID of the parent job that will trigger this model.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`timeZone` string, The time zone of this model.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`hidden` boolean, The hidden status of the item.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`createdAt` string, The time the model was created.

`updatedAt` string, The time the model was updated.

`currentBuildState` string, The status of the current model build. One of "succeeded", "failed", "queued", or "running,"or "idle", if no build has been attempted.

`currentBuildException` string, Exception message, if applicable, of the current model build.

`builds` array, An array containing the following fields:

- id integer, The ID of the model build.
- name string, The name of the model build.
- createdAt string, The time the model build was created.
- description string, A description of the model build.
- rootMeanSquaredError number, A key metric for continuous models. Nil for other model types.
- rSquaredError number, A key metric for continuous models. Nil for other model types.
- rocAuc number, A key metric for binary, multinomial, and ordinal models. Nil for other model types.

`predictions` array, An array containing the following fields:

- id integer, The ID of the model to which to apply the prediction.
- tableName string, The qualified name of the table on which to apply the predictive model.
- primaryKey array, The primary key or composite keys of the table being predicted.
- limitingSQL string, A SQL WHERE clause used to scope the rows to be predicted.
- outputTable string, The qualified name of the table to be created which will contain the model's predictions.

- schedule object,
- state string, The status of the prediction. One of: "succeeded", "failed", "queued", or "running,"or "idle", if no build has been attempted.

`lastOutputLocation`
: string, The output JSON for the last build.

`archived`
: string, The archival status of the requested item(s).

`models_put_projects` *Add a Model to a project*

Description

Add a Model to a project

Usage

```
models_put_projects(id, project_id)
```

Arguments

`id`	integer required. The ID of the Model.
`project_id`	integer required. The ID of the project.

Value

An empty HTTP response

`models_put_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
models_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

models_put_shares_users

Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

```
models_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

models_put_transfer *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
models_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. ID of target user
include_dependencies	boolean required. Whether or not to give manage permissions on all dependencies
email_body	string optional. Custom body text for e-mail sent on transfer.
send_email	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies — array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

multipart_upload *Upload to files endpoint in parts.*

Description

If a future::plan has been set, will be carried out in parallel.

Usage

```
multipart_upload(file, name = "", chunk_size = 32 * 1024, expires_at = NULL)
```

Arguments

file	the file
name	name of the upload, defaults to
chunk_size	size of the chunks in bytes
expires_at	when the file expires (default never).

notebooks_delete *Archive a Notebook (deprecated, use archiving endpoints instead)*

Description

Archive a Notebook (deprecated, use archiving endpoints instead)

Usage

```
notebooks_delete(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

notebooks_delete_deployments *Delete a Notebook deployment*

Description

Delete a Notebook deployment

Usage

```
notebooks_delete_deployments(notebook_id, deployment_id)
```

Arguments

notebook_id	integer required. The ID of the owning Notebook
deployment_id	integer required. The ID for this deployment

Value

An empty HTTP response

notebooks_delete_projects

Remove a Notebook from a project

Description

Remove a Notebook from a project

Usage

```
notebooks_delete_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Notebook.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

notebooks_delete_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
notebooks_delete_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

notebooks_delete_shares_users *Revoke the permissions a user has on this object*

Description

Revoke the permissions a user has on this object

Usage

```
notebooks_delete_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

notebooks_get *Get a Notebook*

Description

Get a Notebook

Usage

```
notebooks_get(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for this notebook.
name	string, The name of this notebook.
language	string, The kernel language of this notebook ("python3" or "r"). Defaults to "python3".
description	string, The description of this notebook.

`notebookUrl` string, Time-limited URL to get the .ipynb file for this notebook.

`notebookPreviewUrl`
string, Time-limited URL to get the .htm preview file for this notebook.

`requirementsUrl`
string, Time-limited URL to get the requirements.txt file for this notebook.

`fileId` string, The file ID for the S3 file containing the .ipynb file.

`requirementsFileId`
string, The file ID for the S3 file containing the requirements.txt file.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`dockerImageName`
string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).

`instanceType` string, The EC2 instance type to deploy to.

`memory` integer, The amount of memory allocated to the notebook.

`cpu` integer, The amount of cpu allocated to the the notebook.

`createdAt` string,

`updatedAt` string,

`mostRecentDeployment`
list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.

- createdAt string,
- updatedAt string,
- notebookId integer, The ID of the owning Notebook

`credentials` array, A list of credential IDs to pass to the notebook.

`environmentVariables` list, Environment variables to be passed into the Notebook.

`idleTimeout` integer, How long the notebook will stay alive without any kernel activity.

`partitionLabel` string, The partition label used to run this object.

`gitRepoId` integer, The ID of the git repository.

`gitRepoUrl` string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`gitRef` string, The git reference if git repo is specified

`gitPath` string, The path to the .ipynb file in the git repo that will be started up on notebook launch

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`hidden` boolean, The hidden status of the item.

`notebooks_get_deployments`

Get details about a Notebook deployment

Description

Get details about a Notebook deployment

Usage

```
notebooks_get_deployments(notebook_id, deployment_id)
```

Arguments

`notebook_id`	integer required. The ID of the owning Notebook
`deployment_id`	integer required. The ID for this deployment

Value

A list containing the following elements:

`deploymentId`	integer, The ID for this deployment.
`userId`	integer, The ID of the owner.
`host`	string, Domain of the deployment.
`name`	string, Name of the deployment.

dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
displayUrl	string, A signed URL for viewing the deployed item.
instanceType	string, The EC2 instance type requested for the deployment.
memory	integer, The memory allocated to the deployment, in MB.
cpu	integer, The cpu allocated to the deployment, in millicores.
state	string, The state of the deployment.
stateMessage	string, A detailed description of the state.
maxMemoryUsage	number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
maxCpuUsage	number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
createdAt	string,
updatedAt	string,
notebookId	integer, The ID of the owning Notebook

`notebooks_get_git_commits`
Get file contents at git ref

Description

Get file contents at git ref

Usage

```
notebooks_get_git_commits(id, commit_hash)
```

Arguments

id	integer required. The ID of the item.
commit_hash	string required. The SHA (full or shortened) of the desired git commit.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

`notebooks_list` *List Notebooks*

Description

List Notebooks

Usage

```
notebooks_list(
  hidden = NULL,
  archived = NULL,
  author = NULL,
  status = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

`hidden`	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
`archived`	string optional. The archival status of the requested item(s).
`author`	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.
`status`	string optional. If specified, returns notebooks with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'pending', 'idle'.
`limit`	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
`page_num`	integer optional. Page number of the results to return. Defaults to the first page, 1.
`order`	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at.
`order_dir`	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

`id`	integer, The ID for this notebook.
`name`	string, The name of this notebook.

`language` string, The kernel language of this notebook ("python3" or "r"). Defaults to "python3".

`description` string, The description of this notebook.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`createdAt` string,

`updatedAt` string,

`mostRecentDeployment`
list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- notebookId integer, The ID of the owning Notebook

`archived` string, The archival status of the requested item(s).

`notebooks_list_dependencies`
List dependent objects for this object

Description

List dependent objects for this object

Usage

```
notebooks_list_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

---

notebooks_list_deployments

List deployments for a Notebook

Description

List deployments for a Notebook

Usage

```
notebooks_list_deployments(
  notebook_id,
  deployment_id = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

notebook_id	integer required. The ID of the owning Notebook
deployment_id	integer optional. The ID for this deployment
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

deploymentId	integer, The ID for this deployment.
userId	integer, The ID of the owner.
host	string, Domain of the deployment.
name	string, Name of the deployment.
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
instanceType	string, The EC2 instance type requested for the deployment.
memory	integer, The memory allocated to the deployment, in MB.
cpu	integer, The cpu allocated to the deployment, in millicores.
state	string, The state of the deployment.
stateMessage	string, A detailed description of the state.
maxMemoryUsage	number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
maxCpuUsage	number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
createdAt	string,
updatedAt	string,
notebookId	integer, The ID of the owning Notebook

`notebooks_list_deployments_logs` *Get the logs for a Notebook deployment*

Description

Get the logs for a Notebook deployment

Usage

```
notebooks_list_deployments_logs(
  id,
  deployment_id,
  start_at = NULL,
  end_at = NULL,
  limit = NULL
)
```

Arguments

`id`	integer required. The ID of the owning Notebook.
`deployment_id`	integer required. The ID for this deployment.
`start_at`	string optional. Log entries with a lower timestamp will be omitted.
`end_at`	string optional. Log entries with a higher timestamp will be omitted.
`limit`	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

`message`	string, The log message.
`stream`	string, The stream of the log. One of "stdout", "stderr".
`createdAt`	string, The time the log was created.
`source`	string, The source of the log. One of "system", "user".

notebooks_list_git *Get the git metadata attached to an item*

Description

Get the git metadata attached to an item

Usage

```
notebooks_list_git(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,
gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

notebooks_list_git_commits *Get the git commits for an item on the current branch*

Description

Get the git commits for an item on the current branch

Usage

```
notebooks_list_git_commits(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

commitHash	string, The SHA of the commit.
authorName	string, The name of the commit's author.
date	string, The commit's timestamp.
message	string, The commit message.

---

notebooks_list_projects

List the projects a Notebook belongs to

Description

List the projects a Notebook belongs to

Usage

```
notebooks_list_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the Notebook.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`autoShare` boolean,

`createdAt` string,

`updatedAt` string,

`archived` string, The archival status of the requested item(s).

`notebooks_list_shares` *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
notebooks_list_shares(id)
```

Arguments

`id` integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

notebooks_list_update_links *Get URLs to update notebook*

Description

Get URLs to update notebook

Usage

```
notebooks_list_update_links(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

updateUrl	string, Time-limited URL to PUT new contents of the .ipynb file for this notebook.
updatePreviewUrl	string, Time-limited URL to PUT new contents of the .htm preview file for this notebook.

notebooks_patch *Update some attributes of this Notebook*

Description

Update some attributes of this Notebook

Usage

```
notebooks_patch(
  id,
  name = NULL,
  language = NULL,
  description = NULL,
  file_id = NULL,
  requirements_file_id = NULL,
  requirements = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  instance_type = NULL,
```

```
  memory = NULL,
  cpu = NULL,
  credentials = NULL,
  environment_variables = NULL,
  idle_timeout = NULL,
  partition_label = NULL,
  git_repo_url = NULL,
  git_ref = NULL,
  git_path = NULL
)
```

Arguments

id	integer required. The ID for this notebook.
name	string optional. The name of this notebook.
language	string optional. The kernel language of this notebook ("python3" or "r"). Defaults to "python3".
description	string optional. The description of this notebook.
file_id	string optional. The file ID for the S3 file containing the .ipynb file.
requirements_file_id	string optional. The file ID for the S3 file containing the requirements.txt file.
requirements	string optional. The requirements txt file.
docker_image_name	string optional. The name of the docker image to pull from DockerHub.
docker_image_tag	string optional. The tag of the docker image to pull from DockerHub (default: latest).
instance_type	string optional. The EC2 instance type to deploy to.
memory	integer optional. The amount of memory allocated to the notebook.
cpu	integer optional. The amount of cpu allocated to the the notebook.
credentials	array optional. A list of credential IDs to pass to the notebook.
environment_variables	list optional. Environment variables to be passed into the Notebook.
idle_timeout	integer optional. How long the notebook will stay alive without any kernel activity.
partition_label	string optional. The partition label used to run this object.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref	string optional. The git reference if git repo is specified
git_path	string optional. The path to the .ipynb file in the git repo that will be started up on notebook launch

Value

A list containing the following elements:

`id` integer, The ID for this notebook.

`name` string, The name of this notebook.

`language` string, The kernel language of this notebook ("python3" or "r"). Defaults to "python3".

`description` string, The description of this notebook.

`notebookUrl` string, Time-limited URL to get the .ipynb file for this notebook.

`notebookPreviewUrl` string, Time-limited URL to get the .htm preview file for this notebook.

`requirementsUrl` string, Time-limited URL to get the requirements.txt file for this notebook.

`fileId` string, The file ID for the S3 file containing the .ipynb file.

`requirementsFileId` string, The file ID for the S3 file containing the requirements.txt file.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`dockerImageName` string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).

`instanceType` string, The EC2 instance type to deploy to.

`memory` integer, The amount of memory allocated to the notebook.

`cpu` integer, The amount of cpu allocated to the the notebook.

`createdAt` string,

`updatedAt` string,

`mostRecentDeployment` list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.

- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- notebookId integer, The ID of the owning Notebook

`credentials` array, A list of credential IDs to pass to the notebook.

`environmentVariables` list, Environment variables to be passed into the Notebook.

`idleTimeout` integer, How long the notebook will stay alive without any kernel activity.

`partitionLabel` string, The partition label used to run this object.

`gitRepoId` integer, The ID of the git repository.

`gitRepoUrl` string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`gitRef` string, The git reference if git repo is specified

`gitPath` string, The path to the .ipynb file in the git repo that will be started up on notebook launch

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`hidden` boolean, The hidden status of the item.

`notebooks_patch_git` *Update an attached git file*

Description

Update an attached git file

Usage

```
notebooks_patch_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,
gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

---

notebooks_post *Create a Notebook*

Description

Create a Notebook

Usage

```
notebooks_post(
  name = NULL,
  language = NULL,
  description = NULL,
  file_id = NULL,
  requirements_file_id = NULL,
```

```
  requirements = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  instance_type = NULL,
  memory = NULL,
  cpu = NULL,
  credentials = NULL,
  environment_variables = NULL,
  idle_timeout = NULL,
  partition_label = NULL,
  git_repo_url = NULL,
  git_ref = NULL,
  git_path = NULL,
  hidden = NULL
)
```

Arguments

name	string optional. The name of this notebook.
language	string optional. The kernel language of this notebook ("python3" or "r"). Defaults to "python3".
description	string optional. The description of this notebook.
file_id	string optional. The file ID for the S3 file containing the .ipynb file.
requirements_file_id	string optional. The file ID for the S3 file containing the requirements.txt file.
requirements	string optional. The requirements txt file.
docker_image_name	string optional. The name of the docker image to pull from DockerHub.
docker_image_tag	string optional. The tag of the docker image to pull from DockerHub (default: latest).
instance_type	string optional. The EC2 instance type to deploy to.
memory	integer optional. The amount of memory allocated to the notebook.
cpu	integer optional. The amount of cpu allocated to the the notebook.
credentials	array optional. A list of credential IDs to pass to the notebook.
environment_variables	list optional. Environment variables to be passed into the Notebook.
idle_timeout	integer optional. How long the notebook will stay alive without any kernel activity.
partition_label	string optional. The partition label used to run this object.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref	string optional. The git reference if git repo is specified
git_path	string optional. The path to the .ipynb file in the git repo that will be started up on notebook launch
hidden	boolean optional. The hidden status of the item.

Value

A list containing the following elements:

- `id` integer, The ID for this notebook.
- `name` string, The name of this notebook.
- `language` string, The kernel language of this notebook ("python3" or "r"). Defaults to "python3".
- `description` string, The description of this notebook.
- `notebookUrl` string, Time-limited URL to get the .ipynb file for this notebook.
- `notebookPreviewUrl` string, Time-limited URL to get the .htm preview file for this notebook.
- `requirementsUrl` string, Time-limited URL to get the requirements.txt file for this notebook.
- `fileId` string, The file ID for the S3 file containing the .ipynb file.
- `requirementsFileId` string, The file ID for the S3 file containing the requirements.txt file.
- `user` list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.
- `dockerImageName` string, The name of the docker image to pull from DockerHub.
- `dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).
- `instanceType` string, The EC2 instance type to deploy to.
- `memory` integer, The amount of memory allocated to the notebook.
- `cpu` integer, The amount of cpu allocated to the the notebook.
- `createdAt` string,
- `updatedAt` string,
- `mostRecentDeployment` list, A list containing the following elements:
  - deploymentId integer, The ID for this deployment.
  - userId integer, The ID of the owner.
  - host string, Domain of the deployment.
  - name string, Name of the deployment.
  - dockerImageName string, The name of the docker image to pull from DockerHub.
  - dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
  - displayUrl string, A signed URL for viewing the deployed item.

- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- notebookId integer, The ID of the owning Notebook

`credentials` array, A list of credential IDs to pass to the notebook.

`environmentVariables` list, Environment variables to be passed into the Notebook.

`idleTimeout` integer, How long the notebook will stay alive without any kernel activity.

`partitionLabel` string, The partition label used to run this object.

`gitRepoId` integer, The ID of the git repository.

`gitRepoUrl` string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`gitRef` string, The git reference if git repo is specified

`gitPath` string, The path to the .ipynb file in the git repo that will be started up on notebook launch

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`hidden` boolean, The hidden status of the item.

`notebooks_post_clone` *Clone this Notebook*

Description

Clone this Notebook

Usage

```
notebooks_post_clone(id)
```

Arguments

`id` integer required.

Value

A list containing the following elements:

`id` integer, The ID for this notebook.

`name` string, The name of this notebook.

`language` string, The kernel language of this notebook ("python3" or "r"). Defaults to "python3".

`description` string, The description of this notebook.

`notebookUrl` string, Time-limited URL to get the .ipynb file for this notebook.

`notebookPreviewUrl` string, Time-limited URL to get the .htm preview file for this notebook.

`requirementsUrl` string, Time-limited URL to get the requirements.txt file for this notebook.

`fileId` string, The file ID for the S3 file containing the .ipynb file.

`requirementsFileId` string, The file ID for the S3 file containing the requirements.txt file.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`dockerImageName` string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).

`instanceType` string, The EC2 instance type to deploy to.

`memory` integer, The amount of memory allocated to the notebook.

`cpu` integer, The amount of cpu allocated to the the notebook.

`createdAt` string,

`updatedAt` string,

`mostRecentDeployment` list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.

- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- notebookId integer, The ID of the owning Notebook

`credentials` array, A list of credential IDs to pass to the notebook.

`environmentVariables` list, Environment variables to be passed into the Notebook.

`idleTimeout` integer, How long the notebook will stay alive without any kernel activity.

`partitionLabel` string, The partition label used to run this object.

`gitRepoId` integer, The ID of the git repository.

`gitRepoUrl` string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`gitRef` string, The git reference if git repo is specified

`gitPath` string, The path to the .ipynb file in the git repo that will be started up on notebook launch

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`hidden` boolean, The hidden status of the item.

`notebooks_post_deployments` *Deploy a Notebook*

Description

Deploy a Notebook

Usage

```
notebooks_post_deployments(notebook_id, deployment_id = NULL)
```

Arguments

`notebook_id` integer required. The ID of the owning Notebook

`deployment_id` integer optional. The ID for this deployment

Value

A list containing the following elements:

deploymentId	integer, The ID for this deployment.
userId	integer, The ID of the owner.
host	string, Domain of the deployment.
name	string, Name of the deployment.
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
displayUrl	string, A signed URL for viewing the deployed item.
instanceType	string, The EC2 instance type requested for the deployment.
memory	integer, The memory allocated to the deployment, in MB.
cpu	integer, The cpu allocated to the deployment, in millicores.
state	string, The state of the deployment.
stateMessage	string, A detailed description of the state.
maxMemoryUsage	number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
maxCpuUsage	number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
createdAt	string,
updatedAt	string,
notebookId	integer, The ID of the owning Notebook

---

notebooks_post_git_commits

Commit and push a new version of the file

Description

Commit and push a new version of the file

Usage

```
notebooks_post_git_commits(id, content, message, file_hash)
```

Arguments

id	integer required. The ID of the item.
content	string required. The contents to commit to the file.
message	string required. A commit message describing the changes being made.
file_hash	string required. The full SHA of the file being replaced.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

`notebooks_put` *Replace all attributes of this Notebook*

Description

Replace all attributes of this Notebook

Usage

```
notebooks_put(
  id,
  name = NULL,
  language = NULL,
  description = NULL,
  file_id = NULL,
  requirements_file_id = NULL,
  requirements = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  instance_type = NULL,
  memory = NULL,
  cpu = NULL,
  credentials = NULL,
  environment_variables = NULL,
  idle_timeout = NULL,
  partition_label = NULL,
  git_repo_url = NULL,
  git_ref = NULL,
  git_path = NULL
)
```

Arguments

id	integer required. The ID for this notebook.
name	string optional. The name of this notebook.
language	string optional. The kernel language of this notebook ("python3" or "r"). Defaults to "python3".

`description` string optional. The description of this notebook.

`file_id` string optional. The file ID for the S3 file containing the .ipynb file.

`requirements_file_id`
string optional. The file ID for the S3 file containing the requirements.txt file.

`requirements` string optional. The requirements txt file.

`docker_image_name`
string optional. The name of the docker image to pull from DockerHub.

`docker_image_tag`
string optional. The tag of the docker image to pull from DockerHub (default: latest).

`instance_type` string optional. The EC2 instance type to deploy to.

`memory` integer optional. The amount of memory allocated to the notebook.

`cpu` integer optional. The amount of cpu allocated to the the notebook.

`credentials` array optional. A list of credential IDs to pass to the notebook.

`environment_variables`
list optional. Environment variables to be passed into the Notebook.

`idle_timeout` integer optional. How long the notebook will stay alive without any kernel activity.

`partition_label`
string optional. The partition label used to run this object.

`git_repo_url` string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`git_ref` string optional. The git reference if git repo is specified

`git_path` string optional. The path to the .ipynb file in the git repo that will be started up on notebook launch

Value

A list containing the following elements:

`id` integer, The ID for this notebook.

`name` string, The name of this notebook.

`language` string, The kernel language of this notebook ("python3" or "r"). Defaults to "python3".

`description` string, The description of this notebook.

`notebookUrl` string, Time-limited URL to get the .ipynb file for this notebook.

`notebookPreviewUrl`
string, Time-limited URL to get the .htm preview file for this notebook.

`requirementsUrl`
string, Time-limited URL to get the requirements.txt file for this notebook.

`fileId` string, The file ID for the S3 file containing the .ipynb file.

`requirementsFileId`
string, The file ID for the S3 file containing the requirements.txt file.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`dockerImageName`
string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).

`instanceType` string, The EC2 instance type to deploy to.

`memory` integer, The amount of memory allocated to the notebook.

`cpu` integer, The amount of cpu allocated to the the notebook.

`createdAt` string,

`updatedAt` string,

`mostRecentDeployment`
list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- notebookId integer, The ID of the owning Notebook

`credentials` array, A list of credential IDs to pass to the notebook.

`environmentVariables`
list, Environment variables to be passed into the Notebook.

`idleTimeout` integer, How long the notebook will stay alive without any kernel activity.

partitionLabel	string, The partition label used to run this object.
gitRepoId	integer, The ID of the git repository.
gitRepoUrl	string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
gitRef	string, The git reference if git repo is specified
gitPath	string, The path to the .ipynb file in the git repo that will be started up on notebook launch
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
archived	string, The archival status of the requested item(s).
hidden	boolean, The hidden status of the item.

`notebooks_put_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
notebooks_put_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ID for this notebook.
name	string, The name of this notebook.
language	string, The kernel language of this notebook ("python3" or "r"). Defaults to "python3".
description	string, The description of this notebook.
notebookUrl	string, Time-limited URL to get the .ipynb file for this notebook.
notebookPreviewUrl	string, Time-limited URL to get the .htm preview file for this notebook.
requirementsUrl	string, Time-limited URL to get the requirements.txt file for this notebook.
fileId	string, The file ID for the S3 file containing the .ipynb file.
requirementsFileId	string, The file ID for the S3 file containing the requirements.txt file.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`dockerImageName` string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).

`instanceType` string, The EC2 instance type to deploy to.

`memory` integer, The amount of memory allocated to the notebook.

`cpu` integer, The amount of cpu allocated to the the notebook.

`createdAt` string,

`updatedAt` string,

`mostRecentDeployment` list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- notebookId integer, The ID of the owning Notebook

`credentials` array, A list of credential IDs to pass to the notebook.

`environmentVariables` list, Environment variables to be passed into the Notebook.

`idleTimeout` integer, How long the notebook will stay alive without any kernel activity.

partitionLabel	string, The partition label used to run this object.
gitRepoId	integer, The ID of the git repository.
gitRepoUrl	string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
gitRef	string, The git reference if git repo is specified
gitPath	string, The path to the .ipynb file in the git repo that will be started up on notebook launch
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
archived	string, The archival status of the requested item(s).
hidden	boolean, The hidden status of the item.

`notebooks_put_git` *Attach an item to a file in a git repo*

Description

Attach an item to a file in a git repo

Usage

```
notebooks_put_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,
gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

---

notebooks_put_projects

Add a Notebook to a project

Description

Add a Notebook to a project

Usage

```
notebooks_put_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Notebook.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

`notebooks_put_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
notebooks_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`notebooks_put_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
notebooks_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_ids`	array required. An array of one or more user IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`notebooks_put_transfer`
Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
notebooks_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies`
: array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object

- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`notifications_list` *Receive a stream of notifications as they come in*

Description

Receive a stream of notifications as they come in

Usage

```
notifications_list(last_event_id = NULL, r = NULL, mock = NULL)
```

Arguments

`last_event_id`	string optional. allows browser to keep track of last event fired
`r`	string optional. specifies retry/reconnect timeout
`mock`	string optional. used for testing

Value

An empty HTTP response

`ontology_list` *List the ontology of column names Civis uses*

Description

List the ontology of column names Civis uses

Usage

```
ontology_list(subset = NULL)
```

Arguments

`subset`	string optional. A subset of fields to return.

Value

An array containing the following fields:

key	string,
title	string,
desc	string, A description of this field.
aliases	array,

---

organizations_delete_favorites
Unfavorite an item for your organization

Description

Unfavorite an item for your organization

Usage

```
organizations_delete_favorites(id)
```

Arguments

id	integer required. The id of the favorite.

Value

An empty HTTP response

---

organizations_list_favorites
List Favorites

Description

List Favorites

Usage

```
organizations_list_favorites(
  object_id = NULL,
  object_type = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

object_id	integer optional. The id of the object. If specified as a query parameter, must also specify object_type parameter.
object_type	string optional. The type of the object that is favorited. Valid options: Container Script, Identity Resolution, Import, Python Script, R Script, dbt Script, JavaScript Script, SQL Script, Template Script, Project, Workflow, Table, Tableau Report, Service Report, HTML Report, SQL Report
limit	integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, object_type, object_id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The id of the favorite.
objectId	integer, The id of the object. If specified as a query parameter, must also specify object_type parameter.
objectType	string, The type of the object that is favorited. Valid options: Container Script, Identity Resolution, Import, Python Script, R Script, dbt Script, JavaScript Script, SQL Script, Template Script, Project, Workflow, Table, Tableau Report, Service Report, HTML Report, SQL Report
objectName	string, The name of the object that is favorited.
createdAt	string, The time this favorite was created.
objectUpdatedAt	string, The time the object that is favorited was last updated
objectAuthor	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
position	integer, The rank position of this favorite. Use the patch users/me/favorites/:id/ranking/ endpoints to update.

`organizations_post_favorites` *Favorite an item for your organization*

Description

Favorite an item for your organization

Usage

```
organizations_post_favorites(object_id, object_type)
```

Arguments

`object_id`	integer required. The id of the object. If specified as a query parameter, must also specify object_type parameter.
`object_type`	string required. The type of the object that is favorited. Valid options: Container Script, Identity Resolution, Import, Python Script, R Script, dbt Script, JavaScript Script, SQL Script, Template Script, Project, Workflow, Table, Tableau Report, Service Report, HTML Report, SQL Report

Value

A list containing the following elements:

`id`	integer, The id of the favorite.
`objectId`	integer, The id of the object. If specified as a query parameter, must also specify object_type parameter.
`objectType`	string, The type of the object that is favorited. Valid options: Container Script, Identity Resolution, Import, Python Script, R Script, dbt Script, JavaScript Script, SQL Script, Template Script, Project, Workflow, Table, Tableau Report, Service Report, HTML Report, SQL Report
`objectName`	string, The name of the object that is favorited.
`createdAt`	string, The time this favorite was created.
`objectUpdatedAt`	string, The time the object that is favorited was last updated
`objectAuthor`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.

permission_sets_delete_resources

Delete a resource in a permission set

Description

Delete a resource in a permission set

Usage

```
permission_sets_delete_resources(id, name)
```

Arguments

id	integer required. The ID for this permission set.
name	string required. The name of this resource.

Value

An empty HTTP response

permission_sets_delete_resources_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
permission_sets_delete_resources_shares_groups(id, name, group_id)
```

Arguments

id	integer required. The ID for this permission set.
name	string required. The name of this resource.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

permission_sets_delete_resources_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
permission_sets_delete_resources_shares_users(id, name, user_id)
```

Arguments

id	integer required. The ID for this permission set.
name	string required. The name of this resource.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

permission_sets_delete_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
permission_sets_delete_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

permission_sets_delete_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
permission_sets_delete_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

permission_sets_get *Get a Permission Set*

Description

Get a Permission Set

Usage

```
permission_sets_get(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for this permission set.
name	string, The name of this permission set.
description	string, A description of this permission set.
author	list, A list containing the following elements: • id integer, The ID of this user.

- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`createdAt` string,

`updatedAt` string,

`archived` string, The archival status of the requested item(s).

`permission_sets_get_resources` *Get a resource in a permission set*

Description

Get a resource in a permission set

Usage

```
permission_sets_get_resources(id, name)
```

Arguments

`id`	integer required. The ID for this permission set.
`name`	string required. The name of this resource.

Value

A list containing the following elements:

`permissionSetId` integer, The ID for the permission set this resource belongs to.

`name` string, The name of this resource.

`description` string, A description of this resource.

`createdAt` string,

`updatedAt` string,

`permission_sets_list` *List Permission Sets*

Description

List Permission Sets

Usage

```
permission_sets_list(
  archived = NULL,
  author = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

`archived`	string optional. The archival status of the requested item(s).
`author`	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.
`limit`	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
`page_num`	integer optional. Page number of the results to return. Defaults to the first page, 1.
`order`	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at.
`order_dir`	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

`id`	integer, The ID for this permission set.
`name`	string, The name of this permission set.
`description`	string, A description of this permission set.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials.

- online boolean, Whether this user is online.

createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

`permission_sets_list_dependencies`
List dependent objects for this object

Description

List dependent objects for this object

Usage

```
permission_sets_list_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

permission_sets_list_resources
List resources in a permission set

Description

List resources in a permission set

Usage

```
permission_sets_list_resources(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID for this permission set.
limit	integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to name. Must be one of: name, id, updated_at, created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

permissionSetId	integer, The ID for the permission set this resource belongs to.
name	string, The name of this resource.
description	string, A description of this resource.
createdAt	string,
updatedAt	string,

`permission_sets_list_resources_shares` *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
permission_sets_list_resources_shares(id, name)
```

Arguments

`id`	integer required. The ID for this permission set.
`name`	string required. The name of this resource.

Value

An array containing the following fields:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
`totalGroupShares`	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

permission_sets_list_shares
List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
permission_sets_list_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers — list, A list containing the following elements:

- users array,
- groups array,

writers — list, A list containing the following elements:

- users array,
- groups array,

owners — list, A list containing the following elements:

- users array,
- groups array,

totalUserShares — integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares — integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

permission_sets_list_users_permissions *Get all permissions for a user, in this permission set*

Description

Get all permissions for a user, in this permission set

Usage

```
permission_sets_list_users_permissions(id, user_id)
```

Arguments

id	integer required. The ID for this permission set.
user_id	integer required. The ID for the user.

Value

An array containing the following fields:

resourceName	string, The name of the resource.
read	boolean, If true, the user has read permission on this resource.
write	boolean, If true, the user has write permission on this resource.
manage	boolean, If true, the user has manage permission on this resource.

permission_sets_patch *Update some attributes of this Permission Set*

Description

Update some attributes of this Permission Set

Usage

```
permission_sets_patch(id, name = NULL, description = NULL)
```

Arguments

id	integer required. The ID for this permission set.
name	string optional. The name of this permission set.
description	string optional. A description of this permission set.

Value

A list containing the following elements:

id	integer, The ID for this permission set.
name	string, The name of this permission set.
description	string, A description of this permission set.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

---

permission_sets_patch_resources

Update a resource in a permission set

Description

Update a resource in a permission set

Usage

```
permission_sets_patch_resources(id, name, description = NULL)
```

Arguments

id	integer required. The ID for this permission set.
name	string required. The name of this resource.
description	string optional. A description of this resource.

Value

A list containing the following elements:

permissionSetId	integer, The ID for the permission set this resource belongs to.
name	string, The name of this resource.
description	string, A description of this resource.
createdAt	string,
updatedAt	string,

`permission_sets_post` *Create a Permission Set*

Description

Create a Permission Set

Usage

```
permission_sets_post(name, description = NULL)
```

Arguments

`name`	string required. The name of this permission set.
`description`	string optional. A description of this permission set.

Value

A list containing the following elements:

`id`	integer, The ID for this permission set.
`name`	string, The name of this permission set.
`description`	string, A description of this permission set.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`createdAt`	string,
`updatedAt`	string,
`archived`	string, The archival status of the requested item(s).

permission_sets_post_resources *Create a resource in a permission set*

Description

Create a resource in a permission set

Usage

```
permission_sets_post_resources(id, name, description = NULL)
```

Arguments

id	integer required. The ID for this permission set.
name	string required. The name of this resource.
description	string optional. A description of this resource.

Value

A list containing the following elements:

permissionSetId	integer, The ID for the permission set this resource belongs to.
name	string, The name of this resource.
description	string, A description of this resource.
createdAt	string,
updatedAt	string,

permission_sets_put *Replace all attributes of this Permission Set*

Description

Replace all attributes of this Permission Set

Usage

```
permission_sets_put(id, name, description = NULL)
```

Arguments

id	integer required. The ID for this permission set.
name	string required. The name of this permission set.
description	string optional. A description of this permission set.

Value

A list containing the following elements:

id	integer, The ID for this permission set.
name	string, The name of this permission set.
description	string, A description of this permission set.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

`permission_sets_put_archive`
Update the archive status of this object

Description

Update the archive status of this object

Usage

```
permission_sets_put_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ID for this permission set.
name	string, The name of this permission set.
description	string, A description of this permission set.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name.

- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`createdAt` string,

`updatedAt` string,

`archived` string, The archival status of the requested item(s).

`permission_sets_put_resources_shares_groups`
Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
permission_sets_put_resources_shares_groups(
  id,
  name,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id` integer required. The ID for this permission set.

`name` string required. The name of this resource.

`group_ids` array required. An array of one or more group IDs.

`permission_level` string required. Options are: "read", "write", or "manage".

`share_email_body` string optional. Custom body text for e-mail sent on a share.

`send_shared_email` boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers` list, A list containing the following elements:

- users array,
- groups array,

writers list, A list containing the following elements:

- users array,
- groups array,

owners list, A list containing the following elements:

- users array,
- groups array,

totalUserShares
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`permission_sets_put_resources_shares_users`
Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

```
permission_sets_put_resources_shares_users(
  id,
  name,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id integer required. The ID for this permission set.

name string required. The name of this resource.

user_ids array required. An array of one or more user IDs.

permission_level
string required. Options are: "read", "write", or "manage".

share_email_body
string optional. Custom body text for e-mail sent on a share.

send_shared_email
boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- `readers` list, A list containing the following elements:
  - users array,
  - groups array,
- `writers` list, A list containing the following elements:
  - users array,
  - groups array,
- `owners` list, A list containing the following elements:
  - users array,
  - groups array,
- `totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- `totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`permission_sets_put_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
permission_sets_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

- `id` integer required. The ID of the resource that is shared.
- `group_ids` array required. An array of one or more group IDs.
- `permission_level` string required. Options are: "read", "write", or "manage".
- `share_email_body` string optional. Custom body text for e-mail sent on a share.
- `send_shared_email` boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- `readers` list, A list containing the following elements:
  - users array,
  - groups array,
- `writers` list, A list containing the following elements:
  - users array,
  - groups array,
- `owners` list, A list containing the following elements:
  - users array,
  - groups array,
- `totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- `totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`permission_sets_put_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
permission_sets_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

- `id` integer required. The ID of the resource that is shared.
- `user_ids` array required. An array of one or more user IDs.
- `permission_level` string required. Options are: "read", "write", or "manage".
- `share_email_body` string optional. Custom body text for e-mail sent on a share.
- `send_shared_email` boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- `readers` list, A list containing the following elements:
  - users array,
  - groups array,
- `writers` list, A list containing the following elements:
  - users array,
  - groups array,
- `owners` list, A list containing the following elements:
  - users array,
  - groups array,
- `totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- `totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`permission_sets_put_transfer` *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
permission_sets_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

- `id` integer required. The ID of the resource that is shared.
- `user_id` integer required. ID of target user
- `include_dependencies` boolean required. Whether or not to give manage permissions on all dependencies
- `email_body` string optional. Custom body text for e-mail sent on transfer.
- `send_email` boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies` array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`plot.civis_ml_classifier`
Decile plot for classification with civis_ml

Description

Decile plot for classification with civis_ml

Usage

```
## S3 method for class 'civis_ml_classifier'
plot(x, name = NULL, ...)
```

Arguments

`x`	`civis_ml` object.
`name`	Name of the class in a multiclass model to plot.
`...`	unused.

Value

A `ggplot2` plot object is returned invisibly.

plot.civis_ml_regressor *Y-yhat plot for regression with civis_ml*

Description

Y-yhat plot for regression with civis_ml

Usage

```
## S3 method for class 'civis_ml_regressor'
plot(x, ...)
```

Arguments

x	civis_ml object
...	unused

Value

A ggplot2 plot object is returned invisibly.

predictions_get *Show the specified prediction*

Description

Show the specified prediction

Usage

```
predictions_get(id)
```

Arguments

id	integer required. The ID of the prediction.

Value

A list containing the following elements:

id	integer, The ID of the prediction.
modelId	integer, The ID of the model used for this prediction.
scoredTableId	integer, The ID of the source table for this prediction.

`scoredTableName`
: string, The name of the source table for this prediction.

`outputTableName`
: string, The name of the output table for this prediction.

`state`
: string, The state of the last run of this prediction.

`error`
: string, The error, if any, of the last run of this prediction.

`startedAt`
: string, The start time of the last run of this prediction.

`finishedAt`
: string, The end time of the last run of this prediction.

`lastRun`
: list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`scoredTables`
: array, An array containing the following fields:

- id integer, The ID of the table with created predictions.
- schema string, The schema of table with created predictions.
- name string, The name of table with created predictions.
- createdAt string, The time when the table with created predictions was created.
- scoreStats array, An array of metrics on the created predictions.

`schedule`
: list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`limitingSQL`
: string, A SQL WHERE clause used to scope the rows to be predicted.

`primaryKey`
: array, The primary key or composite keys of the table being predicted.

`predictions_list` *List predictions*

Description

List predictions

Usage

```
predictions_list(model_id = NULL)
```

Arguments

`model_id`	integer optional. If specified, only return predictions associated with this model ID.

Value

An array containing the following fields:

`id`	integer, The ID of the prediction.
`modelId`	integer, The ID of the model used for this prediction.
`scoredTableId`	integer, The ID of the source table for this prediction.
`scoredTableName`	string, The name of the source table for this prediction.
`outputTableName`	string, The name of the output table for this prediction.
`state`	string, The state of the last run of this prediction.
`error`	string, The error, if any, of the last run of this prediction.
`startedAt`	string, The start time of the last run of this prediction.
`finishedAt`	string, The end time of the last run of this prediction.
`lastRun`	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.

predictions_list_schedules *Show the prediction schedule*

Description

Show the prediction schedule

Usage

```
predictions_list_schedules(id)
```

Arguments

id	integer required. ID of the prediction associated with this schedule.

Value

A list containing the following elements:

id	integer, ID of the prediction associated with this schedule.
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
scoreOnModelBuild	boolean, Whether the prediction will run after a rebuild of the associated model.

print.civis_api *Print results from a Civis API call*

Description

Print results from a Civis API call

Usage

```
## S3 method for class 'civis_api'
print(x, ...)
```

Arguments

x	A civis_api response.
...	Further arguments passed to str

Value

The ‘civis_api‘ object, invisibly.

Examples

```
## Not run:
query_civis("SELECT * FROM schema.my_table", "database", preview_rows = 100)
csv_to_civis("file.csv", "my_database", "schema.my_table")
print(databases_list(), max = 5)

## End(Not run)
```

---

projects_delete *Archive a project (deprecated, use the /archive endpoint instead)*

Description

Archive a project (deprecated, use the /archive endpoint instead)

Usage

```
projects_delete(project_id)
```

Arguments

project_id	integer required.

Value

An empty HTTP response

`projects_delete_parent_projects`
Remove an item from a Parent Project

Description

Remove an item from a Parent Project

Usage

```
projects_delete_parent_projects(id, parent_project_id)
```

Arguments

`id`	integer required. The ID of the item.
`parent_project_id`	integer required. The ID of the Parent Project.

Value

An empty HTTP response

`projects_delete_shares_groups`
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
projects_delete_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

projects_delete_shares_users *Revoke the permissions a user has on this object*

Description

Revoke the permissions a user has on this object

Usage

```
projects_delete_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

projects_get *Get a detailed view of a project and the objects in it*

Description

Get a detailed view of a project and the objects in it

Usage

```
projects_get(project_id)
```

Arguments

project_id	integer required.

Value

A list containing the following elements:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username.

- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of this project.

`description` string, A description of the project.

`users` array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`autoShare` boolean,

`createdAt` string,

`updatedAt` string,

`tables` array, An array containing the following fields:

- schema string,
- name string,
- rowCount integer,
- columnCount integer,
- createdAt string,
- updatedAt string,

`surveys` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,

`scripts` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

`imports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,

- state string,
- lastRun object,

`exports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

`models` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

`notebooks` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

`services` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

`workflows` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,
- lastExecution object,

`reports` array, An array containing the following fields:

- id integer, The item's ID.

- createdAt string,
- updatedAt string,
- name string,
- state string,

`scriptTemplates`
array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,

`files` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- fileName string,
- fileSize integer,
- expired boolean,

`enhancements` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- lastRun object,

`projects` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- description string,

`allObjects` array, An array containing the following fields:

- projectId integer,
- objectId integer,
- objectType string,
- fcoType string,
- subType string,
- name string,
- icon string,
- author string,
- updatedAt string,
- autoShare boolean,
- archived string, The archival status of the requested item(s).

- hidden boolean, The hidden status of the item.
- myPermissionLevel string, Your permission level on the object. One of "read", "write", or "manage".

`note` string,

`canCurrentUserEnableAutoShare`
boolean, A flag for if the current user can enable auto-sharing mode for this project.

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`parentProject` list, A list containing the following elements:

- id integer, The parent project's ID.
- name integer, The parent project's name.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`projects_list` *List projects*

Description

List projects

Usage

```
projects_list(
  permission = NULL,
  auto_share = NULL,
  author = NULL,
  hidden = NULL,
  archived = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

`permission`	string optional. A permissions string, one of "read", "write", or "manage". Lists only projects for which the current user has that permission.
`auto_share`	boolean optional. Used to filter projects based on whether the project is autoshare enabled or not.
`author`	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.

hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
archived	string optional. The archival status of the requested item(s).
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 1000.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

projects_list_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
projects_list_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

projects_list_parent_projects

List the Parent Projects an item belongs to

Description

List the Parent Projects an item belongs to

Usage

```
projects_list_parent_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the item.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

---

projects_list_shares *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
projects_list_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`projects_post` *Create a project*

Description

Create a project

Usage

```
projects_post(name, description, note = NULL, auto_share = NULL, hidden = NULL)
```

Arguments

`name`	string required. The name of this project.
`description`	string required. A description of the project.
`note`	string optional. Notes for the project.
`auto_share`	boolean optional. If true, objects within the project will be automatically shared when the project is shared or objects are added.
`hidden`	boolean optional. The hidden status of the item.

Value

A list containing the following elements:

`id` integer, The ID for this project.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of this project.

`description` string, A description of the project.

`users` array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`autoShare` boolean,

`createdAt` string,

`updatedAt` string,

`tables` array, An array containing the following fields:

- schema string,
- name string,
- rowCount integer,
- columnCount integer,
- createdAt string,
- updatedAt string,

`surveys` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,

`scripts` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,

- lastRun object,

`imports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

`exports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

`models` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

`notebooks` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

`services` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

`workflows` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,
- lastExecution object,

`reports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

`scriptTemplates` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,

`files` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- fileName string,
- fileSize integer,
- expired boolean,

`enhancements` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- lastRun object,

`projects` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- description string,

`allObjects` array, An array containing the following fields:

- projectId integer,
- objectId integer,
- objectType string,

- fcoType string,
- subType string,
- name string,
- icon string,
- author string,
- updatedAt string,
- autoShare boolean,
- archived string, The archival status of the requested item(s).
- hidden boolean, The hidden status of the item.
- myPermissionLevel string, Your permission level on the object. One of "read", "write", or "manage".

`note` string,

`canCurrentUserEnableAutoShare`
boolean, A flag for if the current user can enable auto-sharing mode for this project.

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`parentProject` list, A list containing the following elements:

- id integer, The parent project's ID.
- name integer, The parent project's name.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`projects_post_clone` *Clone this*

Description

Clone this

Usage

```
projects_post_clone(id, clone_schedule = NULL, clone_notifications = NULL)
```

Arguments

`id` integer required. The ID for this project.

`clone_schedule` boolean optional. If true, also copy the schedule for all applicable project objects.

`clone_notifications`
boolean optional. If true, also copy the notifications for all applicable project objects.

Value

A list containing the following elements:

`id` integer, The ID for this project.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of this project.

`description` string, A description of the project.

`users` array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`autoShare` boolean,

`createdAt` string,

`updatedAt` string,

`tables` array, An array containing the following fields:

- schema string,
- name string,
- rowCount integer,
- columnCount integer,
- createdAt string,
- updatedAt string,

`surveys` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,

`scripts` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,

- lastRun object,

`imports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

`exports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

`models` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

`notebooks` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

`services` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

`workflows` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,
- lastExecution object,

`reports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

`scriptTemplates` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,

`files` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- fileName string,
- fileSize integer,
- expired boolean,

`enhancements` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- lastRun object,

`projects` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- description string,

`allObjects` array, An array containing the following fields:

- projectId integer,
- objectId integer,
- objectType string,

- fcoType string,
- subType string,
- name string,
- icon string,
- author string,
- updatedAt string,
- autoShare boolean,
- archived string, The archival status of the requested item(s).
- hidden boolean, The hidden status of the item.
- myPermissionLevel string, Your permission level on the object. One of "read", "write", or "manage".

`note` string,

`canCurrentUserEnableAutoShare`
boolean, A flag for if the current user can enable auto-sharing mode for this project.

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`parentProject` list, A list containing the following elements:

- id integer, The parent project's ID.
- name integer, The parent project's name.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`projects_put` *Update a project*

Description

Update a project

Usage

```
projects_put(project_id, name = NULL, description = NULL, note = NULL)
```

Arguments

`project_id`	integer required.
`name`	string optional. The name of this project.
`description`	string optional. A description of the project.
`note`	string optional. Notes for the project.

Value

A list containing the following elements:

`id` integer, The ID for this project.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of this project.

`description` string, A description of the project.

`users` array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`autoShare` boolean,

`createdAt` string,

`updatedAt` string,

`tables` array, An array containing the following fields:

- schema string,
- name string,
- rowCount integer,
- columnCount integer,
- createdAt string,
- updatedAt string,

`surveys` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,

`scripts` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,

- lastRun object,

`imports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

`exports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

`models` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

`notebooks` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

`services` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

`workflows` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,
- lastExecution object,

`reports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

`scriptTemplates` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,

`files` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- fileName string,
- fileSize integer,
- expired boolean,

`enhancements` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- lastRun object,

`projects` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- description string,

`allObjects` array, An array containing the following fields:

- projectId integer,
- objectId integer,
- objectType string,

- fcoType string,
- subType string,
- name string,
- icon string,
- author string,
- updatedAt string,
- autoShare boolean,
- archived string, The archival status of the requested item(s).
- hidden boolean, The hidden status of the item.
- myPermissionLevel string, Your permission level on the object. One of "read", "write", or "manage".

`note` string,

`canCurrentUserEnableAutoShare`
boolean, A flag for if the current user can enable auto-sharing mode for this project.

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`parentProject` list, A list containing the following elements:

- id integer, The parent project's ID.
- name integer, The parent project's name.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`projects_put_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
projects_put_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id` integer, The ID for this project.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of this project.

`description` string, A description of the project.

`users` array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`autoShare` boolean,

`createdAt` string,

`updatedAt` string,

`tables` array, An array containing the following fields:

- schema string,
- name string,
- rowCount integer,
- columnCount integer,
- createdAt string,
- updatedAt string,

`surveys` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,

`scripts` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,

- lastRun object,

`imports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

`exports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

`models` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

`notebooks` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

`services` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

`workflows` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,
- lastExecution object,

`reports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

`scriptTemplates`
array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,

`files` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- fileName string,
- fileSize integer,
- expired boolean,

`enhancements` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- lastRun object,

`projects` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- description string,

`allObjects` array, An array containing the following fields:

- projectId integer,
- objectId integer,
- objectType string,

- fcoType string,
- subType string,
- name string,
- icon string,
- author string,
- updatedAt string,
- autoShare boolean,
- archived string, The archival status of the requested item(s).
- hidden boolean, The hidden status of the item.
- myPermissionLevel string, Your permission level on the object. One of "read", "write", or "manage".

`note` string,

`canCurrentUserEnableAutoShare`
boolean, A flag for if the current user can enable auto-sharing mode for this project.

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`parentProject` list, A list containing the following elements:

- id integer, The parent project's ID.
- name integer, The parent project's name.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`projects_put_auto_share`
Enable or disable Auto-Share on a project

Description

Enable or disable Auto-Share on a project

Usage

```
projects_put_auto_share(project_id, auto_share)
```

Arguments

`project_id`	integer required.
`auto_share`	boolean required. A toggle for sharing the objects within the project when the project is shared or objects are added.

Value

A list containing the following elements:

`id` integer, The ID for this project.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of this project.

`description` string, A description of the project.

`users` array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`autoShare` boolean,

`createdAt` string,

`updatedAt` string,

`tables` array, An array containing the following fields:

- schema string,
- name string,
- rowCount integer,
- columnCount integer,
- createdAt string,
- updatedAt string,

`surveys` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,

`scripts` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,

- lastRun object,

`imports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

`exports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

`models` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

`notebooks` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

`services` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

`workflows` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,
- lastExecution object,

`reports` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

`scriptTemplates`
array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,

`files` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- fileName string,
- fileSize integer,
- expired boolean,

`enhancements` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- lastRun object,

`projects` array, An array containing the following fields:

- id integer, The item's ID.
- createdAt string,
- updatedAt string,
- name string,
- description string,

`allObjects` array, An array containing the following fields:

- projectId integer,
- objectId integer,
- objectType string,

- fcoType string,
- subType string,
- name string,
- icon string,
- author string,
- updatedAt string,
- autoShare boolean,
- archived string, The archival status of the requested item(s).
- hidden boolean, The hidden status of the item.
- myPermissionLevel string, Your permission level on the object. One of "read", "write", or "manage".

`note` string,

`canCurrentUserEnableAutoShare`
boolean, A flag for if the current user can enable auto-sharing mode for this project.

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`parentProject` list, A list containing the following elements:

- id integer, The parent project's ID.
- name integer, The parent project's name.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`projects_put_parent_projects`
Add an item to a Parent Project

Description

Add an item to a Parent Project

Usage

```
projects_put_parent_projects(id, parent_project_id)
```

Arguments

`id` integer required. The ID of the item.

`parent_project_id`
integer required. The ID of the Parent Project.

Value

An empty HTTP response

`projects_put_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
projects_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`projects_put_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
projects_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers
: list, A list containing the following elements:
  - users array,
  - groups array,

writers
: list, A list containing the following elements:
  - users array,
  - groups array,

owners
: list, A list containing the following elements:
  - users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`projects_put_transfer` *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
projects_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies`
: array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object

- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`publish_addin` *Add in to Publish an R Markdown file to Platform Reports from RStudio*

Description

This function is called when the "Publish To Civis" button is pressed in the RStudio IDE AddIn menu. It will attempt to save the current changes of the active document and push it to the appropriate location in Platform. RMarkdown files will be rendered to html with `publish_rmd` and published as a report. Currently only RMarkdown files are supported.

Usage

```
publish_addin()
```

See Also

`publish_rmd` for publishing RMarkdown files

`publish_html` *Publish HTML to Platform Reports*

Description

Publish HTML to Platform Reports

Usage

```
publish_html(
  html_file,
  report_id = NULL,
  report_name = NULL,
  provide_api_key = NULL,
  project_id = NULL
)
```

Arguments

html_file	string, HTML file
report_id	integer, ID of an existing report, if provided, the contents of the report will be replaced. If report_id is NULL, a new report will be created.
report_name	string, Title of report in platform
provide_api_key	bool, Set to true to include API key in report.
project_id	integer, Project_id that the report should be added to.

See Also

publish_rmd to publish an R Markdown document to Civis

Examples

```
## Not run:
# Uploads html and prints link to report in Platform
publish_html("my_beautiful_report.html")

## End(Not run)
```

---

publish_rmd *Publish an R Markdown file to Platform Reports*

Description

Publish an R Markdown file to Platform Reports

Usage

```
publish_rmd(
  rmd_file,
  report_id = NULL,
  report_name = NULL,
  provide_api_key = NULL,
  project_id = NULL,
  ...
)
```

Arguments

rmd_file	string, R Markdown file (.Rmd)
report_id	integer, ID of an existing report, if provided, the contents of the report will be replaced. If report_id is NULL, a new report will be created.
report_name	string, Title of report in platform

provide_api_key
: bool, Set to true to include API key in report.

project_id
: integer, Project_id that the report should be added to.

...
: additional parameters to send to `rmarkdown::render`. Note: A temporary file will be used for `output_file` if `output_file` is not explicitly set and `input` will be overwritten with `rmd_file`.

Details

This function also supports passing `report_id`, `report_name`, `provide_api_key` and `project_id` as metadata in the report's YAML front matter. Just as the title of an RMarkdown document can be set with `title: "my title!"`, these parameters can be set like

```
civis:
  report_name: "My Report Name"
  report_id: 9000
```

Since `report_id` is set, this code will overwrite the existing report with that number, which may be useful when updating a report on a schedule. Any argument passed in explicitly to `publish_rmd` will be used in place of the corresponding argument set in YAML metadata.

Note

`rmarkdown::render` depends on a recent version of `pandoc`. `pandoc` is distributed with RStudio and thus, `publish_rmd` will work in an RStudio environment. Outside of RStudio `pandoc` may be installed or `knitr::knit2html` can be used to convert an R Markdown document into html. The html can then be published to Civis with `publish_html`.

See Also

`publish_html` to publish html to Civis

Examples

```
## Not run:

# Publish a standard report
publish_rmd("my_beautiful_doc.Rmd")

# Publish a parameterized R Markdown document
# See: http://rmarkdown.rstudio.com/developer_parameterized_reports.html
params <- list("region" = "east", start = as.Date("2015-02-01"))
publish_rmd("my_parameterized_doc.Rmd", params=params)

## End(Not run)
```

`queries_delete` *Sets Query Hidden to true*

Description

Sets Query Hidden to true

Usage

```
queries_delete(id)
```

Arguments

`id`	integer required. The query ID.

Value

A list containing the following elements:

`id`	integer, The query ID.
`database`	integer, The database ID.
`sql`	string, The SQL to execute.
`credential`	integer, The credential ID.
`resultRows`	array, A preview of rows returned by the query.
`resultColumns`	array, A preview of columns returned by the query.
`error`	string, The error message for this run, if present.
`startedAt`	string, The start time of the last run.
`finishedAt`	string, The end time of the last run.
`state`	string, The state of the last run. One of queued, running, succeeded, failed, and cancelled.
`scriptId`	integer, The ID of the script associated with this query.
`exception`	string, Deprecated and not used.
`createdAt`	string,
`updatedAt`	string,
`lastRunId`	integer, The ID of the last run.
`hidden`	boolean, The hidden status of the item.
`archived`	string, The archival status of the requested item(s).
`name`	string, The name of the query.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name.

- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`reportId` integer, The ID of the report associated with this query.

`queries_delete_runs` *Cancel a run*

Description

Cancel a run

Usage

```
queries_delete_runs(id, run_id)
```

Arguments

`id`	integer required. The ID of the Query job.
`run_id`	integer required. The ID of the run.

Value

An empty HTTP response

`queries_get` *Get details about a query*

Description

Get details about a query

Usage

```
queries_get(id)
```

Arguments

`id`	integer required. The query ID.

Value

A list containing the following elements:

id	integer, The query ID.
database	integer, The database ID.
sql	string, The SQL to execute.
credential	integer, The credential ID.
resultRows	array, A preview of rows returned by the query.
resultColumns	array, A preview of columns returned by the query.
error	string, The error message for this run, if present.
startedAt	string, The start time of the last run.
finishedAt	string, The end time of the last run.
state	string, The state of the last run. One of queued, running, succeeded, failed, and cancelled.
scriptId	integer, The ID of the script associated with this query.
exception	string, Deprecated and not used.
createdAt	string,
updatedAt	string,
lastRunId	integer, The ID of the last run.
hidden	boolean, The hidden status of the item.
archived	string, The archival status of the requested item(s).
name	string, The name of the query.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
reportId	integer, The ID of the report associated with this query.

`queries_get_runs` *Check status of a run*

Description

Check status of a run

Usage

```
queries_get_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Query job.
run_id	integer required. The ID of the run.

Value

A list containing the following elements:

id	integer, The ID of the run.
queryId	integer, The ID of the Query job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

---

queries_list *List queries*

Description

List queries

Usage

```
queries_list(
  query = NULL,
  database_id = NULL,
  credential_id = NULL,
  author_id = NULL,
  created_before = NULL,
  created_after = NULL,
  started_before = NULL,
  started_after = NULL,
  state = NULL,
  exclude_results = NULL,
  hidden = NULL,
  archived = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

query	string optional. Space delimited query for searching queries by their SQL. Supports wild card characters "?" for any single character, and "*" for zero or more characters.
database_id	integer optional. The database ID.
credential_id	integer optional. The credential ID.
author_id	integer optional. The author of the query.
created_before	string optional. An upper bound for the creation date of the query.
created_after	string optional. A lower bound for the creation date of the query.
started_before	string optional. An upper bound for the start date of the last run.
started_after	string optional. A lower bound for the start date of the last run.
state	array optional. The state of the last run. One or more of queued, running, succeeded, failed, and cancelled. Specify multiple values as a comma-separated list (e.g., "A,B").
exclude_results	boolean optional. If true, does not return cached query results.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
archived	string optional. The archival status of the requested item(s).
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, started_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The query ID.
database	integer, The database ID.
sql	string, The SQL to execute.
credential	integer, The credential ID.
resultRows	array, A preview of rows returned by the query.
resultColumns	array, A preview of columns returned by the query.
error	string, The error message for this run, if present.
startedAt	string, The start time of the last run.
finishedAt	string, The end time of the last run.

state	string, The state of the last run. One of queued, running, succeeded, failed, and cancelled.
scriptId	integer, The ID of the script associated with this query.
exception	string, Deprecated and not used.
createdAt	string,
updatedAt	string,
lastRunId	integer, The ID of the last run.
archived	string, The archival status of the requested item(s).
previewRows	integer, The number of rows to save from the query's result (maximum: 1000).
reportId	integer, The ID of the report associated with this query.

`queries_list_runs` *List runs for the given Query job*

Description

List runs for the given Query job

Usage

```
queries_list_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the Query job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
queryId	integer, The ID of the Query job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

---

queries_list_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

```
queries_list_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the Query job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

queries_post *Execute a query*

Description

Execute a query

Usage

```
queries_post(
  database,
  sql,
  preview_rows,
  credential = NULL,
  hidden = NULL,
  interactive = NULL,
  include_header = NULL,
  compression = NULL,
  column_delimiter = NULL,
  unquoted = NULL,
  filename_prefix = NULL
)
```

Arguments

database	integer required. The database ID.
sql	string required. The SQL to execute.
preview_rows	integer required. The number of rows to save from the query's result (maximum: 1000).
credential	integer optional. The credential ID.
hidden	boolean optional. The hidden status of the item.
interactive	boolean optional. Deprecated and not used.
include_header	boolean optional. Whether the CSV output should include a header row [default: true].
compression	string optional. The type of compression. One of gzip or zip, or none [default: gzip].
column_delimiter	string optional. The delimiter to use. One of comma or tab, or pipe [default: comma].
unquoted	boolean optional. If true, will not quote fields.
filename_prefix	string optional. The output filename prefix.

Value

A list containing the following elements:

id	integer, The query ID.
database	integer, The database ID.
sql	string, The SQL to execute.
credential	integer, The credential ID.
resultRows	array, A preview of rows returned by the query.
resultColumns	array, A preview of columns returned by the query.
error	string, The error message for this run, if present.
startedAt	string, The start time of the last run.
finishedAt	string, The end time of the last run.
state	string, The state of the last run. One of queued, running, succeeded, failed, and cancelled.
scriptId	integer, The ID of the script associated with this query.
exception	string, Deprecated and not used.
createdAt	string,
updatedAt	string,
lastRunId	integer, The ID of the last run.
hidden	boolean, The hidden status of the item.
archived	string, The archival status of the requested item(s).
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
interactive	boolean, Deprecated and not used.
previewRows	integer, The number of rows to save from the query's result (maximum: 1000).
includeHeader	boolean, Whether the CSV output should include a header row [default: true].
compression	string, The type of compression. One of gzip or zip, or none [default: gzip].
columnDelimiter	string, The delimiter to use. One of comma or tab, or pipe [default: comma].
unquoted	boolean, If true, will not quote fields.
filenamePrefix	string, The output filename prefix.
reportId	integer, The ID of the report associated with this query.

queries_post_runs *Start a run*

Description

Start a run

Usage

```
queries_post_runs(id)
```

Arguments

id	integer required. The ID of the Query job.

Value

A list containing the following elements:

id	integer, The ID of the run.
queryId	integer, The ID of the Query job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

queries_put_scripts *Update the query's associated script*

Description

Update the query's associated script

Usage

```
queries_put_scripts(id, script_id)
```

Arguments

id	integer required. The query ID.
script_id	integer required. The ID of the script associated with this query.

Value

A list containing the following elements:

id	integer, The query ID.
database	integer, The database ID.
sql	string, The SQL to execute.
credential	integer, The credential ID.
resultRows	array, A preview of rows returned by the query.
resultColumns	array, A preview of columns returned by the query.
error	string, The error message for this run, if present.
startedAt	string, The start time of the last run.
finishedAt	string, The end time of the last run.
state	string, The state of the last run. One of queued, running, succeeded, failed, and cancelled.
scriptId	integer, The ID of the script associated with this query.
exception	string, Deprecated and not used.
createdAt	string,
updatedAt	string,
lastRunId	integer, The ID of the last run.
hidden	boolean, The hidden status of the item.
archived	string, The archival status of the requested item(s).
name	string, The name of the query.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
reportId	integer, The ID of the report associated with this query.

---

query_civis *Run a Query on Platform*

Description

Utility to run queries that return no output.

A default database can be set using `options(civis.default_db = "my_database")`. If there is only one database available, this database will automatically be used as the default.

Usage

```
query_civis(x, ...)

## S3 method for class 'sql'
query_civis(x, database = NULL, verbose = FALSE, ...)

## S3 method for class 'numeric'
query_civis(x, verbose = FALSE, ...)

## S3 method for class 'character'
query_civis(x, database = NULL, verbose = FALSE, ...)
```

Arguments

x	sql("..."), "query", or id of an existing sql script.
...	arguments passed to queries_post.
database	string, Name of database where query is run.
verbose	bool, Print detailed updates of job status.

Methods (by class)

- query_civis(sql): Run a SQL query.
- query_civis(numeric): Run a SQL query from a previous SQL query id.
- query_civis(character): Run a SQL query.

See Also

read_civis for downloading results of SQL scripts from Civis Platform as a data frame.

io

Other io: download_civis(), query_civis_file(), read_civis(), write_civis(), write_civis_file()

Examples

```
## Not run:
query_civis("GRANT ALL ON schema.my_table TO GROUP admin", "database", credential=0000)

## End(Not run)
```

query_civis_file *Export results from a query to S3 and return a file id.*

Description

Exports results from a Redshift SQL query, and returns the id of the file on S3 for use with read_civis or download_civis.

Usage

```
query_civis_file(x, ...)

## S3 method for class 'character'
query_civis_file(
  x,
  database = NULL,
  job_name = NULL,
  hidden = TRUE,
  verbose = verbose,
  csv_settings = NULL,
  ...
)

## S3 method for class 'sql'
query_civis_file(
  x,
  database = NULL,
  job_name = NULL,
  hidden = TRUE,
  verbose = FALSE,
  csv_settings = NULL,
  ...
)

## S3 method for class 'numeric'
query_civis_file(x, database = NULL, verbose = FALSE, ...)
```

Arguments

x	"schema.table", sql("query"), or a sql script job id.
...	Options passed to scripts_post_sql, including credential.
database	string, Name of database where data frame is to be uploaded. If no database is specified, uses options(civis.default_db).
job_name	string, Name of the job (default: "Civis S3 Export Via R Client").
hidden	bool, Whether the job is hidden.
verbose	bool, Set to TRUE to print intermediate progress indicators.
csv_settings	See scripts_post_sql for details.

Details

By default, the export uses the default csv_settings in scripts_post_sql, which is a gzipped csv.

Methods (by class)

- query_civis_file(character): Export a "schema.table" to a file id.
- query_civis_file(sql): Export results of a query to a file id.
- query_civis_file(numeric): Run an existing sql script and return the file id of the results on S3.

See Also

Other io: download_civis(), query_civis(), read_civis(), write_civis(), write_civis_file()

Examples

```
## Not run:
id <- query_civis_file("schema.tablename", database = "my_database")
df <- read_civis(id, using = read.csv)

query <- sql("SELECT * FROM table JOIN other_table USING id WHERE var1 < 23")
id <- query_civis_file(query)
df <- read_civis(id, using = read.csv)

id <- query_civis_file(query_id, credential_id = 0000)
df <- read_civis(id, using = read.csv)

## End(Not run)
```

---

read_civis *Read tables and files from Civis Platform*

Description

read_civis loads a table from Redshift as a data frame if given a "schema.table" or sql("query") as the first argument, or loads a file from Amazon S3 (the files endpoint) if a file id is given. Run outputs from any Civis platform script are returned if a civis_script is given.

A default database can be set using options(civis.default_db = "my_database"). If there is only one database available, this database will automatically be used as the default.

Usage

```
read_civis(x, ...)

## S3 method for class 'numeric'
read_civis(x, using = read.csv, verbose = FALSE, ...)
```

```
## S3 method for class 'character'
read_civis(x, database = NULL, ...)

## S3 method for class 'sql'
read_civis(
  x,
  database = NULL,
  using = utils::read.csv,
  job_name = NULL,
  hidden = TRUE,
  verbose = FALSE,
  ...
)

## S3 method for class 'civis_script'
read_civis(x, using, regex = NULL, ...)
```

Arguments

x	"schema.table", sql("query"), or a file id.
...	arguments passed to using.
using	function, Function to convert the file to a data frame or to unserialize. the file (e.g. read.csv or readRDS).
verbose	bool, Set to TRUE to print intermediate progress indicators.
database	string, Name of database where data frame is to be uploaded. If no database is specified, uses options(civis.default_db).
job_name	string, Name of the job (default: "Civis Export Via R Client").
hidden	bool, Whether the job is hidden.
regex	Regex of matching run output names.

Details

By default, read_civis.numeric assumes the file is a CSV. For reading a serialized R object, set using = readRDS for example.

If using = NULL, read_civis.civis_script will return all JSONValues with name matching regex. Otherwise all File run outputs matching regex will be read into memory with using. Results are always a named list. If the script has no outputs, an empty list will be returned.

Methods (by class)

- read_civis(numeric): Return a file as a data frame
- read_civis(character): Return all columns from a table as a data frame.
- read_civis(sql): Return a SQL query as a data frame.
- read_civis(civis_script): Return run outputs of a civis_script as a named list.

See Also

Other io: download_civis(), query_civis(), query_civis_file(), write_civis(), write_civis_file()

Examples

```
## Not run:
# Read all columns in a single table
df <- read_civis("schema.my_table", database = "my_database")

# Read data from a SQL select statement
query <- sql("SELECT * FROM table JOIN other_table USING id WHERE var1 < 23")
df <- read_civis(query, database = "my_database")

# Read an R object from the files endpoint.
id <- write_civis_file(df)
df <- read_civis(id)

# Read a text file or csv from the files endpoint.
id <- write_civis_file("my_csv.csv")
df <- read_civis(id)

# Read JSONValues from a civis script
vals <- read_civis(civis_script(1234))

# Read File run outputs from a civis script
df <- read_civis(civis_script(1234), regex = '.csv', using = read.csv)
obj <- read_civis(civis_script(1234), regex = '.rds', using = readRDS)

# Gracefully handle when read_civis.sql returns no rows
query <- sql("SELECT * FROM table WHERE 1 = 2")
mean_x <- tryCatch({
  df <- read_civis(query, database = "my_database")
  mean(df$x)
}, empty_result_error = function(e) {
   NA
})

## End(Not run)
```

---

refresh_table *Refresh a table*

Description

Refreshes a table on Redshift using tables_post_refresh, which runs the table scanner and updates table meta-data.

Usage

```
refresh_table(tablename, database = NULL, verbose = FALSE)
```

Arguments

tablename	string, Name of table and schema "schema.tablename"
database	string, Name of database where data frame is to be uploaded. If no database is specified, uses options(civis.default_db)
verbose	bool, Set to TRUE to print intermediate progress indicators.

Value

Returns table meta-data from tables_get.

See Also

Other tables: get_table_id(), transfer_table()

---

remote_hosts_delete_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
remote_hosts_delete_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

remote_hosts_delete_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
remote_hosts_delete_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

remote_hosts_get *Get a Remote Host*

Description

Get a Remote Host

Usage

```
remote_hosts_get(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID of the remote host.
name	string, The human readable name for the remote host.

`type` string, The type of remote host. One of: RemoteHostTypes::Bigquery, RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce

`url` string, The URL for the remote host.

`description` string, The description of the remote host.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`createdAt` string,

`updatedAt` string,

`remote_hosts_list` *List Remote Hosts*

Description

List Remote Hosts

Usage

```
remote_hosts_list(type = NULL)
```

Arguments

`type` string optional. The type of remote host. One of: RemoteHostTypes::Bigquery, RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce

Value

An array containing the following fields:

`id` integer, The ID of the remote host.

`name` string, The human readable name for the remote host.

type	string, The type of remote host. One of: RemoteHostTypes::Bigquery, RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce
url	string, The URL for the remote host.

`remote_hosts_list_data_sets`

List data sets available from a remote host

Description

List data sets available from a remote host

Usage

```
remote_hosts_list_data_sets(
  id,
  credential_id = NULL,
  username = NULL,
  password = NULL,
  q = NULL,
  s = NULL
)
```

Arguments

id	integer required. The ID of the remote host.
credential_id	integer optional. The credential ID.
username	string optional. The user name for remote host.
password	string optional. The password for remote host.
q	string optional. The query string for data set.
s	boolean optional. If true will only return schemas, otherwise, the results will be the full path.

Value

An array containing the following fields:

name	string, The path to a data_set.
fullPath	boolean, Boolean that indicates whether further querying needs to be done before the table can be selected.

`remote_hosts_list_shares` *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
remote_hosts_list_shares(id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`remote_hosts_patch` *Update some attributes of this Remote Host*

Description

Update some attributes of this Remote Host

Usage

```
remote_hosts_patch(
  id,
  name = NULL,
  type = NULL,
  url = NULL,
  description = NULL
)
```

Arguments

`id`	integer required. The ID of the remote host.
`name`	string optional. The human readable name for the remote host.
`type`	string optional. The type of remote host. One of: RemoteHostTypes::Bigquery, RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce
`url`	string optional. The URL for the remote host.
`description`	string optional. The description of the remote host.

Value

A list containing the following elements:

`id`	integer, The ID of the remote host.
`name`	string, The human readable name for the remote host.
`type`	string, The type of remote host. One of: RemoteHostTypes::Bigquery, RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce
`url`	string, The URL for the remote host.
`description`	string, The description of the remote host.
`myPermissionLevel`	string, Your permission level on the object. One of "read", "write", or "manage".

user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
createdAt	string,
updatedAt	string,

`remote_hosts_post` *Create a Remote Host*

Description

Create a Remote Host

Usage

```
remote_hosts_post(name, url, type)
```

Arguments

name	string required. The human readable name for the remote host.
url	string required. The URL for the remote host.
type	string required. The type of remote host. One of: RemoteHostTypes::Bigquery, RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce

Value

A list containing the following elements:

id	integer, The ID of the remote host.
name	string, The human readable name for the remote host.
type	string, The type of remote host. One of: RemoteHostTypes::Bigquery, RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce
url	string, The URL for the remote host.
description	string, The description of the remote host.

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`user`
: list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`createdAt`
: string,

`updatedAt`
: string,

`remote_hosts_post_authenticate`
: *Authenticate against a remote host using either a credential or a user name and password*

Description

Authenticate against a remote host using either a credential or a user name and password

Usage

```
remote_hosts_post_authenticate(
  id,
  credential_id = NULL,
  username = NULL,
  password = NULL
)
```

Arguments

`id`	integer required. The ID of the remote host.
`credential_id`	integer optional. The credential ID.
`username`	string optional. The user name for remote host.
`password`	string optional. The password for remote host.

Value

An empty HTTP response

`remote_hosts_put` *Replace all attributes of this Remote Host*

Description

Replace all attributes of this Remote Host

Usage

```
remote_hosts_put(id, name, type, url, description)
```

Arguments

`id`	integer required. The ID of the remote host.
`name`	string required. The human readable name for the remote host.
`type`	string required. The type of remote host. One of: RemoteHostTypes::Bigquery, RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce
`url`	string required. The URL for the remote host.
`description`	string required. The description of the remote host.

Value

A list containing the following elements:

`id`	integer, The ID of the remote host.
`name`	string, The human readable name for the remote host.
`type`	string, The type of remote host. One of: RemoteHostTypes::Bigquery, RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce
`url`	string, The URL for the remote host.
`description`	string, The description of the remote host.
`myPermissionLevel`	string, Your permission level on the object. One of "read", "write", or "manage".
`user`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.

`createdAt` string,

`updatedAt` string,

`remote_hosts_put_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
remote_hosts_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`remote_hosts_put_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
remote_hosts_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`
: integer required. The ID of the resource that is shared.

`user_ids`
: array required. An array of one or more user IDs.

`permission_level`
: string required. Options are: "read", "write", or "manage".

`share_email_body`
: string optional. Custom body text for e-mail sent on a share.

`send_shared_email`
: boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`
: list, A list containing the following elements:

- users array,
- groups array,

`writers`
: list, A list containing the following elements:

- users array,
- groups array,

owners list, A list containing the following elements:

- users array,
- groups array,

totalUserShares
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

reports_delete_grants *Revoke permission for this report to perform Civis platform API operations on your behalf*

Description

Revoke permission for this report to perform Civis platform API operations on your behalf

Usage

```
reports_delete_grants(id)
```

Arguments

id	integer required. The ID of this report.

Value

An empty HTTP response

---

reports_delete_projects *Remove a Report from a project*

Description

Remove a Report from a project

Usage

```
reports_delete_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Report.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

---

reports_delete_services_projects
Remove a Service Report from a project

Description

Remove a Service Report from a project

Usage

```
reports_delete_services_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Service Report.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

---

reports_delete_services_shares_groups
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
reports_delete_services_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

`reports_delete_services_shares_users`
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
reports_delete_services_shares_users(id, user_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. The ID of the user.

Value

An empty HTTP response

`reports_delete_shares_groups`
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
reports_delete_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

`reports_delete_shares_users`
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
reports_delete_shares_users(id, user_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. The ID of the user.

Value

An empty HTTP response

`reports_delete_sql_projects`
Remove a SQL Report from a project

Description

Remove a SQL Report from a project

Usage

```
reports_delete_sql_projects(id, project_id)
```

Arguments

`id`	integer required. The ID of the SQL Report.
`project_id`	integer required. The ID of the project.

Value

An empty HTTP response

`reports_delete_sql_shares_groups` *Revoke the permissions a group has on this object*

Description

Revoke the permissions a group has on this object

Usage

```
reports_delete_sql_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

`reports_delete_sql_shares_users` *Revoke the permissions a user has on this object*

Description

Revoke the permissions a user has on this object

Usage

```
reports_delete_sql_shares_users(id, user_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. The ID of the user.

Value

An empty HTTP response

`reports_get` *Get a single report*

Description

Get a single report

Usage

```
reports_get(id)
```

Arguments

`id`	integer required. The ID of this report.

Value

A list containing the following elements:

`id`	integer, The ID of this report.
`name`	string, The name of the report.
`user`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`createdAt`	string, The creation time for this report.
`updatedAt`	string, The last updated at time for this report.
`type`	string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL
`description`	string, The user-defined description of the report.
`myPermissionLevel`	string, Your permission level on the object. One of "read", "write", or "manage".
`archived`	string, The archival status of the requested item(s).
`projects`	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
`state`	string, The status of the report's last run.
`finishedAt`	string, The time that the report's last run finished.
`vizUpdatedAt`	string, The time that the report's visualization was last updated.
`script`	list, A list containing the following elements:

- id integer, The ID for the script.
- name string, The name of the script.
- sql string, The raw SQL query for the script, if applicable.

`jobPath` string, The link to details of the job that backs this report.

`tableauId` integer,

`templateId` integer, The ID of the template used for this report.

`authThumbnailUrl` string, URL for a thumbnail of the report.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`hidden` boolean, The hidden status of the item.

`authDataUrl` string, DEPRECATED: For legacy reports

`authCodeUrl` string, Link to code to render in the report.

`config` string, Any configuration metadata for this report.

`validOutputFile` boolean, Whether the job (a script or a query) that backs the report currently has a valid output file.

`provideAPIKey` boolean, Whether the report requests an API Key from the report viewer.

`apiKey` string, A Civis API key that can be used by this report.

`apiKeyId` integer, The ID of the API key. Can be used for auditing API use by this report.

`appState` list, Any application state blob for this report.

`useViewersTableauUsername` boolean, Apply user level filtering on Tableau reports.

`reports_get_git_commits` *Get file contents at git ref*

Description

Get file contents at git ref

Usage

```
reports_get_git_commits(id, commit_hash)
```

Arguments

id	integer required. The ID of the item.
commit_hash	string required. The SHA (full or shortened) of the desired git commit.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

---

reports_get_services *Get a single service report*

Description

Get a single service report

Usage

```
reports_get_services(id)
```

Arguments

id	integer required. The ID of this report.

Value

A list containing the following elements:

id	integer, The ID of this report.
name	string, The name of the report.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
createdAt	string, The creation time for this report.
updatedAt	string, The last updated at time for this report.
type	string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL

description	string, The user-defined description of the report.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
archived	string, The archival status of the requested item(s).
host	string, The host for the service report
displayUrl	string, The URL to display the service report.
serviceId	integer, The id of the backing service
provideAPIKey	boolean, Whether the report requests an API Key from the report viewer.
apiKey	string, A Civis API key that can be used by this report.
apiKeyId	integer, The ID of the API key. Can be used for auditing API use by this report.

`reports_get_sql` *Get a single SQL report*

Description

Get a single SQL report

Usage

```
reports_get_sql(id)
```

Arguments

id	integer required. The ID of this report.

Value

A list containing the following elements:

id	integer, The ID of this report.
name	string, The name of the report.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
createdAt	string, The creation time for this report.
updatedAt	string, The last updated at time for this report.
type	string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL
description	string, The user-defined description of the report.

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`archived`
: string, The archival status of the requested item(s).

`config`
: string, The configuration of the report visualization.

`query`
: list, A list containing the following elements:

- id integer, The query ID.
- database integer, The database ID.
- sql string, The SQL to execute.
- credential integer, The credential ID.
- resultRows array, A preview of rows returned by the query.
- resultColumns array, A preview of columns returned by the query.
- error string, The error message for this run, if present.
- startedAt string, The start time of the last run.
- finishedAt string, The end time of the last run.
- state string, The state of the last run. One of queued, running, succeeded, failed, and cancelled.
- runningAs list . A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

`reports_list` *List Reports*

Description

List Reports

Usage

```
reports_list(
  type = NULL,
  template_id = NULL,
  author = NULL,
  hidden = NULL,
  archived = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

type	string optional. If specified, return report of these types. It accepts a comma-separated list, possible values are 'tableau' or 'other'.
template_id	integer optional. If specified, return reports using the provided Template.
author	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
archived	string optional. The archival status of the requested item(s).
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of this report.
name	string, The name of the report.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
createdAt	string, The creation time for this report.
updatedAt	string, The last updated at time for this report.
type	string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL
archived	string, The archival status of the requested item(s).
projects	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
state	string, The status of the report's last run.
finishedAt	string, The time that the report's last run finished.
vizUpdatedAt	string, The time that the report's visualization was last updated.

`script`	list, A list containing the following elements: • id integer, The ID for the script. • name string, The name of the script. • sql string, The raw SQL query for the script, if applicable.
`jobPath`	string, The link to details of the job that backs this report.
`tableauId`	integer,
`templateId`	integer, The ID of the template used for this report.
`authThumbnailUrl`	string, URL for a thumbnail of the report.
`lastRun`	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.

`reports_list_dependencies`

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
reports_list_dependencies(id, user_id = NULL)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer optional. ID of target user

Value

An array containing the following fields:

`objectType`	string, Dependent object type
`fcoType`	string, Human readable dependent object type
`id`	integer, Dependent object ID
`name`	string, Dependent object name, or nil if the requesting user cannot read this object

permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

`reports_list_git` *Get the git metadata attached to an item*

Description

Get the git metadata attached to an item

Usage

```
reports_list_git(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,
gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

reports_list_git_commits *Get the git commits for an item on the current branch*

Description

Get the git commits for an item on the current branch

Usage

```
reports_list_git_commits(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

commitHash	string, The SHA of the commit.
authorName	string, The name of the commit's author.
date	string, The commit's timestamp.
message	string, The commit message.

reports_list_projects *List the projects a Report belongs to*

Description

List the projects a Report belongs to

Usage

```
reports_list_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the Report.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

---

reports_list_services_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
reports_list_services_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

`reports_list_services_projects`

List the projects a Service Report belongs to

Description

List the projects a Service Report belongs to

Usage

```
reports_list_services_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the Service Report.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.

description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

`reports_list_services_shares`
List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
reports_list_services_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`reports_list_shares` *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
reports_list_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers
: list, A list containing the following elements:

- users array,
- groups array,

writers
: list, A list containing the following elements:

- users array,
- groups array,

owners
: list, A list containing the following elements:

- users array,
- groups array,

totalUserShares
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

reports_list_sql_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
reports_list_sql_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

reports_list_sql_projects

List the projects a SQL Report belongs to

Description

List the projects a SQL Report belongs to

Usage

```
reports_list_sql_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the SQL Report.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

---

reports_list_sql_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
reports_list_sql_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

reports_patch *Update a report*

Description

Update a report

Usage

```
reports_patch(
  id,
  name = NULL,
  script_id = NULL,
  code_body = NULL,
  config = NULL,
  app_state = NULL,
  provide_api_key = NULL,
  template_id = NULL,
  use_viewers_tableau_username = NULL,
  description = NULL
)
```

Arguments

id	integer required. The ID of the report to modify.
name	string optional. The name of the report.
script_id	integer optional. The ID of the job (a script or a query) used to create this report.
code_body	string optional. The code for the report visualization.
config	string optional.
app_state	list optional. The application state blob for this report.
provide_api_key	boolean optional. Allow the report to provide an API key to front-end code.
template_id	integer optional. The ID of the template used for this report. If null is passed, no template will back this report. Changes to the backing template will reset the report appState.
use_viewers_tableau_username	boolean optional. Apply user level filtering on Tableau reports.
description	string optional. The user-defined description of the report.

Value

A list containing the following elements:

id	integer, The ID of this report.
name	string, The name of the report.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
createdAt	string, The creation time for this report.
updatedAt	string, The last updated at time for this report.
type	string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL
description	string, The user-defined description of the report.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
archived	string, The archival status of the requested item(s).
projects	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
state	string, The status of the report's last run.
finishedAt	string, The time that the report's last run finished.

`vizUpdatedAt` string, The time that the report's visualization was last updated.

`script` list, A list containing the following elements:

- id integer, The ID for the script.
- name string, The name of the script.
- sql string, The raw SQL query for the script, if applicable.

`jobPath` string, The link to details of the job that backs this report.

`tableauId` integer,

`templateId` integer, The ID of the template used for this report.

`authThumbnailUrl` string, URL for a thumbnail of the report.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`hidden` boolean, The hidden status of the item.

`authDataUrl` string, DEPRECATED: For legacy reports

`authCodeUrl` string, Link to code to render in the report.

`config` string, Any configuration metadata for this report.

`validOutputFile` boolean, Whether the job (a script or a query) that backs the report currently has a valid output file.

`provideAPIKey` boolean, Whether the report requests an API Key from the report viewer.

`apiKey` string, A Civis API key that can be used by this report.

`apiKeyId` integer, The ID of the API key. Can be used for auditing API use by this report.

`appState` list, Any application state blob for this report.

`useViewersTableauUsername` boolean, Apply user level filtering on Tableau reports.

`reports_patch_git` *Update an attached git file*

Description

Update an attached git file

Usage

```
reports_patch_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,
gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

reports_patch_services *Update some attributes of this service report*

Description

Update some attributes of this service report

Usage

```
reports_patch_services(id, name = NULL, provide_api_key = NULL)
```

Arguments

id	integer required. The ID of this report.
name	string optional. The name of the service report.
provide_api_key	boolean optional. Whether the report requests an API Key from the report viewer.

Value

A list containing the following elements:

id	integer, The ID of this report.
name	string, The name of the report.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
createdAt	string, The creation time for this report.
updatedAt	string, The last updated at time for this report.
type	string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL
description	string, The user-defined description of the report.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
archived	string, The archival status of the requested item(s).
host	string, The host for the service report
displayUrl	string, The URL to display the service report.
serviceId	integer, The id of the backing service

provideAPIKey	boolean, Whether the report requests an API Key from the report viewer.
apiKey	string, A Civis API key that can be used by this report.
apiKeyId	integer, The ID of the API key. Can be used for auditing API use by this report.

`reports_patch_sql` *Update a SQL report*

Description

Update a SQL report

Usage

```
reports_patch_sql(
  id,
  query_id = NULL,
  name = NULL,
  config = NULL,
  description = NULL
)
```

Arguments

id	integer required. The ID of this report.
query_id	integer optional. The ID of the query used to create this report.
name	string optional. The name of the report.
config	string optional. The configuration of the report visualization.
description	string optional. The user-defined description of the report.

Value

A list containing the following elements:

id	integer, The ID of this report.
name	string, The name of the report.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
createdAt	string, The creation time for this report.
updatedAt	string, The last updated at time for this report.

`type` string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL

`description` string, The user-defined description of the report.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`config` string, The configuration of the report visualization.

`query` list, A list containing the following elements:

- id integer, The query ID.
- database integer, The database ID.
- sql string, The SQL to execute.
- credential integer, The credential ID.
- resultRows array, A preview of rows returned by the query.
- resultColumns array, A preview of columns returned by the query.
- error string, The error message for this run, if present.
- startedAt string, The start time of the last run.
- finishedAt string, The end time of the last run.
- state string, The state of the last run. One of queued, running, succeeded, failed, and cancelled.
- runningAs list . A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

`reports_post` *Create an HTML report*

Description

Create an HTML report

Usage

```
reports_post(
  script_id = NULL,
  name = NULL,
  code_body = NULL,
  app_state = NULL,
  provide_api_key = NULL,
  template_id = NULL,
  description = NULL,
  hidden = NULL
)
```

Arguments

`script_id`	integer optional. The ID of the job (a script or a query) used to create this report.
`name`	string optional. The name of the report.
`code_body`	string optional. The code for the report visualization.
`app_state`	list optional. Any application state blob for this report.
`provide_api_key`	boolean optional. Allow the report to provide an API key to front-end code.
`template_id`	integer optional. The ID of the template used for this report.
`description`	string optional. The user-defined description of the report.
`hidden`	boolean optional. The hidden status of the item.

Value

A list containing the following elements:

`id`	integer, The ID of this report.
`name`	string, The name of the report.
`user`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`createdAt`	string, The creation time for this report.
`updatedAt`	string, The last updated at time for this report.
`type`	string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL
`description`	string, The user-defined description of the report.
`myPermissionLevel`	string, Your permission level on the object. One of "read", "write", or "manage".
`archived`	string, The archival status of the requested item(s).
`projects`	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
`state`	string, The status of the report's last run.
`finishedAt`	string, The time that the report's last run finished.
`vizUpdatedAt`	string, The time that the report's visualization was last updated.
`script`	list, A list containing the following elements: • id integer, The ID for the script. • name string, The name of the script. • sql string, The raw SQL query for the script, if applicable.

jobPath	string, The link to details of the job that backs this report.
tableauId	integer,
templateId	integer, The ID of the template used for this report.
authThumbnailUrl	string, URL for a thumbnail of the report.
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
hidden	boolean, The hidden status of the item.
authDataUrl	string, DEPRECATED: For legacy reports
authCodeUrl	string, Link to code to render in the report.
config	string, Any configuration metadata for this report.
validOutputFile	boolean, Whether the job (a script or a query) that backs the report currently has a valid output file.
provideAPIKey	boolean, Whether the report requests an API Key from the report viewer.
apiKey	string, A Civis API key that can be used by this report.
apiKeyId	integer, The ID of the API key. Can be used for auditing API use by this report.
appState	list, Any application state blob for this report.
useViewersTableauUsername	boolean, Apply user level filtering on Tableau reports.

`reports_post_git_commits`
Commit and push a new version of the file

Description

Commit and push a new version of the file

Usage

```
reports_post_git_commits(id, content, message, file_hash)
```

Arguments

id	integer required. The ID of the item.
content	string required. The contents to commit to the file.
message	string required. A commit message describing the changes being made.
file_hash	string required. The full SHA of the file being replaced.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

---

reports_post_grants *Grant this report the ability to perform Civis platform API operations on your behalf*

Description

Grant this report the ability to perform Civis platform API operations on your behalf

Usage

```
reports_post_grants(id)
```

Arguments

id	integer required. The ID of this report.

Value

A list containing the following elements:

id	integer, The ID of this report.
name	string, The name of the report.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
createdAt	string, The creation time for this report.
updatedAt	string, The last updated at time for this report.
type	string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL
description	string, The user-defined description of the report.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`state` string, The status of the report's last run.

`finishedAt` string, The time that the report's last run finished.

`vizUpdatedAt` string, The time that the report's visualization was last updated.

`script` list, A list containing the following elements:

- id integer, The ID for the script.
- name string, The name of the script.
- sql string, The raw SQL query for the script, if applicable.

`jobPath` string, The link to details of the job that backs this report.

`tableauId` integer,

`templateId` integer, The ID of the template used for this report.

`authThumbnailUrl` string, URL for a thumbnail of the report.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`hidden` boolean, The hidden status of the item.

`authDataUrl` string, DEPRECATED: For legacy reports

`authCodeUrl` string, Link to code to render in the report.

`config` string, Any configuration metadata for this report.

`validOutputFile` boolean, Whether the job (a script or a query) that backs the report currently has a valid output file.

`provideAPIKey` boolean, Whether the report requests an API Key from the report viewer.

`apiKey` string, A Civis API key that can be used by this report.

`apiKeyId` integer, The ID of the API key. Can be used for auditing API use by this report.

`appState` list, Any application state blob for this report.

`useViewersTableauUsername` boolean, Apply user level filtering on Tableau reports.

reports_post_refresh *Refresh the data in this Tableau report*

Description

Refresh the data in this Tableau report

Usage

```
reports_post_refresh(id)
```

Arguments

id	integer required. The ID of this report.

Value

A list containing the following elements:

id	integer, The ID of this report.
organization	list, A list containing the following elements: • id integer, The ID of this organization. • tableauRefreshUsage integer, The number of tableau refreshes used this month. • tableauRefreshLimit integer, The number of monthly tableau refreshes permitted to this organization. • tableauRefreshHistory array, The number of tableau refreshes used this month.

reports_post_services *Create a service report*

Description

Create a service report

Usage

```
reports_post_services(service_id, provide_api_key = NULL)
```

Arguments

service_id	integer required. The id of the backing service
provide_api_key	boolean optional. Whether the report requests an API Key from the report viewer.

Value

A list containing the following elements:

id	integer, The ID of this report.
name	string, The name of the report.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
createdAt	string, The creation time for this report.
updatedAt	string, The last updated at time for this report.
type	string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL
description	string, The user-defined description of the report.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
archived	string, The archival status of the requested item(s).
host	string, The host for the service report
displayUrl	string, The URL to display the service report.
serviceId	integer, The id of the backing service
provideAPIKey	boolean, Whether the report requests an API Key from the report viewer.
apiKey	string, A Civis API key that can be used by this report.
apiKeyId	integer, The ID of the API key. Can be used for auditing API use by this report.

---

reports_post_sql *Create a SQL report*

Description

Create a SQL report

Usage

```
reports_post_sql(query_id, name, config, description = NULL)
```

Arguments

query_id	integer required. The ID of the query used to create this report.
name	string required. The name of the report.
config	string required. The configuration of the report visualization.
description	string optional. The user-defined description of the report.

Value

A list containing the following elements:

`id` integer, The ID of this report.

`name` string, The name of the report.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`createdAt` string, The creation time for this report.

`updatedAt` string, The last updated at time for this report.

`type` string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL

`description` string, The user-defined description of the report.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`config` string, The configuration of the report visualization.

`query` list, A list containing the following elements:

- id integer, The query ID.
- database integer, The database ID.
- sql string, The SQL to execute.
- credential integer, The credential ID.
- resultRows array, A preview of rows returned by the query.
- resultColumns array, A preview of columns returned by the query.
- error string, The error message for this run, if present.
- startedAt string, The start time of the last run.
- finishedAt string, The end time of the last run.
- state string, The state of the last run. One of queued, running, succeeded, failed, and cancelled.
- runningAs list . A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

`reports_post_sql_refresh` *Refresh the data in a SQL report*

Description

Refresh the data in a SQL report

Usage

```
reports_post_sql_refresh(id)
```

Arguments

`id`	integer required. The ID of this report.

Value

A list containing the following elements:

`id`	integer, The ID of this report.
`name`	string, The name of the report.
`user`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`createdAt`	string, The creation time for this report.
`updatedAt`	string, The last updated at time for this report.
`type`	string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL
`description`	string, The user-defined description of the report.
`myPermissionLevel`	string, Your permission level on the object. One of "read", "write", or "manage".
`archived`	string, The archival status of the requested item(s).
`config`	string, The configuration of the report visualization.
`query`	list, A list containing the following elements: • id integer, The query ID. • database integer, The database ID. • sql string, The SQL to execute. • credential integer, The credential ID. • resultRows array, A preview of rows returned by the query.

- resultColumns array, A preview of columns returned by the query.
- error string, The error message for this run, if present.
- startedAt string, The start time of the last run.
- finishedAt string, The end time of the last run.
- state string, The state of the last run. One of queued, running, succeeded, failed, and cancelled.
- runningAs list . A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

`reports_put_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
reports_put_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id`	integer, The ID of this report.
`name`	string, The name of the report.
`user`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`createdAt`	string, The creation time for this report.
`updatedAt`	string, The last updated at time for this report.

`type` string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL

`description` string, The user-defined description of the report.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`state` string, The status of the report's last run.

`finishedAt` string, The time that the report's last run finished.

`vizUpdatedAt` string, The time that the report's visualization was last updated.

`script` list, A list containing the following elements:

- id integer, The ID for the script.
- name string, The name of the script.
- sql string, The raw SQL query for the script, if applicable.

`jobPath` string, The link to details of the job that backs this report.

`tableauId` integer,

`templateId` integer, The ID of the template used for this report.

`authThumbnailUrl` string, URL for a thumbnail of the report.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`hidden` boolean, The hidden status of the item.

`authDataUrl` string, DEPRECATED: For legacy reports

`authCodeUrl` string, Link to code to render in the report.

`config` string, Any configuration metadata for this report.

`validOutputFile` boolean, Whether the job (a script or a query) that backs the report currently has a valid output file.

`provideAPIKey` boolean, Whether the report requests an API Key from the report viewer.

`apiKey` string, A Civis API key that can be used by this report.

`apiKeyId` integer, The ID of the API key. Can be used for auditing API use by this report.

`appState` list, Any application state blob for this report.

`useViewersTableauUsername` boolean, Apply user level filtering on Tableau reports.

reports_put_git *Attach an item to a file in a git repo*

Description

Attach an item to a file in a git repo

Usage

```
reports_put_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,

gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

`reports_put_projects` *Add a Report to a project*

Description

Add a Report to a project

Usage

```
reports_put_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Report.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

`reports_put_services_archive`
Update the archive status of this object

Description

Update the archive status of this object

Usage

```
reports_put_services_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ID of this report.
name	string, The name of the report.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
createdAt	string, The creation time for this report.
updatedAt	string, The last updated at time for this report.
type	string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL
description	string, The user-defined description of the report.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
archived	string, The archival status of the requested item(s).
host	string, The host for the service report
displayUrl	string, The URL to display the service report.
serviceId	integer, The id of the backing service
provideAPIKey	boolean, Whether the report requests an API Key from the report viewer.
apiKey	string, A Civis API key that can be used by this report.
apiKeyId	integer, The ID of the API key. Can be used for auditing API use by this report.

---

reports_put_services_projects

Add a Service Report to a project

Description

Add a Service Report to a project

Usage

```
reports_put_services_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Service Report.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

`reports_put_services_shares_groups`
Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
reports_put_services_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`reports_put_services_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
reports_put_services_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_ids`	array required. An array of one or more user IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`
: list, A list containing the following elements:

- users array,
- groups array,

`writers`
: list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares`
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`reports_put_services_transfer`
Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
reports_put_services_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies` array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`reports_put_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
reports_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers` list, A list containing the following elements:
- users array,
- groups array,

`writers` list, A list containing the following elements:
- users array,
- groups array,

`owners` list, A list containing the following elements:
- users array,
- groups array,

`totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`reports_put_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
reports_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id` integer required. The ID of the resource that is shared.

`user_ids` array required. An array of one or more user IDs.

`permission_level` string required. Options are: "read", "write", or "manage".

`share_email_body` string optional. Custom body text for e-mail sent on a share.

`send_shared_email` boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- `readers` list, A list containing the following elements:
  - users array,
  - groups array,
- `writers` list, A list containing the following elements:
  - users array,
  - groups array,
- `owners` list, A list containing the following elements:
  - users array,
  - groups array,
- `totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- `totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`reports_put_sql_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
reports_put_sql_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

- `id` integer, The ID of this report.
- `name` string, The name of the report.
- `user` list, A list containing the following elements:
  - id integer, The ID of this user.

- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`createdAt` string, The creation time for this report.

`updatedAt` string, The last updated at time for this report.

`type` string, The type of the report. One of: ReportTypes::HTML, ReportTypes::Tableau, ReportTypes::ShinyApp, ReportTypes::SQL

`description` string, The user-defined description of the report.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`config` string, The configuration of the report visualization.

`query` list, A list containing the following elements:

- id integer, The query ID.
- database integer, The database ID.
- sql string, The SQL to execute.
- credential integer, The credential ID.
- resultRows array, A preview of rows returned by the query.
- resultColumns array, A preview of columns returned by the query.
- error string, The error message for this run, if present.
- startedAt string, The start time of the last run.
- finishedAt string, The end time of the last run.
- state string, The state of the last run. One of queued, running, succeeded, failed, and cancelled.
- runningAs list . A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

`reports_put_sql_projects` *Add a SQL Report to a project*

Description

Add a SQL Report to a project

Usage

```
reports_put_sql_projects(id, project_id)
```

Arguments

id	integer required. The ID of the SQL Report.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

---

reports_put_sql_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
reports_put_sql_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- `readers` list, A list containing the following elements:
  - users array,
  - groups array,
- `writers` list, A list containing the following elements:
  - users array,
  - groups array,
- `owners` list, A list containing the following elements:
  - users array,
  - groups array,
- `totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- `totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`reports_put_sql_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
reports_put_sql_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

- `id` integer required. The ID of the resource that is shared.
- `user_ids` array required. An array of one or more user IDs.
- `permission_level` string required. Options are: "read", "write", or "manage".
- `share_email_body` string optional. Custom body text for e-mail sent on a share.
- `send_shared_email` boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- `readers` list, A list containing the following elements:
  - users array,
  - groups array,
- `writers` list, A list containing the following elements:
  - users array,
  - groups array,
- `owners` list, A list containing the following elements:
  - users array,
  - groups array,
- `totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- `totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`reports_put_sql_transfer` *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
reports_put_sql_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

- `id` integer required. The ID of the resource that is shared.
- `user_id` integer required. ID of target user
- `include_dependencies` boolean required. Whether or not to give manage permissions on all dependencies
- `email_body` string optional. Custom body text for e-mail sent on transfer.
- `send_email` boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

reports_put_transfer *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
reports_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. ID of target user
include_dependencies	boolean required. Whether or not to give manage permissions on all dependencies
email_body	string optional. Custom body text for e-mail sent on transfer.
send_email	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies` array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`resolved` *Check whether a CivisFuture has resolved.*

Description

Check whether a CivisFuture has resolved.

Usage

```
resolved(future, ...)
```

Arguments

`future`	CivisFuture object.
`...`	unused for CivisFuture.

`roles_list` *List Roles*

Description

List Roles

Usage

```
roles_list(limit = NULL, page_num = NULL, order = NULL, order_dir = NULL)
```

Arguments

limit	integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id	integer, ID of the Role.
name	string, The name of the Role.
slug	string, The slug.
description	string, The description of the Role.

---

run_civis *Evaluate an R expression in a Civis Platform container*

Description

Evaluate an R expression in a Civis Platform container

Usage

```
run_civis(expr, ...)
```

Arguments

expr	code to evaluate
...	arguments to CivisFuture

Details

run_civis blocks until completion. For non-blocking calls, use futures directly with civis_platform. Attempts are made at detecting and installing necessary packages within the container, and detecting global variables required in expr.

See Also

Other script_utils: civis_script(), fetch_output_file_ids(), run_template()

Examples

```
## Not run:
run_civis(2+2)

# specify required resources, and a specific image
run_civis(2+2,
  required_resources = list(cpu = 1024, memory = 2048),
  docker_image_name='image',
  docker_image_tag = 'latest')


## End(Not run)
```

`run_template` *Run a template script*

Description

Run a template script

Usage

```
run_template(id, arguments, JSONValue = FALSE, ...)
```

Arguments

`id`	id of the template script.
`arguments`	list of arguments to the script.
`JSONValue`	bool (default FALSE) If true, returns the JSON values instead of the file_ids
`...`	additional arguments to `scripts_post_custom`

Value

If JSONValue is FALSE, File ids of any run outputs are returned. If JSONValue is TRUE, JSON values of first JSON run output is returned. If there are no JSON outputs, warning message is printed and nothing is returned If there are more than 1 JSON outputs, warning message is printed and the first JSON output is returned.

See Also

Other script_utils: `civis_script()`, `fetch_output_file_ids()`, `run_civis()`

Examples

```
## Not run:
# Try a search for the template id
search_list('template name', type = 'template_script')

# Run the template
run_template(id, arguments = list(arg1 = 1, arg2 = 2), ...)

# Run the template and return JSON value outputs
run_template(id, arguments = list(arg1 = 1, arg2 = 2), JSONValue=TRUE, ...)

## End(Not run)
```

---

scripts_delete *Archive a script (deprecated, use archive endpoints instead)*

Description

Archive a script (deprecated, use archive endpoints instead)

Usage

```
scripts_delete(id)
```

Arguments

id	integer required. The ID for the script.

Value

An empty HTTP response

---

scripts_delete_containers *Archive a container (deprecated, use archive endpoints)*

Description

Archive a container (deprecated, use archive endpoints)

Usage

```
scripts_delete_containers(id)
```

Arguments

id	integer required. The ID for the script.

Value

An empty HTTP response

`scripts_delete_containers_projects`
Remove a Container Script from a project

Description

Remove a Container Script from a project

Usage

```
scripts_delete_containers_projects(id, project_id)
```

Arguments

`id`	integer required. The ID of the Container Script.
`project_id`	integer required. The ID of the project.

Value

An empty HTTP response

`scripts_delete_containers_runs`
Cancel a run

Description

Cancel a run

Usage

```
scripts_delete_containers_runs(id, run_id)
```

Arguments

`id`	integer required. The ID of the Container job.
`run_id`	integer required. The ID of the run.

Value

An empty HTTP response

scripts_delete_containers_shares_groups
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
scripts_delete_containers_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

scripts_delete_containers_shares_users
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
scripts_delete_containers_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

scripts_delete_custom *Archive a Custom Script (deprecated, use archiving endpoints instead)*

Description

Archive a Custom Script (deprecated, use archiving endpoints instead)

Usage

```
scripts_delete_custom(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

scripts_delete_custom_projects *Remove a Custom Script from a project*

Description

Remove a Custom Script from a project

Usage

```
scripts_delete_custom_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Custom Script.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

scripts_delete_custom_runs *Cancel a run*

Description

Cancel a run

Usage

```
scripts_delete_custom_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Custom job.
run_id	integer required. The ID of the run.

Value

An empty HTTP response

scripts_delete_custom_shares_groups *Revoke the permissions a group has on this object*

Description

Revoke the permissions a group has on this object

Usage

```
scripts_delete_custom_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

scripts_delete_custom_shares_users *Revoke the permissions a user has on this object*

Description

Revoke the permissions a user has on this object

Usage

```
scripts_delete_custom_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

scripts_delete_dbt *Archive a dbt Script (deprecated, use archiving endpoints instead)*

Description

Archive a dbt Script (deprecated, use archiving endpoints instead)

Usage

```
scripts_delete_dbt(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

scripts_delete_dbt_projects *Remove a dbt Script from a project*

Description

Remove a dbt Script from a project

Usage

```
scripts_delete_dbt_projects(id, project_id)
```

Arguments

id	integer required. The ID of the dbt Script.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

scripts_delete_dbt_runs *Cancel a run*

Description

Cancel a run

Usage

```
scripts_delete_dbt_runs(id, run_id)
```

Arguments

id	integer required. The ID of the dbt job.
run_id	integer required. The ID of the run.

Value

An empty HTTP response

scripts_delete_dbt_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
scripts_delete_dbt_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

scripts_delete_dbt_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
scripts_delete_dbt_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

scripts_delete_javascript

Archive a JavaScript Script (deprecated, use archiving endpoints instead)

Description

Archive a JavaScript Script (deprecated, use archiving endpoints instead)

Usage

```
scripts_delete_javascript(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

scripts_delete_javascript_projects

Remove a JavaScript Script from a project

Description

Remove a JavaScript Script from a project

Usage

```
scripts_delete_javascript_projects(id, project_id)
```

Arguments

id	integer required. The ID of the JavaScript Script.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

`scripts_delete_javascript_runs`
Cancel a run

Description

Cancel a run

Usage

```
scripts_delete_javascript_runs(id, run_id)
```

Arguments

`id`	integer required. The ID of the Javascript job.
`run_id`	integer required. The ID of the run.

Value

An empty HTTP response

`scripts_delete_javascript_shares_groups`
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
scripts_delete_javascript_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

scripts_delete_javascript_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
scripts_delete_javascript_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

scripts_delete_python3

Archive a Python Script (deprecated, use archiving endpoints instead)

Description

Archive a Python Script (deprecated, use archiving endpoints instead)

Usage

```
scripts_delete_python3(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

scripts_delete_python3_projects
Remove a Python Script from a project

Description

Remove a Python Script from a project

Usage

```
scripts_delete_python3_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Python Script.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

scripts_delete_python3_runs
Cancel a run

Description

Cancel a run

Usage

```
scripts_delete_python3_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Python job.
run_id	integer required. The ID of the run.

Value

An empty HTTP response

scripts_delete_python3_shares_groups
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
scripts_delete_python3_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

scripts_delete_python3_shares_users
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
scripts_delete_python3_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

scripts_delete_r *Archive an R Script (deprecated, use archiving endpoints instead)*

Description

Archive an R Script (deprecated, use archiving endpoints instead)

Usage

```
scripts_delete_r(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

scripts_delete_r_projects *Remove an R Script from a project*

Description

Remove an R Script from a project

Usage

```
scripts_delete_r_projects(id, project_id)
```

Arguments

id	integer required. The ID of the R Script.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

scripts_delete_r_runs *Cancel a run*

Description

Cancel a run

Usage

```
scripts_delete_r_runs(id, run_id)
```

Arguments

id	integer required. The ID of the R job.
run_id	integer required. The ID of the run.

Value

An empty HTTP response

scripts_delete_r_shares_groups *Revoke the permissions a group has on this object*

Description

Revoke the permissions a group has on this object

Usage

```
scripts_delete_r_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

scripts_delete_r_shares_users *Revoke the permissions a user has on this object*

Description

Revoke the permissions a user has on this object

Usage

```
scripts_delete_r_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

scripts_delete_sql *Archive a SQL Script (deprecated, use archiving endpoints instead)*

Description

Archive a SQL Script (deprecated, use archiving endpoints instead)

Usage

```
scripts_delete_sql(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

scripts_delete_sql_projects

Remove a SQL Script from a project

Description

Remove a SQL Script from a project

Usage

```
scripts_delete_sql_projects(id, project_id)
```

Arguments

id	integer required. The ID of the SQL Script.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

scripts_delete_sql_runs

Cancel a run

Description

Cancel a run

Usage

```
scripts_delete_sql_runs(id, run_id)
```

Arguments

id	integer required. The ID of the SQL job.
run_id	integer required. The ID of the run.

Value

An empty HTTP response

scripts_delete_sql_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
scripts_delete_sql_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

scripts_delete_sql_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
scripts_delete_sql_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

`scripts_get` *Get details about a SQL script (legacy)*

Description

Get details about a SQL script (legacy)

Usage

```
scripts_get(id)
```

Arguments

`id`	integer required. The ID for the script.

Value

A list containing the following elements:

`id`	integer, The ID for the script.
`name`	string, The name of the script.
`type`	string, The type of script.
`createdAt`	string, The time this script was created.
`updatedAt`	string, The time this script was last updated.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`state`	string, The status of the script's last run.
`finishedAt`	string, The time that the script's last run finished.
`category`	string, The category of the script.
`projects`	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
`parentId`	integer, The ID of the parent job that will trigger this script
`userContext`	string, "runner" or "author", who to execute the script as when run as a template.
`params`	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value.

- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.

- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`sql` string, The raw SQL query for the script.

`expandedArguments` list, Expanded arguments for use in injecting into different environments.

`templateScriptId` integer, The ID of the template script, if any.

`scripts_get_containers` *View a container*

Description

View a container

Usage

```
scripts_get_containers(id)
```

Arguments

`id`	integer required. The ID for the script.

Value

A list containing the following elements:

`id`	integer, The ID for the script.
`fromTemplateAliases`	array, An array containing the following fields: • id integer, The id of the Alias object. • objectId integer, The id of the object • alias string, The alias of the object
`name`	string, The name of the container.
`type`	string, The type of the script (e.g Container)
`createdAt`	string, The time this script was created.
`updatedAt`	string, The time the script was last updated.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`state`	string, The status of the script's last run.
`finishedAt`	string, The time that the script's last run finished.
`category`	string, The category of the script.
`projects`	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template script.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`requiredResources`
list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`repoHttpUri` string, The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

`repoRef` string, The tag or branch of the github repo to clone into the container.

`remoteHostCredentialId`
integer, The id of the database credentials to pass into the environment of the container.

`gitCredentialId`
integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

dockerCommand	string, The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image's ENTRYPOINT/CMD.
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub.
instanceType	string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
cancelTimeout	integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
timeZone	string, The time zone of this script.
partitionLabel	string, The partition label used to run this object.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
hidden	boolean, The hidden status of the item.
archived	string, The archival status of the requested item(s).
targetProjectId	integer, Target project to which script outputs will be added.
runningAsId	integer, The ID of the runner of this script.

`scripts_get_containers_runs` *Check status of a run*

Description

Check status of a run

Usage

```
scripts_get_containers_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Container job.
run_id	integer required. The ID of the run.

Value

A list containing the following elements:

id	integer, The ID of the run.
containerId	integer, The ID of the Container job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
inputs	list, The inputs of the script at the time of the run, including params and arguments.
maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores.

---

scripts_get_custom *Get a Custom Script*

Description

Get a Custom Script

Usage

```
scripts_get_custom(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for the script.
fromTemplateAliases	array, An array containing the following fields: • id integer, The id of the Alias object. • objectId integer, The id of the object

- alias string, The alias of the object

`name` string, The name of the script.

`type` string, The type of the script (e.g Custom)

`backingScriptType`
string, The type of the script backing this template (e.g Python)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string,

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
: integer, The ID of the template that this script is backing.

`fromTemplateId`
: integer, The ID of the template script.

`uiReportUrl`
: integer, The url of the custom HTML.

`uiReportId`
: integer, The id of the report with the custom HTML.

`uiReportProvideAPIKey`
: boolean, Whether the ui report requests an API Key from the report viewer.

`templateScriptName`
: string, The name of the template script.

`templateNote`
: string, The template's note.

`remoteHostId`
: integer, The remote host ID that this script will connect to.

`credentialId`
: integer, The credential that this script will use.

`codePreview`
: string, The code that this script will run with arguments inserted.

`schedule`
: list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications`
: list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs`
: list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`targetProjectId` integer, Target project to which script outputs will be added.

`lastSuccessfulRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
- memory integer, The amount of RAM to allocate for the container (in MB).
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`partitionLabel` string, The partition label used to run this object. Only applicable for jobs using Docker.

`runningAsId` integer, The ID of the runner of this script.

scripts_get_custom_runs *Check status of a run*

Description

Check status of a run

Usage

```
scripts_get_custom_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Custom job.
run_id	integer required. The ID of the run.

Value

A list containing the following elements:

id	integer, The ID of the run.
customId	integer, The ID of the Custom job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
inputs	list, The inputs of the script at the time of the run, including params and arguments.
maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB. Only available if the backing script is a Python, R, or container script.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores. Only available if the backing script is a Python, R, or container script.

scripts_get_dbt *Get a dbt Script*

Description

Get a dbt Script

Usage

```
scripts_get_dbt(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for the script.
name	string, The name of the script.
type	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
createdAt	string, The time this script was created.
updatedAt	string, The time the script was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the script's last run.
finishedAt	string, The time that the script's last run finished.
category	string, The category of the script.
projects	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
parentId	integer, The ID of the parent job that will trigger this script
userContext	string, "runner" or "author", who to execute the script as when run as a template.
params	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value.

- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.

- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.

- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`partitionLabel` string, The partition label used to run this object.

`runningAsId` integer, The ID of the runner of this script.

`dbtProject` list, A list containing the following elements:

- target string, Which profile target to use. Ignored when used in conjunction with generate_profiles.
- schema string, The output schema for dbt to use.
- projectDir string, The path to dbt_project.yml. Defaults to the root of the repository. Generates 'DBT_PROJECT_DIR' environment variable.
- profilesDir string, The path to the profiles.yml file to be used by dbt. Ignored when used in conjunction with generate_profiles. Generates 'DBT_PROFILES_DIR' environment variable.
- dbtVersion string, The version of dbt to use. Generates 'DBT_VERSION' environment variable.
- dbtCommand string, The primary dbt command to run. Valid commands are build, run, test, compile, and retry.
- dbtCommandLineArgs string, Additional command line arguments to pass to dbt. Ignored when dbt retry command is selected.
- docsReportId string, The ID of the HTML report hosting the static dbt docs for this job. Updates every time a run succeeds. This report will be automatically shared with all users who are shared on the job.
- skipDocsGeneration boolean, Whether to skip dbt docs generation. If true, the linked docs report will not be updated when the script runs. Defaults to false.
- generateProfiles boolean, Whether to generate the profiles.yml file when running the script. Defaults to false.

`repoHttpUri` string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`repoRef` string, A git reference specifying an unambiguous version of the file. Can be a branch name, a tag, or the full or shortened SHA of a commit. Defaults to 'main'.

`targetDatabase` list, A list containing the following elements:

- remoteHostId integer, The remote host ID that this script will connect to.
- credentialId integer, The credential that this script will use.

scripts_get_dbt_runs *Check status of a run*

Description

Check status of a run

Usage

```
scripts_get_dbt_runs(id, run_id)
```

Arguments

id	integer required. The ID of the dbt job.
run_id	integer required. The ID of the run.

Value

A list containing the following elements:

id	integer, The ID of the run.
dbtId	integer, The ID of the dbt job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
inputs	list, The inputs of the script at the time of the run, including params and arguments.
maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores.

`scripts_get_javascript` *Get a JavaScript Script*

Description

Get a JavaScript Script

Usage

```
scripts_get_javascript(id)
```

Arguments

`id`	integer required.

Value

A list containing the following elements:

`id`	integer, The ID for the script.
`name`	string, The name of the script.
`type`	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
`createdAt`	string, The time this script was created.
`updatedAt`	string, The time the script was last updated.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`state`	string, The status of the script's last run.
`finishedAt`	string, The time that the script's last run finished.
`category`	string, The category of the script.
`projects`	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
`parentId`	integer, The ID of the parent job that will trigger this script
`userContext`	string, "runner" or "author", who to execute the script as when run as a template.
`params`	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value.

- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.

- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`source` string, The body/text of the script.

`remoteHostId` integer, The remote host ID that this script will connect to.

`credentialId` integer, The credential that this script will use.

`runningAsId` integer, The ID of the runner of this script.

scripts_get_javascript_git_commits
Get file contents at git ref

Description

Get file contents at git ref

Usage

```
scripts_get_javascript_git_commits(id, commit_hash)
```

Arguments

id	integer required. The ID of the item.
commit_hash	string required. The SHA (full or shortened) of the desired git commit.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

scripts_get_javascript_runs
Check status of a run

Description

Check status of a run

Usage

```
scripts_get_javascript_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Javascript job.
run_id	integer required. The ID of the run.

Value

A list containing the following elements:

id	integer, The ID of the run.
javascriptId	integer, The ID of the Javascript job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

---

scripts_get_python3 *Get a Python Script*

Description

Get a Python Script

Usage

```
scripts_get_python3(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for the script.
name	string, The name of the script.
type	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
createdAt	string, The time this script was created.
updatedAt	string, The time the script was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials.

- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.

- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

hidden	boolean, The hidden status of the item.
targetProjectId	integer, Target project to which script outputs will be added.
archived	string, The archival status of the requested item(s).
requiredResources	list, A list containing the following elements: • cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares. • memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB. • diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.
instanceType	string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
cancelTimeout	integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
dockerImageTag	string, The tag of the docker image to pull from DockerHub.
partitionLabel	string, The partition label used to run this object.
runningAsId	integer, The ID of the runner of this script.
source	string, The body/text of the script.

`scripts_get_python3_git_commits`
Get file contents at git ref

Description

Get file contents at git ref

Usage

```
scripts_get_python3_git_commits(id, commit_hash)
```

Arguments

id	integer required. The ID of the item.
commit_hash	string required. The SHA (full or shortened) of the desired git commit.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

---

scripts_get_python3_runs

Check status of a run

Description

Check status of a run

Usage

```
scripts_get_python3_runs(id, run_id)
```

Arguments

id	integer required. The ID of the Python job.
run_id	integer required. The ID of the run.

Value

A list containing the following elements:

id	integer, The ID of the run.
pythonId	integer, The ID of the Python job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
inputs	list, The inputs of the script at the time of the run, including params and arguments.
maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores.

scripts_get_r *Get an R Script*

Description

Get an R Script

Usage

```
scripts_get_r(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for the script.
name	string, The name of the script.
type	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
createdAt	string, The time this script was created.
updatedAt	string, The time the script was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the script's last run.
finishedAt	string, The time that the script's last run finished.
category	string, The category of the script.
projects	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
parentId	integer, The ID of the parent job that will trigger this script
userContext	string, "runner" or "author", who to execute the script as when run as a template.
params	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value.

- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.

- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.

- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType	string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
cancelTimeout	integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
dockerImageTag	string, The tag of the docker image to pull from DockerHub.
partitionLabel	string, The partition label used to run this object.
runningAsId	integer, The ID of the runner of this script.
source	string, The body/text of the script.

`scripts_get_r_git_commits`
Get file contents at git ref

Description

Get file contents at git ref

Usage

```
scripts_get_r_git_commits(id, commit_hash)
```

Arguments

id	integer required. The ID of the item.
commit_hash	string required. The SHA (full or shortened) of the desired git commit.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

`scripts_get_r_runs` *Check status of a run*

Description

Check status of a run

Usage

```
scripts_get_r_runs(id, run_id)
```

Arguments

`id`	integer required. The ID of the R job.
`run_id`	integer required. The ID of the run.

Value

A list containing the following elements:

`id`	integer, The ID of the run.
`rId`	integer, The ID of the R job.
`state`	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
`isCancelRequested`	boolean, True if run cancel requested, else false.
`createdAt`	string, The time the run was created.
`startedAt`	string, The time the run started at.
`finishedAt`	string, The time the run completed.
`error`	string, The error, if any, returned by the run.
`inputs`	list, The inputs of the script at the time of the run, including params and arguments.
`maxMemoryUsage`	number, If the run has finished, the maximum amount of memory used during the run, in MB.
`maxCpuUsage`	number, If the run has finished, the maximum amount of cpu used during the run, in millicores.

`scripts_get_sql` *Get a SQL Script*

Description

Get a SQL Script

Usage

```
scripts_get_sql(id)
```

Arguments

`id`	integer required.

Value

A list containing the following elements:

`id`	integer, The ID for the script.
`name`	string, The name of the script.
`type`	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
`createdAt`	string, The time this script was created.
`updatedAt`	string, The time the script was last updated.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`state`	string, The status of the script's last run.
`finishedAt`	string, The time that the script's last run finished.
`category`	string, The category of the script.
`projects`	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
`parentId`	integer, The ID of the parent job that will trigger this script
`userContext`	string, "runner" or "author", who to execute the script as when run as a template.
`params`	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value.

- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.

- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`sql` string, The raw SQL query for the script.

`expandedArguments` list, Expanded arguments for use in injecting into different environments.

`remoteHostId` integer, The remote host ID that this script will connect to.

`credentialId` integer, The credential that this script will use.

`codePreview` string, The code that this script will run with arguments inserted.

`csvSettings` list, A list containing the following elements:

- includeHeader boolean, Whether or not to include headers in the output data. Default: true
- compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
- columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
- unquoted boolean, Whether or not to quote fields. Default: false
- forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false
- filenamePrefix string, A user specified filename prefix for the output file to have. Default: null
- maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

`runningAsId` integer, The ID of the runner of this script.

`scripts_get_sql_git_commits`
Get file contents at git ref

Description

Get file contents at git ref

Usage

```
scripts_get_sql_git_commits(id, commit_hash)
```

Arguments

`id`	integer required. The ID of the item.
`commit_hash`	string required. The SHA (full or shortened) of the desired git commit.

Value

A list containing the following elements:

`content`	string, The file's contents.
`type`	string, The file's type.
`size`	integer, The file's size.
`fileHash`	string, The SHA of the file.

`scripts_get_sql_runs` *Check status of a run*

Description

Check status of a run

Usage

```
scripts_get_sql_runs(id, run_id)
```

Arguments

`id`	integer required. The ID of the SQL job.
`run_id`	integer required. The ID of the run.

Value

A list containing the following elements:

`id`	integer, The ID of the run.
`sqlId`	integer, The ID of the SQL job.
`state`	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
`isCancelRequested`	boolean, True if run cancel requested, else false.
`createdAt`	string, The time the run was created.
`startedAt`	string, The time the run started at.
`finishedAt`	string, The time the run completed.
`error`	string, The error, if any, returned by the run.
`inputs`	list, The inputs of the script at the time of the run, including params and arguments.
`output`	array, An array containing the following fields: • outputName string, The name of the output file. • fileId integer, The unique ID of the output file. • path string, The temporary link to download this output file, valid for 36 hours.
`outputCachedOn`	string, The time that the output was originally exported, if a cache entry was used by the run.

scripts_list *List Scripts*

Description

List Scripts

Usage

```
scripts_list(
  type = NULL,
  category = NULL,
  author = NULL,
  status = NULL,
  hidden = NULL,
  archived = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

type	string optional. If specified, return items of these types. The valid types are sql, python3, javascript, r, containers, and dbt.
category	string optional. A job category for filtering scripts. Must be one of script, import, export, and enhancement.
author	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.
status	string optional. If specified, returns items with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'failed', 'succeeded', 'idle', 'scheduled'.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
archived	string optional. The archival status of the requested item(s).
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at, last_run.updated_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`isTemplate` boolean, Whether others scripts use this one as a template.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`archived` string, The archival status of the requested item(s).

`templateScriptId` integer, The ID of the template script, if any.

scripts_list_containers_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
scripts_list_containers_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

scripts_list_containers_projects

List the projects a Container Script belongs to

Description

List the projects a Container Script belongs to

Usage

```
scripts_list_containers_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the Container Script.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

---

scripts_list_containers_runs

List runs for the given Container job

Description

List runs for the given Container job

Usage

```
scripts_list_containers_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the Container job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
containerId	integer, The ID of the Container job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores.

---

scripts_list_containers_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

```
scripts_list_containers_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the Container job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

---

scripts_list_containers_runs_outputs

List the outputs for a run

Description

List the outputs for a run

Usage

```
scripts_list_containers_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the container script.
run_id	integer required. The ID of the run.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.

order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

---

scripts_list_containers_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
scripts_list_containers_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,

totalUserShares
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_list_custom` *List Custom Scripts*

Description

List Custom Scripts

Usage

```
scripts_list_custom(
  from_template_id = NULL,
  author = NULL,
  status = NULL,
  hidden = NULL,
  archived = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

from_template_id
: string optional. If specified, return scripts based on the template with this ID. Specify multiple IDs as a comma-separated list.

author
: string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.

status
: string optional. If specified, returns items with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'failed', 'succeeded', 'idle', 'scheduled'.

hidden
: boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

archived
: string optional. The archival status of the requested item(s).

limit
: integer optional. Number of results to return. Defaults to its maximum of 50.

page_num
: integer optional. Page number of the results to return. Defaults to the first page, 1.

order
: string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at.

order_dir
: string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g Custom)

`backingScriptType` string, The type of the script backing this template (e.g Python)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`fromTemplateId` integer, The ID of the template script.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`archived` string, The archival status of the requested item(s).

`lastSuccessfulRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

scripts_list_custom_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
scripts_list_custom_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

scripts_list_custom_projects

List the projects a Custom Script belongs to

Description

List the projects a Custom Script belongs to

Usage

```
scripts_list_custom_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the Custom Script.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

---

scripts_list_custom_runs

List runs for the given Custom job

Description

List runs for the given Custom job

Usage

```
scripts_list_custom_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the Custom job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
customId	integer, The ID of the Custom job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB. Only available if the backing script is a Python, R, or container script.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores. Only available if the backing script is a Python, R, or container script.

---

scripts_list_custom_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

```
scripts_list_custom_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the Custom job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

---

scripts_list_custom_runs_outputs

List the outputs for a run

Description

List the outputs for a run

Usage

```
scripts_list_custom_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the custom script.
run_id	integer required. The ID of the run.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.

order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

---

scripts_list_custom_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
scripts_list_custom_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,

totalUserShares
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_list_dbt_dependencies` *List dependent objects for this object*

Description

List dependent objects for this object

Usage

```
scripts_list_dbt_dependencies(id, user_id = NULL)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer optional. ID of target user

Value

An array containing the following fields:

`objectType`	string, Dependent object type
`fcoType`	string, Human readable dependent object type
`id`	integer, Dependent object ID
`name`	string, Dependent object name, or nil if the requesting user cannot read this object
`permissionLevel`	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
`description`	string, Additional information about the dependency, if relevant
`shareable`	boolean, Whether or not the requesting user can share this object.

`scripts_list_dbt_projects` *List the projects a dbt Script belongs to*

Description

List the projects a dbt Script belongs to

Usage

```
scripts_list_dbt_projects(id, hidden = NULL)
```

Arguments

`id`	integer required. The ID of the dbt Script.
`hidden`	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

`id`	integer, The ID for this project.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The name of this project.
`description`	string, A description of the project.
`users`	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`autoShare`	boolean,
`createdAt`	string,
`updatedAt`	string,
`archived`	string, The archival status of the requested item(s).

scripts_list_dbt_runs *List runs for the given dbt job*

Description

List runs for the given dbt job

Usage

```
scripts_list_dbt_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the dbt job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
dbtId	integer, The ID of the dbt job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores.

`scripts_list_dbt_runs_logs` *Get the logs for a run*

Description

Get the logs for a run

Usage

```
scripts_list_dbt_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the dbt job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

scripts_list_dbt_runs_outputs
List the outputs for a run

Description

List the outputs for a run

Usage

```
scripts_list_dbt_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the dbt script.
run_id	integer required. The ID of the run.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

scripts_list_dbt_shares *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
scripts_list_dbt_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers list, A list containing the following elements:

- users array,
- groups array,

writers list, A list containing the following elements:

- users array,
- groups array,

owners list, A list containing the following elements:

- users array,
- groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

scripts_list_history *Get the run history and outputs of this script*

Description

Get the run history and outputs of this script

Usage

```
scripts_list_history(id)
```

Arguments

id	integer required. The ID for the script.

Value

A list containing the following elements:

id	integer, The ID of this run.
sqlId	integer, The ID of this sql.
state	string, The state of this run.
isCancelRequested	boolean, True if run cancel requested, else false.
finishedAt	string, The time that this run finished.
error	string, The error message for this run, if present.
output	array, An array containing the following fields: • outputName string, The name of the output file. • fileId integer, The unique ID of the output file. • path string, The temporary link to download this output file, valid for 36 hours.

scripts_list_javascript_dependencies *List dependent objects for this object*

Description

List dependent objects for this object

Usage

```
scripts_list_javascript_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

---

scripts_list_javascript_git

Get the git metadata attached to an item

Description

Get the git metadata attached to an item

Usage

```
scripts_list_javascript_git(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements:

- id integer, The ID for this git repository.
- repoUrl string, The URL for this git repository.
- createdAt string,
- updatedAt string,

`gitRefType` string, Specifies if the file is versioned by branch or tag.

`pullFromGit` boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

`scripts_list_javascript_git_commits`
Get the git commits for an item on the current branch

Description

Get the git commits for an item on the current branch

Usage

```
scripts_list_javascript_git_commits(id)
```

Arguments

`id`	integer required. The ID of the item.

Value

A list containing the following elements:

`commitHash`	string, The SHA of the commit.
`authorName`	string, The name of the commit's author.
`date`	string, The commit's timestamp.
`message`	string, The commit message.

`scripts_list_javascript_projects` *List the projects a JavaScript Script belongs to*

Description

List the projects a JavaScript Script belongs to

Usage

```
scripts_list_javascript_projects(id, hidden = NULL)
```

Arguments

`id`	integer required. The ID of the JavaScript Script.
`hidden`	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

`id`	integer, The ID for this project.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The name of this project.
`description`	string, A description of the project.
`users`	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`autoShare`	boolean,
`createdAt`	string,
`updatedAt`	string,
`archived`	string, The archival status of the requested item(s).

scripts_list_javascript_runs
List runs for the given Javascript job

Description

List runs for the given Javascript job

Usage

```
scripts_list_javascript_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the Javascript job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
javascriptId	integer, The ID of the Javascript job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

scripts_list_javascript_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

```
scripts_list_javascript_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the Javascript job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

scripts_list_javascript_runs_outputs

List the outputs for a run

Description

List the outputs for a run

Usage

```
scripts_list_javascript_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the javascript script.
run_id	integer required. The ID of the run.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

---

scripts_list_javascript_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
scripts_list_javascript_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

scripts_list_python3_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
scripts_list_python3_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

---

scripts_list_python3_git

Get the git metadata attached to an item

Description

Get the git metadata attached to an item

Usage

```
scripts_list_python3_git(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,
gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

scripts_list_python3_git_commits

Get the git commits for an item on the current branch

Description

Get the git commits for an item on the current branch

Usage

```
scripts_list_python3_git_commits(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

commitHash	string, The SHA of the commit.
authorName	string, The name of the commit's author.
date	string, The commit's timestamp.
message	string, The commit message.

scripts_list_python3_projects

List the projects a Python Script belongs to

Description

List the projects a Python Script belongs to

Usage

```
scripts_list_python3_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the Python Script.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

`id`	integer, The ID for this project.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The name of this project.
`description`	string, A description of the project.
`users`	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`autoShare`	boolean,
`createdAt`	string,
`updatedAt`	string,
`archived`	string, The archival status of the requested item(s).

`scripts_list_python3_runs` *List runs for the given Python job*

Description

List runs for the given Python job

Usage

```
scripts_list_python3_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the Python job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
pythonId	integer, The ID of the Python job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores.

---

scripts_list_python3_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

```
scripts_list_python3_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the Python job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

---

scripts_list_python3_runs_outputs

List the outputs for a run

Description

List the outputs for a run

Usage

```
scripts_list_python3_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the python script.
run_id	integer required. The ID of the run.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.

order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

---

scripts_list_python3_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
scripts_list_python3_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,

totalUserShares
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_list_r_dependencies` *List dependent objects for this object*

Description

List dependent objects for this object

Usage

```
scripts_list_r_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

scripts_list_r_git *Get the git metadata attached to an item*

Description

Get the git metadata attached to an item

Usage

```
scripts_list_r_git(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,
gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

scripts_list_r_git_commits *Get the git commits for an item on the current branch*

Description

Get the git commits for an item on the current branch

Usage

```
scripts_list_r_git_commits(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

commitHash	string, The SHA of the commit.
authorName	string, The name of the commit's author.
date	string, The commit's timestamp.
message	string, The commit message.

---

scripts_list_r_projects

List the projects an R Script belongs to

Description

List the projects an R Script belongs to

Usage

```
scripts_list_r_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the R Script.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

`scripts_list_r_runs` *List runs for the given R job*

Description

List runs for the given R job

Usage

```
scripts_list_r_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the R job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
rId	integer, The ID of the R job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores.

---

scripts_list_r_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

```
scripts_list_r_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

id	integer required. The ID of the R job.
run_id	integer required. The ID of the run.
last_id	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id	integer, The ID of the log.
createdAt	string, The time the log was created.
message	string, The log message.
level	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

`scripts_list_r_runs_outputs`
List the outputs for a run

Description

List the outputs for a run

Usage

```
scripts_list_r_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the r script.
run_id	integer required. The ID of the run.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

---

scripts_list_r_shares *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
scripts_list_r_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

scripts_list_sql_dependencies *List dependent objects for this object*

Description

List dependent objects for this object

Usage

```
scripts_list_sql_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

scripts_list_sql_git *Get the git metadata attached to an item*

Description

Get the git metadata attached to an item

Usage

```
scripts_list_sql_git(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,
gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

---

scripts_list_sql_git_commits

Get the git commits for an item on the current branch

Description

Get the git commits for an item on the current branch

Usage

```
scripts_list_sql_git_commits(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

commitHash	string, The SHA of the commit.
authorName	string, The name of the commit's author.
date	string, The commit's timestamp.
message	string, The commit message.

`scripts_list_sql_projects`
List the projects a SQL Script belongs to

Description

List the projects a SQL Script belongs to

Usage

```
scripts_list_sql_projects(id, hidden = NULL)
```

Arguments

`id`	integer required. The ID of the SQL Script.
`hidden`	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

`id`	integer, The ID for this project.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The name of this project.
`description`	string, A description of the project.
`users`	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`autoShare`	boolean,
`createdAt`	string,
`updatedAt`	string,
`archived`	string, The archival status of the requested item(s).

scripts_list_sql_runs *List runs for the given SQL job*

Description

List runs for the given SQL job

Usage

```
scripts_list_sql_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the SQL job.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the run.
sqlId	integer, The ID of the SQL job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
output	array, An array containing the following fields:

- outputName string, The name of the output file.
- fileId integer, The unique ID of the output file.
- path string, The temporary link to download this output file, valid for 36 hours.

`outputCachedOn` string, The time that the output was originally exported, if a cache entry was used by the run.

`scripts_list_sql_runs_logs` *Get the logs for a run*

Description

Get the logs for a run

Usage

```
scripts_list_sql_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

`id`	integer required. The ID of the SQL job.
`run_id`	integer required. The ID of the run.
`last_id`	integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted.Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
`limit`	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

`id`	integer, The ID of the log.
`createdAt`	string, The time the log was created.
`message`	string, The log message.
`level`	string, The level of the log. One of unknown,fatal,error,warn,info,debug.

scripts_list_sql_runs_outputs
List the outputs for a run

Description

List the outputs for a run

Usage

```
scripts_list_sql_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required. The ID of the sql script.
run_id	integer required. The ID of the run.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

scripts_list_sql_shares *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
scripts_list_sql_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers list, A list containing the following elements:

- users array,
- groups array,

writers list, A list containing the following elements:

- users array,
- groups array,

owners list, A list containing the following elements:

- users array,
- groups array,

totalUserShares
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

scripts_list_types *List available script types*

Description

List available script types

Usage

```
scripts_list_types()
```

Value

An array containing the following fields:

`name` string, The name of the type.

scripts_patch_containers *Update a container*

Description

Update a container

Usage

```
scripts_patch_containers(
  id,
  name = NULL,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  required_resources = NULL,
  repo_http_uri = NULL,
  repo_ref = NULL,
  remote_host_credential_id = NULL,
  git_credential_id = NULL,
  docker_command = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  instance_type = NULL,
  cancel_timeout = NULL,
```

```
  time_zone = NULL,
  partition_label = NULL,
  target_project_id = NULL,
  running_as_id = NULL
)
```

Arguments

id	integer required. The ID for the script.
name	string optional. The name of the container.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
arguments	list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion

- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`required_resources`
: list optional. A list containing the following elements:

  - cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  - memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
  - diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`repo_http_uri`
: string optional. The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

`repo_ref`
: string optional. The tag or branch of the github repo to clone into the container.

`remote_host_credential_id`
: integer optional. The id of the database credentials to pass into the environment of the container.

`git_credential_id`
: integer optional. The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

`docker_command`
: string optional. The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image's ENTRYPOINT/CMD.

`docker_image_name`
: string optional. The name of the docker image to pull from DockerHub.

`docker_image_tag`
: string optional. The tag of the docker image to pull from DockerHub.

`instance_type` string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancel_timeout` integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`time_zone` string optional. The time zone of this script.

`partition_label` string optional. The partition label used to run this object.

`target_project_id` integer optional. Target project to which script outputs will be added.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`fromTemplateAliases` array, An array containing the following fields:

- id integer, The id of the Alias object.
- objectId integer, The id of the object
- alias string, The alias of the object

`name` string, The name of the container.

`type` string, The type of the script (e.g Container)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template script.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.

- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`requiredResources`
list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`repoHttpUri` string, The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

`repoRef` string, The tag or branch of the github repo to clone into the container.

`remoteHostCredentialId`
integer, The id of the database credentials to pass into the environment of the container.

`gitCredentialId`
integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

`dockerCommand` string, The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image's ENTRYPOINT/CMD.

dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub.
instanceType	string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
cancelTimeout	integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
timeZone	string, The time zone of this script.
partitionLabel	string, The partition label used to run this object.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
hidden	boolean, The hidden status of the item.
archived	string, The archival status of the requested item(s).
targetProjectId	integer, Target project to which script outputs will be added.
runningAsId	integer, The ID of the runner of this script.

`scripts_patch_container_runs` *Update the given run*

Description

Update the given run

Usage

```
scripts_patch_container_runs(id, run_id, error = NULL)
```

Arguments

id	integer required. ID of the Job
run_id	integer required. ID of the Run
error	string optional. The error message to update

Value

An empty HTTP response

scripts_patch_custom *Update some attributes of this Custom Script*

Description

Update some attributes of this Custom Script

Usage

```
scripts_patch_custom(
  id,
  name = NULL,
  parent_id = NULL,
  arguments = NULL,
  remote_host_id = NULL,
  credential_id = NULL,
  schedule = NULL,
  notifications = NULL,
  time_zone = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  partition_label = NULL,
  running_as_id = NULL
)
```

Arguments

id	integer required. The ID for the script.
name	string optional. The name of the script.
parent_id	integer optional. The ID of the parent job that will trigger this script
arguments	list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
remote_host_id	integer optional. The remote host ID that this script will connect to.
credential_id	integer optional. The credential that this script will use.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`time_zone` string optional. The time zone of this script.

`target_project_id`
integer optional. Target project to which script outputs will be added.

`required_resources`
list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
- memory integer, The amount of RAM to allocate for the container (in MB).
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`partition_label`
string optional. The partition label used to run this object. Only applicable for jobs using Docker.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`fromTemplateAliases`
array, An array containing the following fields:

- id integer, The id of the Alias object.
- objectId integer, The id of the object
- alias string, The alias of the object

`name` string, The name of the script.

`type` string, The type of the script (e.g Custom)

`backingScriptType`
string, The type of the script backing this template (e.g Python)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string,

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template script.

`uiReportUrl` integer, The url of the custom HTML.

`uiReportId` integer, The id of the report with the custom HTML.

`uiReportProvideAPIKey`
boolean, Whether the ui report requests an API Key from the report viewer.

`templateScriptName`
string, The name of the template script.

`templateNote` string, The template's note.

`remoteHostId` integer, The remote host ID that this script will connect to.

`credentialId` integer, The credential that this script will use.

`codePreview` string, The code that this script will run with arguments inserted.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.

- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`targetProjectId` integer, Target project to which script outputs will be added.

`lastSuccessfulRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
- memory integer, The amount of RAM to allocate for the container (in MB).
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`partitionLabel` string, The partition label used to run this object. Only applicable for jobs using Docker.

`runningAsId` integer, The ID of the runner of this script.

scripts_patch_dbt *Update some attributes of this dbt Script*

Description

Update some attributes of this dbt Script

Usage

```
scripts_patch_dbt(
  id,
  name = NULL,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  instance_type = NULL,
  cancel_timeout = NULL,
  docker_image_tag = NULL,
  partition_label = NULL,
  running_as_id = NULL,
  dbt_project = NULL,
  repo_http_uri = NULL,
  repo_ref = NULL,
  target_database = NULL
)
```

Arguments

id	integer required. The ID for the script.
name	string optional. The name of the script.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user.

- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`schedule` list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`target_project_id`
integer optional. Target project to which script outputs will be added.

`required_resources`
list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instance_type` string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancel_timeout` integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`docker_image_tag`
string optional. The tag of the docker image to pull from DockerHub.

`partition_label`
string optional. The partition label used to run this object.

`running_as_id` integer optional. The ID of the runner of this script.

`dbt_project` list optional. A list containing the following elements:

- target string, Which profile target to use. Ignored when used in conjunction with generate_profiles.
- schema string, The output schema for dbt to use.
- projectDir string, The path to dbt_project.yml. Defaults to the root of the repository. Generates 'DBT_PROJECT_DIR' environment variable.
- profilesDir string, The path to the profiles.yml file to be used by dbt. Ignored when used in conjunction with generate_profiles. Generates 'DBT_PROFILES_DIR' environment variable.
- dbtVersion string, The version of dbt to use. Generates 'DBT_VERSION' environment variable.
- dbtCommand string, The primary dbt command to run. Valid commands are build, run, test, compile, and retry.
- dbtCommandLineArgs string, Additional command line arguments to pass to dbt. Ignored when dbt retry command is selected.
- docsReportId string, The ID of the HTML report hosting the static dbt docs for this job. Updates every time a run succeeds. This report will be automatically shared with all users who are shared on the job.
- skipDocsGeneration boolean, Whether to skip dbt docs generation. If true, the linked docs report will not be updated when the script runs. Defaults to false.

- generateProfiles boolean, Whether to generate the profiles.yml file when running the script. Defaults to false.

`repo_http_uri` string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`repo_ref` string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, a tag, or the full or shortened SHA of a commit. Defaults to 'main'.

`target_database`
list optional. A list containing the following elements:

- remoteHostId integer, The remote host ID that this script will connect to.
- credentialId integer, The credential that this script will use.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom

- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.

- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`partitionLabel` string, The partition label used to run this object.

`runningAsId` integer, The ID of the runner of this script.

`dbtProject` list, A list containing the following elements:

- target string, Which profile target to use. Ignored when used in conjunction with generate_profiles.
- schema string, The output schema for dbt to use.
- projectDir string, The path to dbt_project.yml. Defaults to the root of the repository. Generates 'DBT_PROJECT_DIR' environment variable.
- profilesDir string, The path to the profiles.yml file to be used by dbt. Ignored when used in conjunction with generate_profiles. Generates 'DBT_PROFILES_DIR' environment variable.
- dbtVersion string, The version of dbt to use. Generates 'DBT_VERSION' environment variable.
- dbtCommand string, The primary dbt command to run. Valid commands are build, run, test, compile, and retry.
- dbtCommandLineArgs string, Additional command line arguments to pass to dbt. Ignored when dbt retry command is selected.
- docsReportId string, The ID of the HTML report hosting the static dbt docs for this job. Updates every time a run succeeds. This report will be automatically shared with all users who are shared on the job.
- skipDocsGeneration boolean, Whether to skip dbt docs generation. If true, the linked docs report will not be updated when the script runs. Defaults to false.
- generateProfiles boolean, Whether to generate the profiles.yml file when running the script. Defaults to false.

`repoHttpUri` string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`repoRef` string, A git reference specifying an unambiguous version of the file. Can be a branch name, a tag, or the full or shortened SHA of a commit. Defaults to 'main'.

`targetDatabase` list, A list containing the following elements:

- remoteHostId integer, The remote host ID that this script will connect to.
- credentialId integer, The credential that this script will use.

`scripts_patch_dbt_runs`

Update the given run

Description

Update the given run

Usage

```
scripts_patch_dbt_runs(id, run_id, error = NULL)
```

Arguments

id	integer required. ID of the Job
run_id	integer required. ID of the Run
error	string optional. The error message to update

Value

An empty HTTP response

---

scripts_patch_javascript

Update some attributes of this JavaScript Script

Description

Update some attributes of this JavaScript Script

Usage

```
scripts_patch_javascript(
  id,
  name = NULL,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  target_project_id = NULL,
  source = NULL,
  remote_host_id = NULL,
  credential_id = NULL,
  running_as_id = NULL
)
```

Arguments

id	integer required. The ID for the script.
name	string optional. The name of the script.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
arguments	list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.

- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`target_project_id` integer optional. Target project to which script outputs will be added.

`source` string optional. The body/text of the script.

`remote_host_id` integer optional. The remote host ID that this script will connect to.

`credential_id` integer optional. The credential that this script will use.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`source` string, The body/text of the script.

`remoteHostId` integer, The remote host ID that this script will connect to.

`credentialId` integer, The credential that this script will use.

`runningAsId` integer, The ID of the runner of this script.

scripts_patch_javascript_git *Update an attached git file*

Description

Update an attached git file

Usage

```
scripts_patch_javascript_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string,

- updatedAt string,

gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

---

scripts_patch_javascript_runs *Update the given run*

Description

Update the given run

Usage

```
scripts_patch_javascript_runs(id, run_id, error = NULL)
```

Arguments

id	integer required. ID of the Job
run_id	integer required. ID of the Run
error	string optional. The error message to update

Value

An empty HTTP response

---

scripts_patch_python3 *Update some attributes of this Python Script*

Description

Update some attributes of this Python Script

Usage

```
scripts_patch_python3(
  id,
  name = NULL,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
```

```
  next_run_at = NULL,
  time_zone = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  instance_type = NULL,
  cancel_timeout = NULL,
  docker_image_tag = NULL,
  partition_label = NULL,
  running_as_id = NULL,
  source = NULL
)
```

Arguments

id	integer required. The ID for the script.
name	string optional. The name of the script.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
arguments	list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on.

- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`target_project_id` integer optional. Target project to which script outputs will be added.

`required_resources` list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instance_type` string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancel_timeout` integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`docker_image_tag` string optional. The tag of the docker image to pull from DockerHub.

`partition_label`
: string optional. The partition label used to run this object.

`running_as_id`
: integer optional. The ID of the runner of this script.

`source`
: string optional. The body/text of the script.

Value

A list containing the following elements:

`id`
: integer, The ID for the script.

`name`
: string, The name of the script.

`type`
: string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt`
: string, The time this script was created.

`updatedAt`
: string, The time the script was last updated.

`author`
: list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state`
: string, The status of the script's last run.

`finishedAt`
: string, The time that the script's last run finished.

`category`
: string, The category of the script.

`projects`
: array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId`
: integer, The ID of the parent job that will trigger this script

`userContext`
: string, "runner" or "author", who to execute the script as when run as a template.

`params`
: array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.

- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.

- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`partitionLabel` string, The partition label used to run this object.

`runningAsId` integer, The ID of the runner of this script.

`source` string, The body/text of the script.

scripts_patch_python3_git *Update an attached git file*

Description

Update an attached git file

Usage

```
scripts_patch_python3_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string,

- updatedAt string,

gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

---

scripts_patch_python3_runs *Update the given run*

Description

Update the given run

Usage

```
scripts_patch_python3_runs(id, run_id, error = NULL)
```

Arguments

id	integer required. ID of the Job
run_id	integer required. ID of the Run
error	string optional. The error message to update

Value

An empty HTTP response

---

scripts_patch_r *Update some attributes of this R Script*

Description

Update some attributes of this R Script

Usage

```
scripts_patch_r(
  id,
  name = NULL,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
```

```
  next_run_at = NULL,
  time_zone = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  instance_type = NULL,
  cancel_timeout = NULL,
  docker_image_tag = NULL,
  partition_label = NULL,
  running_as_id = NULL,
  source = NULL
)
```

Arguments

id	integer required. The ID for the script.
name	string optional. The name of the script.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
arguments	list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on.

- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`target_project_id` integer optional. Target project to which script outputs will be added.

`required_resources` list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instance_type` string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancel_timeout` integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`docker_image_tag` string optional. The tag of the docker image to pull from DockerHub.

`partition_label`
: string optional. The partition label used to run this object.

`running_as_id`
: integer optional. The ID of the runner of this script.

`source`
: string optional. The body/text of the script.

Value

A list containing the following elements:

`id`
: integer, The ID for the script.

`name`
: string, The name of the script.

`type`
: string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt`
: string, The time this script was created.

`updatedAt`
: string, The time the script was last updated.

`author`
: list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state`
: string, The status of the script's last run.

`finishedAt`
: string, The time that the script's last run finished.

`category`
: string, The category of the script.

`projects`
: array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId`
: integer, The ID of the parent job that will trigger this script

`userContext`
: string, "runner" or "author", who to execute the script as when run as a template.

`params`
: array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.

- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.

- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`partitionLabel` string, The partition label used to run this object.

`runningAsId` integer, The ID of the runner of this script.

`source` string, The body/text of the script.

scripts_patch_r_git *Update an attached git file*

Description

Update an attached git file

Usage

```
scripts_patch_r_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,

gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

---

scripts_patch_r_runs *Update the given run*

Description

Update the given run

Usage

```
scripts_patch_r_runs(id, run_id, error = NULL)
```

Arguments

id	integer required. ID of the Job
run_id	integer required. ID of the Run
error	string optional. The error message to update

Value

An empty HTTP response

---

scripts_patch_sql *Update some attributes of this SQL Script*

Description

Update some attributes of this SQL Script

Usage

```
scripts_patch_sql(
  id,
  name = NULL,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
```

```
  target_project_id = NULL,
  sql = NULL,
  remote_host_id = NULL,
  credential_id = NULL,
  csv_settings = NULL,
  running_as_id = NULL
)
```

Arguments

id	integer required. The ID for the script.
name	string optional. The name of the script.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
arguments	list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`target_project_id` integer optional. Target project to which script outputs will be added.

`sql` string optional. The raw SQL query for the script.

`remote_host_id` integer optional. The remote host ID that this script will connect to.

`credential_id` integer optional. The credential that this script will use.

`csv_settings` list optional. A list containing the following elements:

- includeHeader boolean, Whether or not to include headers in the output data. Default: true
- compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
- columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
- unquoted boolean, Whether or not to quote fields. Default: false
- forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false
- filenamePrefix string, A user specified filename prefix for the output file to have. Default: null
- maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id`	integer, The ID for the script.
`name`	string, The name of the script.
`type`	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
`createdAt`	string, The time this script was created.
`updatedAt`	string, The time the script was last updated.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`state`	string, The status of the script's last run.
`finishedAt`	string, The time that the script's last run finished.
`category`	string, The category of the script.
`projects`	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
`parentId`	integer, The ID of the parent job that will trigger this script
`userContext`	string, "runner" or "author", who to execute the script as when run as a template.
`params`	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
`arguments`	list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.

- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`sql` string, The raw SQL query for the script.

`expandedArguments` list, Expanded arguments for use in injecting into different environments.

`remoteHostId` integer, The remote host ID that this script will connect to.

`credentialId` integer, The credential that this script will use.

`codePreview` string, The code that this script will run with arguments inserted.

`csvSettings` list, A list containing the following elements:

- includeHeader boolean, Whether or not to include headers in the output data. Default: true
- compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
- columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
- unquoted boolean, Whether or not to quote fields. Default: false
- forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false
- filenamePrefix string, A user specified filename prefix for the output file to have. Default: null
- maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

`runningAsId` integer, The ID of the runner of this script.

`scripts_patch_sql_git` *Update an attached git file*

Description

Update an attached git file

Usage

```
scripts_patch_sql_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

`id`	integer required. The ID of the item.
`git_ref`	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
`git_branch`	string optional. The git branch that the file is on.
`git_path`	string optional. The path of the file in the repository.
`git_repo_url`	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
`git_ref_type`	string optional. Specifies if the file is versioned by branch or tag.
`pull_from_git`	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

`gitRef`	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
`gitBranch`	string, The git branch that the file is on.
`gitPath`	string, The path of the file in the repository.
`gitRepo`	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,

gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

`scripts_patch_sql_runs` *Update the given run*

Description

Update the given run

Usage

```
scripts_patch_sql_runs(id, run_id, error = NULL)
```

Arguments

id	integer required. ID of the Job
run_id	integer required. ID of the Run
error	string optional. The error message to update

Value

An empty HTTP response

`scripts_post` *Create a script (legacy)*

Description

Create a script (legacy)

Usage

```
scripts_post(
  name,
  remote_host_id,
  credential_id,
  sql,
  params = NULL,
  arguments = NULL,
  template_script_id = NULL,
  notifications = NULL,
  hidden = NULL
)
```

Arguments

name	string required. The name of the script.
remote_host_id	integer required. The database ID.
credential_id	integer required. The credential ID.
sql	string required. The raw SQL query for the script.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
arguments	list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
template_script_id	integer optional. The ID of the template script, if any. A script cannot both have a template script and be a template for other scripts.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully. • successEmailFromName string, Name from which success emails are sent; defaults to "Civis." • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job. • failureEmailAddresses array, Addresses to notify by e-mail when the job fails. • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.

- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`hidden` boolean optional. The hidden status of the item.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.

- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.

- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`templateScriptId` integer, The ID of the template script, if any.

`scripts_post_cancel` *Cancel a run*

Description

Cancel a run

Usage

```
scripts_post_cancel(id)
```

Arguments

`id` integer required. The ID of the job.

Value

A list containing the following elements:

`id`	integer, The ID of the run.
`state`	string, The state of the run, one of 'queued', 'running' or 'cancelled'.
`isCancelRequested`	boolean, True if run cancel requested, else false.

`scripts_post_containers` *Create a container*

Description

Create a container

Usage

```
scripts_post_containers(
  required_resources,
  name = NULL,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  repo_http_uri = NULL,
  repo_ref = NULL,
  remote_host_credential_id = NULL,
  git_credential_id = NULL,
  docker_command = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  instance_type = NULL,
  cancel_timeout = NULL,
  time_zone = NULL,
  partition_label = NULL,
  hidden = NULL,
  target_project_id = NULL,
  running_as_id = NULL
)
```

Arguments

`required_resources`
: list required. A list containing the following elements:
  - cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  - memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
  - diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`name`
: string optional. The name of the container.

`parent_id`
: integer optional. The ID of the parent job that will trigger this script

`user_context`
: string optional. "runner" or "author", who to execute the script as when run as a template.

`params`
: array optional. An array containing the following fields:
  - name string, The variable's name as used within your code.
  - label string, The label to present to users when asking them for the value.
  - description string, A short sentence or fragment describing this parameter to the end user.
  - type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - required boolean, Whether this param is required.
  - value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
  - allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments`
: list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`schedule`
: list optional. A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
  - scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`repo_http_uri` string optional. The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

`repo_ref` string optional. The tag or branch of the github repo to clone into the container.

`remote_host_credential_id` integer optional. The id of the database credentials to pass into the environment of the container.

`git_credential_id` integer optional. The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

`docker_command` string optional. The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image's ENTRYPOINT/CMD.

`docker_image_name` string optional. The name of the docker image to pull from DockerHub.

`docker_image_tag` string optional. The tag of the docker image to pull from DockerHub.

`instance_type` string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancel_timeout` integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`time_zone` string optional. The time zone of this script.

`partition_label` string optional. The partition label used to run this object.

`hidden` boolean optional. The hidden status of the item.

`target_project_id`
integer optional. Target project to which script outputs will be added.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`fromTemplateAliases`
array, An array containing the following fields:

- id integer, The id of the Alias object.
- objectId integer, The id of the object
- alias string, The alias of the object

`name` string, The name of the container.

`type` string, The type of the script (e.g Container)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.

- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template script.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.

- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`requiredResources`
list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`repoHttpUri` string, The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

`repoRef` string, The tag or branch of the github repo to clone into the container.

`remoteHostCredentialId`
integer, The id of the database credentials to pass into the environment of the container.

`gitCredentialId`
integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

`dockerCommand` string, The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image's ENTRYPOINT/CMD.

`dockerImageName`
string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
timeZone	string, The time zone of this script.
partitionLabel	string, The partition label used to run this object.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
hidden	boolean, The hidden status of the item.
archived	string, The archival status of the requested item(s).
targetProjectId	integer, Target project to which script outputs will be added.
runningAsId	integer, The ID of the runner of this script.

`scripts_post_containers_clone` *Clone this Container Script*

Description

Clone this Container Script

Usage

```
scripts_post_containers_clone(
  id,
  clone_schedule = NULL,
  clone_triggers = NULL,
  clone_notifications = NULL
)
```

Arguments

id	integer required. The ID for the script.
clone_schedule	boolean optional. If true, also copy the schedule to the new script.
clone_triggers	boolean optional. If true, also copy the triggers to the new script.
clone_notifications	boolean optional. If true, also copy the notifications to the new script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`fromTemplateAliases`
array, An array containing the following fields:

- id integer, The id of the Alias object.
- objectId integer, The id of the object
- alias string, The alias of the object

`name` string, The name of the container.

`type` string, The type of the script (e.g Container)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.

- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template script.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.

- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`requiredResources`
list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`repoHttpUri` string, The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

`repoRef` string, The tag or branch of the github repo to clone into the container.

`remoteHostCredentialId`
integer, The id of the database credentials to pass into the environment of the container.

`gitCredentialId`
integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

`dockerCommand` string, The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image's ENTRYPOINT/CMD.

`dockerImageName`
string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.

- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`timeZone` string, The time zone of this script.

`partitionLabel` string, The partition label used to run this object.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`targetProjectId` integer, Target project to which script outputs will be added.

`runningAsId` integer, The ID of the runner of this script.

`scripts_post_containers_runs` *Start a run*

Description

Start a run

Usage

```
scripts_post_containers_runs(id)
```

Arguments

`id`	integer required. The ID of the Container job.

Value

A list containing the following elements:

`id`	integer, The ID of the run.
`containerId`	integer, The ID of the Container job.
`state`	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
`isCancelRequested`	boolean, True if run cancel requested, else false.
`createdAt`	string, The time the run was created.
`startedAt`	string, The time the run started at.
`finishedAt`	string, The time the run completed.
`error`	string, The error, if any, returned by the run.
`inputs`	list, The inputs of the script at the time of the run, including params and arguments.

maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores.

`scripts_post_containers_runs_logs`
Add log messages

Description

Add log messages

Usage

```
scripts_post_containers_runs_logs(
  id,
  run_id,
  message = NULL,
  level = NULL,
  messages = NULL,
  child_job_id = NULL
)
```

Arguments

id	integer required. The ID of the script.
run_id	integer required. The ID of the script run.
message	string optional. The log message to store.
level	string optional. The log level of this message [default: info]
messages	array optional. An array containing the following fields: • message string, The log message to store. • level string, The log level of this message [default: info] • createdAt string, The timestamp of this message in ISO 8601 format. This is what logs are ordered by, so it is recommended to use timestamps with nanosecond precision. If absent, defaults to the time that the log was received by the API.
child_job_id	integer optional. The ID of the child job the message came from.

Value

An empty HTTP response

scripts_post_containers_runs_outputs *Add an output for a run*

Description

Add an output for a run

Usage

```
scripts_post_containers_runs_outputs(id, run_id, object_type, object_id)
```

Arguments

id	integer required. The ID of the container script.
run_id	integer required. The ID of the run.
object_type	string required. The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
object_id	integer required. The ID of the output.

Value

A list containing the following elements:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

scripts_post_custom *Create a Custom Script*

Description

Create a Custom Script

Usage

```
scripts_post_custom(
  from_template_id,
  name = NULL,
  parent_id = NULL,
  arguments = NULL,
  remote_host_id = NULL,
  credential_id = NULL,
  schedule = NULL,
  notifications = NULL,
  time_zone = NULL,
  hidden = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  partition_label = NULL,
  running_as_id = NULL
)
```

Arguments

`from_template_id`
: integer required. The ID of the template script.

`name`
: string optional. The name of the script.

`parent_id`
: integer optional. The ID of the parent job that will trigger this script

`arguments`
: list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`remote_host_id`
: integer optional. The remote host ID that this script will connect to.

`credential_id`
: integer optional. The credential that this script will use.

`schedule`
: list optional. A list containing the following elements:

  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
  - scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications`
: list optional. A list containing the following elements:

  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.

- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`time_zone` string optional. The time zone of this script.

`hidden` boolean optional. The hidden status of the item.

`target_project_id`
integer optional. Target project to which script outputs will be added.

`required_resources`
list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
- memory integer, The amount of RAM to allocate for the container (in MB).
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`partition_label`
string optional. The partition label used to run this object. Only applicable for jobs using Docker.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`fromTemplateAliases`
array, An array containing the following fields:

- id integer, The id of the Alias object.
- objectId integer, The id of the object
- alias string, The alias of the object

`name` string, The name of the script.

`type` string, The type of the script (e.g Custom)

`backingScriptType`
string, The type of the script backing this template (e.g Python)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string,

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template script.

`uiReportUrl` integer, The url of the custom HTML.

`uiReportId` integer, The id of the report with the custom HTML.

`uiReportProvideAPIKey`
: boolean, Whether the ui report requests an API Key from the report viewer.

`templateScriptName`
: string, The name of the template script.

`templateNote`
: string, The template's note.

`remoteHostId`
: integer, The remote host ID that this script will connect to.

`credentialId`
: integer, The credential that this script will use.

`codePreview`
: string, The code that this script will run with arguments inserted.

`schedule`
: list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
  - scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications`
: list, A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  - successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  - successOn boolean, If success email notifications are on. Defaults to user's preferences.
  - failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs`
: list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`targetProjectId` integer, Target project to which script outputs will be added.

`lastSuccessfulRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
- memory integer, The amount of RAM to allocate for the container (in MB).
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`partitionLabel` string, The partition label used to run this object. Only applicable for jobs using Docker.

`runningAsId` integer, The ID of the runner of this script.

`scripts_post_custom_clone` *Clone this Custom Script*

Description

Clone this Custom Script

Usage

```
scripts_post_custom_clone(
  id,
  clone_schedule = NULL,
  clone_triggers = NULL,
  clone_notifications = NULL
)
```

Arguments

id	integer required. The ID for the script.
clone_schedule	boolean optional. If true, also copy the schedule to the new script.
clone_triggers	boolean optional. If true, also copy the triggers to the new script.
clone_notifications	boolean optional. If true, also copy the notifications to the new script.

Value

A list containing the following elements:

id	integer, The ID for the script.
fromTemplateAliases	array, An array containing the following fields: • id integer, The id of the Alias object. • objectId integer, The id of the object • alias string, The alias of the object
name	string, The name of the script.
type	string, The type of the script (e.g Custom)
backingScriptType	string, The type of the script backing this template (e.g Python)
createdAt	string, The time this script was created.
updatedAt	string, The time the script was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the script's last run.
finishedAt	string, The time that the script's last run finished.
category	string,
projects	array, An array containing the following fields: • id integer, The ID for the project.

- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template script.

`uiReportUrl` integer, The url of the custom HTML.

`uiReportId` integer, The id of the report with the custom HTML.

`uiReportProvideAPIKey` boolean, Whether the ui report requests an API Key from the report viewer.

`templateScriptName` string, The name of the template script.

`templateNote` string, The template's note.

`remoteHostId` integer, The remote host ID that this script will connect to.

`credentialId` integer, The credential that this script will use.

`codePreview` string, The code that this script will run with arguments inserted.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.

- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`targetProjectId` integer, Target project to which script outputs will be added.

`lastSuccessfulRun`
list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`requiredResources`
list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
- memory integer, The amount of RAM to allocate for the container (in MB).
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`partitionLabel` string, The partition label used to run this object. Only applicable for jobs using Docker.

`runningAsId` integer, The ID of the runner of this script.

`scripts_post_custom_runs` *Start a run*

Description

Start a run

Usage

```
scripts_post_custom_runs(id)
```

Arguments

`id`	integer required. The ID of the Custom job.

Value

A list containing the following elements:

`id`	integer, The ID of the run.
`customId`	integer, The ID of the Custom job.
`state`	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
inputs	list, The inputs of the script at the time of the run, including params and arguments.
maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB. Only available if the backing script is a Python, R, or container script.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores. Only available if the backing script is a Python, R, or container script.

`scripts_post_custom_runs_outputs` *Add an output for a run*

Description

Add an output for a run

Usage

```
scripts_post_custom_runs_outputs(id, run_id, object_type, object_id)
```

Arguments

id	integer required. The ID of the custom script.
run_id	integer required. The ID of the run.
object_type	string required. The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
object_id	integer required. The ID of the output.

Value

A list containing the following elements:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

scripts_post_dbt *Create a dbt Script*

Description

Create a dbt Script

Usage

```
scripts_post_dbt(
  name,
  repo_http_uri,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  hidden = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  instance_type = NULL,
  cancel_timeout = NULL,
  docker_image_tag = NULL,
  partition_label = NULL,
  running_as_id = NULL,
  dbt_project = NULL,
  repo_ref = NULL,
  target_database = NULL
)
```

Arguments

name	string required. The name of the script.
repo_http_uri	string required. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user.

- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`schedule` list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`hidden` boolean optional. The hidden status of the item.

`target_project_id`
integer optional. Target project to which script outputs will be added.

`required_resources`
list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instance_type` string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancel_timeout` integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`docker_image_tag`
string optional. The tag of the docker image to pull from DockerHub.

`partition_label`
string optional. The partition label used to run this object.

`running_as_id` integer optional. The ID of the runner of this script.

`dbt_project` list optional. A list containing the following elements:

- target string, Which profile target to use. Ignored when used in conjunction with generate_profiles.
- schema string, The output schema for dbt to use.
- projectDir string, The path to dbt_project.yml. Defaults to the root of the repository. Generates 'DBT_PROJECT_DIR' environment variable.
- profilesDir string, The path to the profiles.yml file to be used by dbt. Ignored when used in conjunction with generate_profiles. Generates 'DBT_PROFILES_DIR' environment variable.
- dbtVersion string, The version of dbt to use. Generates 'DBT_VERSION' environment variable.
- dbtCommand string, The primary dbt command to run. Valid commands are build, run, test, compile, and retry.
- dbtCommandLineArgs string, Additional command line arguments to pass to dbt. Ignored when dbt retry command is selected.
- docsReportId string, The ID of the HTML report hosting the static dbt docs for this job. Updates every time a run succeeds. This report will be automatically shared with all users who are shared on the job.

- skipDocsGeneration boolean, Whether to skip dbt docs generation. If true, the linked docs report will not be updated when the script runs. Defaults to false.
- generateProfiles boolean, Whether to generate the profiles.yml file when running the script. Defaults to false.

`repo_ref` string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, a tag, or the full or shortened SHA of a commit. Defaults to 'main'.

`target_database`
list optional. A list containing the following elements:

- remoteHostId integer, The remote host ID that this script will connect to.
- credentialId integer, The credential that this script will use.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.

- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.

- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`partitionLabel` string, The partition label used to run this object.

`runningAsId` integer, The ID of the runner of this script.

`dbtProject` list, A list containing the following elements:

- target string, Which profile target to use. Ignored when used in conjunction with generate_profiles.
- schema string, The output schema for dbt to use.
- projectDir string, The path to dbt_project.yml. Defaults to the root of the repository. Generates 'DBT_PROJECT_DIR' environment variable.
- profilesDir string, The path to the profiles.yml file to be used by dbt. Ignored when used in conjunction with generate_profiles. Generates 'DBT_PROFILES_DIR' environment variable.
- dbtVersion string, The version of dbt to use. Generates 'DBT_VERSION' environment variable.
- dbtCommand string, The primary dbt command to run. Valid commands are build, run, test, compile, and retry.
- dbtCommandLineArgs string, Additional command line arguments to pass to dbt. Ignored when dbt retry command is selected.
- docsReportId string, The ID of the HTML report hosting the static dbt docs for this job. Updates every time a run succeeds. This report will be automatically shared with all users who are shared on the job.
- skipDocsGeneration boolean, Whether to skip dbt docs generation. If true, the linked docs report will not be updated when the script runs. Defaults to false.
- generateProfiles boolean, Whether to generate the profiles.yml file when running the script. Defaults to false.

`repoHttpUri` string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`repoRef` string, A git reference specifying an unambiguous version of the file. Can be a branch name, a tag, or the full or shortened SHA of a commit. Defaults to 'main'.

`targetDatabase` list, A list containing the following elements:

- remoteHostId integer, The remote host ID that this script will connect to.
- credentialId integer, The credential that this script will use.

`scripts_post_dbt_clone` *Clone this dbt Script*

Description

Clone this dbt Script

Usage

```
scripts_post_dbt_clone(
  id,
  clone_schedule = NULL,
  clone_triggers = NULL,
  clone_notifications = NULL
)
```

Arguments

`id`	integer required. The ID for the script.
`clone_schedule`	boolean optional. If true, also copy the schedule to the new script.
`clone_triggers`	boolean optional. If true, also copy the triggers to the new script.
`clone_notifications`	boolean optional. If true, also copy the notifications to the new script.

Value

A list containing the following elements:

`id`	integer, The ID for the script.
`name`	string, The name of the script.
`type`	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
`createdAt`	string, The time this script was created.
`updatedAt`	string, The time the script was last updated.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`state`	string, The status of the script's last run.
`finishedAt`	string, The time that the script's last run finished.
`category`	string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.

- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources`
list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`partitionLabel` string, The partition label used to run this object.

`runningAsId` integer, The ID of the runner of this script.

`dbtProject` list, A list containing the following elements:

- target string, Which profile target to use. Ignored when used in conjunction with generate_profiles.
- schema string, The output schema for dbt to use.
- projectDir string, The path to dbt_project.yml. Defaults to the root of the repository. Generates 'DBT_PROJECT_DIR' environment variable.
- profilesDir string, The path to the profiles.yml file to be used by dbt. Ignored when used in conjunction with generate_profiles. Generates 'DBT_PROFILES_DIR' environment variable.
- dbtVersion string, The version of dbt to use. Generates 'DBT_VERSION' environment variable.
- dbtCommand string, The primary dbt command to run. Valid commands are build, run, test, compile, and retry.
- dbtCommandLineArgs string, Additional command line arguments to pass to dbt. Ignored when dbt retry command is selected.
- docsReportId string, The ID of the HTML report hosting the static dbt docs for this job. Updates every time a run succeeds. This report will be automatically shared with all users who are shared on the job.
- skipDocsGeneration boolean, Whether to skip dbt docs generation. If true, the linked docs report will not be updated when the script runs. Defaults to false.
- generateProfiles boolean, Whether to generate the profiles.yml file when running the script. Defaults to false.

`repoHttpUri` string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

repoRef — string, A git reference specifying an unambiguous version of the file. Can be a branch name, a tag, or the full or shortened SHA of a commit. Defaults to 'main'.

targetDatabase — list, A list containing the following elements:

- remoteHostId integer, The remote host ID that this script will connect to.
- credentialId integer, The credential that this script will use.

`scripts_post_dbt_runs` *Start a run*

Description

Start a run

Usage

```
scripts_post_dbt_runs(id)
```

Arguments

id	integer required. The ID of the dbt job.

Value

A list containing the following elements:

id	integer, The ID of the run.
dbtId	integer, The ID of the dbt job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
inputs	list, The inputs of the script at the time of the run, including params and arguments.
maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores.

scripts_post_dbt_runs_outputs
Add an output for a run

Description

Add an output for a run

Usage

```
scripts_post_dbt_runs_outputs(id, run_id, object_type, object_id)
```

Arguments

id	integer required. The ID of the dbt script.
run_id	integer required. The ID of the run.
object_type	string required. The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
object_id	integer required. The ID of the output.

Value

A list containing the following elements:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

scripts_post_javascript
Create a JavaScript Script

Description

Create a JavaScript Script

Usage

```
scripts_post_javascript(
  name,
  source,
  remote_host_id,
  credential_id,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  hidden = NULL,
  target_project_id = NULL,
  running_as_id = NULL
)
```

Arguments

name	string required. The name of the script.
source	string required. The body/text of the script.
remote_host_id	integer required. The remote host ID that this script will connect to.
credential_id	integer required. The credential that this script will use.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`schedule` list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`hidden` boolean optional. The hidden status of the item.

`target_project_id` integer optional. Target project to which script outputs will be added.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName`
: string, The name of the template script.

`links`
: list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule`
: list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications`
: list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs`
: list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt`
: string, The time of the next scheduled run.

`timeZone`
: string, The time zone of this script.

`lastRun`
: list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`hidden`
: boolean, The hidden status of the item.

`targetProjectId`
: integer, Target project to which script outputs will be added.

`archived`
: string, The archival status of the requested item(s).

`source`
: string, The body/text of the script.

`remoteHostId`
: integer, The remote host ID that this script will connect to.

`credentialId`
: integer, The credential that this script will use.

`runningAsId`
: integer, The ID of the runner of this script.

`scripts_post_javascript_clone` *Clone this JavaScript Script*

Description

Clone this JavaScript Script

Usage

```
scripts_post_javascript_clone(
  id,
  clone_schedule = NULL,
  clone_triggers = NULL,
  clone_notifications = NULL
)
```

Arguments

`id`
: integer required. The ID for the script.

`clone_schedule`
: boolean optional. If true, also copy the schedule to the new script.

`clone_triggers`
: boolean optional. If true, also copy the triggers to the new script.

`clone_notifications`
: boolean optional. If true, also copy the notifications to the new script.

Value

A list containing the following elements:

`id`	integer, The ID for the script.
`name`	string, The name of the script.
`type`	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
`createdAt`	string, The time this script was created.
`updatedAt`	string, The time the script was last updated.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`state`	string, The status of the script's last run.
`finishedAt`	string, The time that the script's last run finished.
`category`	string, The category of the script.
`projects`	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
`parentId`	integer, The ID of the parent job that will trigger this script
`userContext`	string, "runner" or "author", who to execute the script as when run as a template.
`params`	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
`arguments`	list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.

- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`source` string, The body/text of the script.

`remoteHostId` integer, The remote host ID that this script will connect to.

`credentialId` integer, The credential that this script will use.

`runningAsId` integer, The ID of the runner of this script.

`scripts_post_javascript_git_checkout` *Checkout content that the existing git_ref points to and save to the object*

Description

Checkout content that the existing git_ref points to and save to the object

Usage

```
scripts_post_javascript_git_checkout(id)
```

Arguments

`id` integer required. The ID of the item.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

---

scripts_post_javascript_git_checkout_latest

Checkout latest commit on the current branch of a script or workflow

---

Description

Checkout latest commit on the current branch of a script or workflow

Usage

```
scripts_post_javascript_git_checkout_latest(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

scripts_post_javascript_git_commits
Commit and push a new version of the file

Description

Commit and push a new version of the file

Usage

```
scripts_post_javascript_git_commits(id, content, message, file_hash)
```

Arguments

id	integer required. The ID of the item.
content	string required. The contents to commit to the file.
message	string required. A commit message describing the changes being made.
file_hash	string required. The full SHA of the file being replaced.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

scripts_post_javascript_runs
Start a run

Description

Start a run

Usage

```
scripts_post_javascript_runs(id)
```

Arguments

id	integer required. The ID of the Javascript job.

Value

A list containing the following elements:

id	integer, The ID of the run.
javascriptId	integer, The ID of the Javascript job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.

---

scripts_post_javascript_runs_outputs

Add an output for a run

Description

Add an output for a run

Usage

```
scripts_post_javascript_runs_outputs(id, run_id, object_type, object_id)
```

Arguments

id	integer required. The ID of the javascript script.
run_id	integer required. The ID of the run.
object_type	string required. The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
object_id	integer required. The ID of the output.

Value

A list containing the following elements:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

scripts_post_python3 *Create a Python Script*

Description

Create a Python Script

Usage

```
scripts_post_python3(
  name,
  source,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  hidden = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  instance_type = NULL,
  cancel_timeout = NULL,
  docker_image_tag = NULL,
  partition_label = NULL,
  running_as_id = NULL
)
```

Arguments

name	string required. The name of the script.
source	string required. The body/text of the script.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom

- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`schedule` list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`hidden` boolean optional. The hidden status of the item.

`target_project_id`
: integer optional. Target project to which script outputs will be added.

`required_resources`
: list optional. A list containing the following elements:

  - cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  - memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
  - diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instance_type`
: string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancel_timeout`
: integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`docker_image_tag`
: string optional. The tag of the docker image to pull from DockerHub.

`partition_label`
: string optional. The partition label used to run this object.

`running_as_id`
: integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id`
: integer, The ID for the script.

`name`
: string, The name of the script.

`type`
: string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt`
: string, The time this script was created.

`updatedAt`
: string, The time the script was last updated.

`author`
: list, A list containing the following elements:

  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

`state`
: string, The status of the script's last run.

`finishedAt`
: string, The time that the script's last run finished.

`category`
: string, The category of the script.

`projects`
: array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.

- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources`
: list, A list containing the following elements:
  - cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  - memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
  - diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType`
: string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout`
: integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`dockerImageTag`
: string, The tag of the docker image to pull from DockerHub.

`partitionLabel`
: string, The partition label used to run this object.

`runningAsId`
: integer, The ID of the runner of this script.

`source`
: string, The body/text of the script.

`scripts_post_python3_clone`
Clone this Python Script

Description

Clone this Python Script

Usage

```
scripts_post_python3_clone(
  id,
  clone_schedule = NULL,
  clone_triggers = NULL,
  clone_notifications = NULL
)
```

Arguments

`id`
: integer required. The ID for the script.

`clone_schedule`
: boolean optional. If true, also copy the schedule to the new script.

`clone_triggers`
: boolean optional. If true, also copy the triggers to the new script.

`clone_notifications`
: boolean optional. If true, also copy the notifications to the new script.

Value

A list containing the following elements:

id	integer, The ID for the script.
name	string, The name of the script.
type	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
createdAt	string, The time this script was created.
updatedAt	string, The time the script was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the script's last run.
finishedAt	string, The time that the script's last run finished.
category	string, The category of the script.
projects	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
parentId	integer, The ID of the parent job that will trigger this script
userContext	string, "runner" or "author", who to execute the script as when run as a template.
params	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
arguments	list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.

- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`partitionLabel` string, The partition label used to run this object.

`runningAsId` integer, The ID of the runner of this script.

`source` string, The body/text of the script.

scripts_post_python3_git_checkout

Checkout content that the existing git_ref points to and save to the object

Description

Checkout content that the existing git_ref points to and save to the object

Usage

```
scripts_post_python3_git_checkout(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

scripts_post_python3_git_checkout_latest

Checkout latest commit on the current branch of a script or workflow

Description

Checkout latest commit on the current branch of a script or workflow

Usage

```
scripts_post_python3_git_checkout_latest(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

---

scripts_post_python3_git_commits

Commit and push a new version of the file

Description

Commit and push a new version of the file

Usage

```
scripts_post_python3_git_commits(id, content, message, file_hash)
```

Arguments

id	integer required. The ID of the item.
content	string required. The contents to commit to the file.
message	string required. A commit message describing the changes being made.
file_hash	string required. The full SHA of the file being replaced.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

`scripts_post_python3_runs` *Start a run*

Description

Start a run

Usage

```
scripts_post_python3_runs(id)
```

Arguments

`id`	integer required. The ID of the Python job.

Value

A list containing the following elements:

`id`	integer, The ID of the run.
`pythonId`	integer, The ID of the Python job.
`state`	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
`isCancelRequested`	boolean, True if run cancel requested, else false.
`createdAt`	string, The time the run was created.
`startedAt`	string, The time the run started at.
`finishedAt`	string, The time the run completed.
`error`	string, The error, if any, returned by the run.
`inputs`	list, The inputs of the script at the time of the run, including params and arguments.
`maxMemoryUsage`	number, If the run has finished, the maximum amount of memory used during the run, in MB.
`maxCpuUsage`	number, If the run has finished, the maximum amount of cpu used during the run, in millicores.

scripts_post_python3_runs_outputs *Add an output for a run*

Description

Add an output for a run

Usage

```
scripts_post_python3_runs_outputs(id, run_id, object_type, object_id)
```

Arguments

id	integer required. The ID of the python script.
run_id	integer required. The ID of the run.
object_type	string required. The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
object_id	integer required. The ID of the output.

Value

A list containing the following elements:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

scripts_post_r *Create an R Script*

Description

Create an R Script

Usage

```
scripts_post_r(
  name,
  source,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  hidden = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  instance_type = NULL,
  cancel_timeout = NULL,
  docker_image_tag = NULL,
  partition_label = NULL,
  running_as_id = NULL
)
```

Arguments

name	string required. The name of the script.
source	string required. The body/text of the script.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.

- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`schedule` list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`hidden` boolean optional. The hidden status of the item.

`target_project_id`
integer optional. Target project to which script outputs will be added.

`required_resources`
list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.

- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instance_type` string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancel_timeout` integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`docker_image_tag` string optional. The tag of the docker image to pull from DockerHub.

`partition_label` string optional. The partition label used to run this object.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.

- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.

- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType	string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
cancelTimeout	integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
dockerImageTag	string, The tag of the docker image to pull from DockerHub.
partitionLabel	string, The partition label used to run this object.
runningAsId	integer, The ID of the runner of this script.
source	string, The body/text of the script.

---

scripts_post_run *Run a SQL script (legacy)*

Description

Run a SQL script (legacy)

Usage

```
scripts_post_run(id)
```

Arguments

id	integer required. The ID for the script.

Value

An empty HTTP response

---

scripts_post_r_clone *Clone this R Script*

Description

Clone this R Script

Usage

```
scripts_post_r_clone(
  id,
  clone_schedule = NULL,
  clone_triggers = NULL,
  clone_notifications = NULL
)
```

Arguments

id	integer required. The ID for the script.
clone_schedule	boolean optional. If true, also copy the schedule to the new script.
clone_triggers	boolean optional. If true, also copy the triggers to the new script.
clone_notifications	boolean optional. If true, also copy the notifications to the new script.

Value

A list containing the following elements:

id	integer, The ID for the script.
name	string, The name of the script.
type	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
createdAt	string, The time this script was created.
updatedAt	string, The time the script was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the script's last run.
finishedAt	string, The time that the script's last run finished.
category	string, The category of the script.
projects	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
parentId	integer, The ID of the parent job that will trigger this script
userContext	string, "runner" or "author", who to execute the script as when run as a template.
params	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value.

- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.

- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.

- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType	string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
cancelTimeout	integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
dockerImageTag	string, The tag of the docker image to pull from DockerHub.
partitionLabel	string, The partition label used to run this object.
runningAsId	integer, The ID of the runner of this script.
source	string, The body/text of the script.

`scripts_post_r_git_checkout` *Checkout content that the existing git_ref points to and save to the object*

Description

Checkout content that the existing git_ref points to and save to the object

Usage

```
scripts_post_r_git_checkout(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

scripts_post_r_git_checkout_latest

Checkout latest commit on the current branch of a script or workflow

Description

Checkout latest commit on the current branch of a script or workflow

Usage

```
scripts_post_r_git_checkout_latest(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

scripts_post_r_git_commits

Commit and push a new version of the file

Description

Commit and push a new version of the file

Usage

```
scripts_post_r_git_commits(id, content, message, file_hash)
```

Arguments

id	integer required. The ID of the item.
content	string required. The contents to commit to the file.
message	string required. A commit message describing the changes being made.
file_hash	string required. The full SHA of the file being replaced.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

---

scripts_post_r_runs *Start a run*

Description

Start a run

Usage

```
scripts_post_r_runs(id)
```

Arguments

id	integer required. The ID of the R job.

Value

A list containing the following elements:

id	integer, The ID of the run.
rId	integer, The ID of the R job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
inputs	list, The inputs of the script at the time of the run, including params and arguments.
maxMemoryUsage	number, If the run has finished, the maximum amount of memory used during the run, in MB.
maxCpuUsage	number, If the run has finished, the maximum amount of cpu used during the run, in millicores.

scripts_post_r_runs_outputs *Add an output for a run*

Description

Add an output for a run

Usage

```
scripts_post_r_runs_outputs(id, run_id, object_type, object_id)
```

Arguments

id	integer required. The ID of the r script.
run_id	integer required. The ID of the run.
object_type	string required. The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
object_id	integer required. The ID of the output.

Value

A list containing the following elements:

objectType	string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId	integer, The ID of the output.
name	string, The name of the output.
link	string, The hypermedia link to the output.
value	string,

scripts_post_sql *Create a SQL Script*

Description

Create a SQL Script

Usage

```
scripts_post_sql(
  name,
  sql,
  remote_host_id,
  credential_id,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  hidden = NULL,
  target_project_id = NULL,
  csv_settings = NULL,
  running_as_id = NULL
)
```

Arguments

name	string required. The name of the script.
sql	string required. The raw SQL query for the script.
remote_host_id	integer required. The remote host ID that this script will connect to.
credential_id	integer required. The credential that this script will use.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.

- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`schedule` list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`hidden` boolean optional. The hidden status of the item.

`target_project_id` integer optional. Target project to which script outputs will be added.

`csv_settings` list optional. A list containing the following elements:

- includeHeader boolean, Whether or not to include headers in the output data. Default: true
- compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip

- columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
- unquoted boolean, Whether or not to quote fields. Default: false
- forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false
- filenamePrefix string, A user specified filename prefix for the output file to have. Default: null
- maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom

- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.

- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`sql` string, The raw SQL query for the script.

`expandedArguments` list, Expanded arguments for use in injecting into different environments.

`remoteHostId` integer, The remote host ID that this script will connect to.

`credentialId` integer, The credential that this script will use.

`codePreview` string, The code that this script will run with arguments inserted.

`csvSettings` list, A list containing the following elements:

- includeHeader boolean, Whether or not to include headers in the output data. Default: true
- compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
- columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma

- unquoted boolean, Whether or not to quote fields. Default: false
- forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false
- filenamePrefix string, A user specified filename prefix for the output file to have. Default: null
- maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

`runningAsId` integer, The ID of the runner of this script.

`scripts_post_sql_clone` *Clone this SQL Script*

Description

Clone this SQL Script

Usage

```
scripts_post_sql_clone(
  id,
  clone_schedule = NULL,
  clone_triggers = NULL,
  clone_notifications = NULL
)
```

Arguments

`id`	integer required. The ID for the script.
`clone_schedule`	boolean optional. If true, also copy the schedule to the new script.
`clone_triggers`	boolean optional. If true, also copy the triggers to the new script.
`clone_notifications`	boolean optional. If true, also copy the notifications to the new script.

Value

A list containing the following elements:

`id`	integer, The ID for the script.
`name`	string, The name of the script.
`type`	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
`createdAt`	string, The time this script was created.
`updatedAt`	string, The time the script was last updated.
`author`	list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.

- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`hidden`
: boolean, The hidden status of the item.

`targetProjectId`
: integer, Target project to which script outputs will be added.

`archived`
: string, The archival status of the requested item(s).

`sql`
: string, The raw SQL query for the script.

`expandedArguments`
: list, Expanded arguments for use in injecting into different environments.

`remoteHostId`
: integer, The remote host ID that this script will connect to.

`credentialId`
: integer, The credential that this script will use.

`codePreview`
: string, The code that this script will run with arguments inserted.

`csvSettings`
: list, A list containing the following elements:

- includeHeader boolean, Whether or not to include headers in the output data. Default: true
- compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
- columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
- unquoted boolean, Whether or not to quote fields. Default: false
- forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false
- filenamePrefix string, A user specified filename prefix for the output file to have. Default: null
- maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

`runningAsId`
: integer, The ID of the runner of this script.

`scripts_post_sql_git_checkout`
: *Checkout content that the existing git_ref points to and save to the object*

Description

Checkout content that the existing git_ref points to and save to the object

Usage

```
scripts_post_sql_git_checkout(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

---

scripts_post_sql_git_checkout_latest

Checkout latest commit on the current branch of a script or workflow

---

Description

Checkout latest commit on the current branch of a script or workflow

Usage

```
scripts_post_sql_git_checkout_latest(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

scripts_post_sql_git_commits *Commit and push a new version of the file*

Description

Commit and push a new version of the file

Usage

```
scripts_post_sql_git_commits(id, content, message, file_hash)
```

Arguments

id	integer required. The ID of the item.
content	string required. The contents to commit to the file.
message	string required. A commit message describing the changes being made.
file_hash	string required. The full SHA of the file being replaced.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

scripts_post_sql_runs *Start a run*

Description

Start a run

Usage

```
scripts_post_sql_runs(id)
```

Arguments

id	integer required. The ID of the SQL job.

Value

A list containing the following elements:

id	integer, The ID of the run.
sqlId	integer, The ID of the SQL job.
state	string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested	boolean, True if run cancel requested, else false.
createdAt	string, The time the run was created.
startedAt	string, The time the run started at.
finishedAt	string, The time the run completed.
error	string, The error, if any, returned by the run.
inputs	list, The inputs of the script at the time of the run, including params and arguments.
output	array, An array containing the following fields: • outputName string, The name of the output file. • fileId integer, The unique ID of the output file. • path string, The temporary link to download this output file, valid for 36 hours.
outputCachedOn	string, The time that the output was originally exported, if a cache entry was used by the run.

`scripts_put_containers` *Edit a container*

Description

Edit a container

Usage

```
scripts_put_containers(
  id,
  required_resources,
  name = NULL,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
```

```
  repo_http_uri = NULL,
  repo_ref = NULL,
  remote_host_credential_id = NULL,
  git_credential_id = NULL,
  docker_command = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  instance_type = NULL,
  cancel_timeout = NULL,
  time_zone = NULL,
  partition_label = NULL,
  target_project_id = NULL,
  running_as_id = NULL
)
```

Arguments

id	integer required. The ID for the script.
required_resources	list required. A list containing the following elements: • cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares. • memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB. • diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.
name	string optional. The name of the container.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.

- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`schedule` list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`repo_http_uri` string optional. The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

`repo_ref` string optional. The tag or branch of the github repo to clone into the container.

`remote_host_credential_id`
integer optional. The id of the database credentials to pass into the environment of the container.

`git_credential_id`
integer optional. The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

`docker_command` string optional. The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image's ENTRYPOINT/CMD.

`docker_image_name`
string optional. The name of the docker image to pull from DockerHub.

`docker_image_tag`
string optional. The tag of the docker image to pull from DockerHub.

`instance_type` string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancel_timeout` integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`time_zone` string optional. The time zone of this script.

`partition_label`
string optional. The partition label used to run this object.

`target_project_id`
integer optional. Target project to which script outputs will be added.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`fromTemplateAliases`
array, An array containing the following fields:

- id integer, The id of the Alias object.
- objectId integer, The id of the object
- alias string, The alias of the object

`name` string, The name of the container.

`type` string, The type of the script (e.g Container)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template script.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.

- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`requiredResources`
list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`repoHttpUri` string, The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

`repoRef` string, The tag or branch of the github repo to clone into the container.

`remoteHostCredentialId`
integer, The id of the database credentials to pass into the environment of the container.

`gitCredentialId`
: integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

`dockerCommand`
: string, The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image's ENTRYPOINT/CMD.

`dockerImageName`
: string, The name of the docker image to pull from DockerHub.

`dockerImageTag`
: string, The tag of the docker image to pull from DockerHub.

`instanceType`
: string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout`
: integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`lastRun`
: list, A list containing the following elements:

  - id integer,
  - state string,
  - createdAt string, The time that the run was queued.
  - startedAt string, The time that the run started.
  - finishedAt string, The time that the run completed.
  - error string, The error message for this run, if present.

`timeZone`
: string, The time zone of this script.

`partitionLabel`
: string, The partition label used to run this object.

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`hidden`
: boolean, The hidden status of the item.

`archived`
: string, The archival status of the requested item(s).

`targetProjectId`
: integer, Target project to which script outputs will be added.

`runningAsId`
: integer, The ID of the runner of this script.

`scripts_put_containers_archive`
Update the archive status of this object

Description

Update the archive status of this object

Usage

```
scripts_put_containers_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ID for the script.
fromTemplateAliases	array, An array containing the following fields: • id integer, The id of the Alias object. • objectId integer, The id of the object • alias string, The alias of the object
name	string, The name of the container.
type	string, The type of the script (e.g Container)
createdAt	string, The time this script was created.
updatedAt	string, The time the script was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the script's last run.
finishedAt	string, The time that the script's last run finished.
category	string, The category of the script.
projects	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
parentId	integer, The ID of the parent job that will trigger this script
userContext	string, "runner" or "author", who to execute the script as when run as a template.
params	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required.

- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template script.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.

- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`requiredResources`
list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`repoHttpUri` string, The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

`repoRef` string, The tag or branch of the github repo to clone into the container.

`remoteHostCredentialId`
integer, The id of the database credentials to pass into the environment of the container.

`gitCredentialId`
integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

`dockerCommand` string, The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image's ENTRYPOINT/CMD.

`dockerImageName`
string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`timeZone` string, The time zone of this script.

`partitionLabel` string, The partition label used to run this object.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`targetProjectId` integer, Target project to which script outputs will be added.

`runningAsId` integer, The ID of the runner of this script.

`scripts_put_containers_projects`
Add a Container Script to a project

Description

Add a Container Script to a project

Usage

```
scripts_put_containers_projects(id, project_id)
```

Arguments

`id`	integer required. The ID of the Container Script.
`project_id`	integer required. The ID of the project.

Value

An empty HTTP response

scripts_put_containers_shares_groups
Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
scripts_put_containers_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_put_containers_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
scripts_put_containers_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers
: list, A list containing the following elements:
- users array,
- groups array,

writers
: list, A list containing the following elements:
- users array,
- groups array,

owners
: list, A list containing the following elements:
- users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_put_containers_transfer` *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
scripts_put_containers_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies`
: array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object

- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`scripts_put_custom` *Replace all attributes of this Custom Script*

Description

Replace all attributes of this Custom Script

Usage

```
scripts_put_custom(
  id,
  name = NULL,
  parent_id = NULL,
  arguments = NULL,
  remote_host_id = NULL,
  credential_id = NULL,
  schedule = NULL,
  notifications = NULL,
  time_zone = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  partition_label = NULL,
  running_as_id = NULL
)
```

Arguments

`id`	integer required. The ID for the script.
`name`	string optional. The name of the script.
`parent_id`	integer optional. The ID of the parent job that will trigger this script
`arguments`	list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
`remote_host_id`	integer optional. The remote host ID that this script will connect to.
`credential_id`	integer optional. The credential that this script will use.
`schedule`	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth

- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`time_zone` string optional. The time zone of this script.

`target_project_id` integer optional. Target project to which script outputs will be added.

`required_resources` list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
- memory integer, The amount of RAM to allocate for the container (in MB).
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`partition_label` string optional. The partition label used to run this object. Only applicable for jobs using Docker.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`fromTemplateAliases`
array, An array containing the following fields:

- id integer, The id of the Alias object.
- objectId integer, The id of the object
- alias string, The alias of the object

`name` string, The name of the script.

`type` string, The type of the script (e.g Custom)

`backingScriptType`
string, The type of the script backing this template (e.g Python)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string,

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.

- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template script.

`uiReportUrl` integer, The url of the custom HTML.

`uiReportId` integer, The id of the report with the custom HTML.

`uiReportProvideAPIKey` boolean, Whether the ui report requests an API Key from the report viewer.

`templateScriptName` string, The name of the template script.

`templateNote` string, The template's note.

`remoteHostId` integer, The remote host ID that this script will connect to.

`credentialId` integer, The credential that this script will use.

`codePreview` string, The code that this script will run with arguments inserted.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.

- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`archived` string, The archival status of the requested item(s).

`targetProjectId` integer, Target project to which script outputs will be added.

`lastSuccessfulRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
- memory integer, The amount of RAM to allocate for the container (in MB).
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`partitionLabel` string, The partition label used to run this object. Only applicable for jobs using Docker.

`runningAsId` integer, The ID of the runner of this script.

`scripts_put_custom_archive`
Update the archive status of this object

Description

Update the archive status of this object

Usage

```
scripts_put_custom_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`fromTemplateAliases` array, An array containing the following fields:

- id integer, The id of the Alias object.
- objectId integer, The id of the object
- alias string, The alias of the object

`name` string, The name of the script.

`type` string, The type of the script (e.g Custom)

`backingScriptType` string, The type of the script backing this template (e.g Python)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string,

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template script.

`uiReportUrl` integer, The url of the custom HTML.

`uiReportId` integer, The id of the report with the custom HTML.

`uiReportProvideAPIKey` boolean, Whether the ui report requests an API Key from the report viewer.

`templateScriptName` string, The name of the template script.

`templateNote` string, The template's note.

`remoteHostId` integer, The remote host ID that this script will connect to.

`credentialId` integer, The credential that this script will use.

`codePreview` string, The code that this script will run with arguments inserted.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`hidden`
: boolean, The hidden status of the item.

`archived`
: string, The archival status of the requested item(s).

`targetProjectId`
: integer, Target project to which script outputs will be added.

`lastSuccessfulRun`
: list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`requiredResources`
: list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
- memory integer, The amount of RAM to allocate for the container (in MB).
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`partitionLabel`
: string, The partition label used to run this object. Only applicable for jobs using Docker.

`runningAsId`
: integer, The ID of the runner of this script.

`scripts_put_custom_projects`
Add a Custom Script to a project

Description

Add a Custom Script to a project

Usage

```
scripts_put_custom_projects(id, project_id)
```

Arguments

`id`
: integer required. The ID of the Custom Script.

`project_id`
: integer required. The ID of the project.

Value

An empty HTTP response

`scripts_put_custom_shares_groups`
Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
scripts_put_custom_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_put_custom_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
scripts_put_custom_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_ids`	array required. An array of one or more user IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`
: list, A list containing the following elements:

- users array,
- groups array,

`writers`
: list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares`
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_put_custom_transfer`
Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
scripts_put_custom_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

scripts_put_dbt *Replace all attributes of this dbt Script*

Description

Replace all attributes of this dbt Script

Usage

```
scripts_put_dbt(
  id,
  name,
  repo_http_uri,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  instance_type = NULL,
  cancel_timeout = NULL,
  docker_image_tag = NULL,
  partition_label = NULL,
  running_as_id = NULL,
  dbt_project = NULL,
  repo_ref = NULL,
  target_database = NULL
)
```

Arguments

id	integer required. The ID for the script.
name	string required. The name of the script.
repo_http_uri	string required. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
arguments	list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.

- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`target_project_id`
integer optional. Target project to which script outputs will be added.

`required_resources`
list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instance_type` string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancel_timeout` integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`docker_image_tag`
string optional. The tag of the docker image to pull from DockerHub.

`partition_label`
string optional. The partition label used to run this object.

`running_as_id` integer optional. The ID of the runner of this script.

`dbt_project` list optional. A list containing the following elements:

- target string, Which profile target to use. Ignored when used in conjunction with generate_profiles.
- schema string, The output schema for dbt to use.
- projectDir string, The path to dbt_project.yml. Defaults to the root of the repository. Generates 'DBT_PROJECT_DIR' environment variable.

- profilesDir string, The path to the profiles.yml file to be used by dbt. Ignored when used in conjunction with generate_profiles. Generates 'DBT_PROFILES_DIR' environment variable.
- dbtVersion string, The version of dbt to use. Generates 'DBT_VERSION' environment variable.
- dbtCommand string, The primary dbt command to run. Valid commands are build, run, test, compile, and retry.
- dbtCommandLineArgs string, Additional command line arguments to pass to dbt. Ignored when dbt retry command is selected.
- docsReportId string, The ID of the HTML report hosting the static dbt docs for this job. Updates every time a run succeeds. This report will be automatically shared with all users who are shared on the job.
- skipDocsGeneration boolean, Whether to skip dbt docs generation. If true, the linked docs report will not be updated when the script runs. Defaults to false.
- generateProfiles boolean, Whether to generate the profiles.yml file when running the script. Defaults to false.

`repo_ref` string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, a tag, or the full or shortened SHA of a commit. Defaults to 'main'.

`target_database` list optional. A list containing the following elements:

- remoteHostId integer, The remote host ID that this script will connect to.
- credentialId integer, The credential that this script will use.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.

- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources`
: list, A list containing the following elements:
  - cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  - memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
  - diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType`
: string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout`
: integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`dockerImageTag`
: string, The tag of the docker image to pull from DockerHub.

`partitionLabel`
: string, The partition label used to run this object.

`runningAsId`
: integer, The ID of the runner of this script.

`dbtProject`
: list, A list containing the following elements:
  - target string, Which profile target to use. Ignored when used in conjunction with generate_profiles.
  - schema string, The output schema for dbt to use.
  - projectDir string, The path to dbt_project.yml. Defaults to the root of the repository. Generates 'DBT_PROJECT_DIR' environment variable.
  - profilesDir string, The path to the profiles.yml file to be used by dbt. Ignored when used in conjunction with generate_profiles. Generates 'DBT_PROFILES_DIR' environment variable.
  - dbtVersion string, The version of dbt to use. Generates 'DBT_VERSION' environment variable.
  - dbtCommand string, The primary dbt command to run. Valid commands are build, run, test, compile, and retry.
  - dbtCommandLineArgs string, Additional command line arguments to pass to dbt. Ignored when dbt retry command is selected.
  - docsReportId string, The ID of the HTML report hosting the static dbt docs for this job. Updates every time a run succeeds. This report will be automatically shared with all users who are shared on the job.
  - skipDocsGeneration boolean, Whether to skip dbt docs generation. If true, the linked docs report will not be updated when the script runs. Defaults to false.
  - generateProfiles boolean, Whether to generate the profiles.yml file when running the script. Defaults to false.

`repoHttpUri`
: string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`repoRef`
: string, A git reference specifying an unambiguous version of the file. Can be a branch name, a tag, or the full or shortened SHA of a commit. Defaults to 'main'.

`targetDatabase` list, A list containing the following elements:

- remoteHostId integer, The remote host ID that this script will connect to.
- credentialId integer, The credential that this script will use.

`scripts_put_dbt_archive`

Update the archive status of this object

Description

Update the archive status of this object

Usage

```
scripts_put_dbt_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id`	integer, The ID for the script.
`name`	string, The name of the script.
`type`	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
`createdAt`	string, The time this script was created.
`updatedAt`	string, The time the script was last updated.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`state`	string, The status of the script's last run.
`finishedAt`	string, The time that the script's last run finished.
`category`	string, The category of the script.
`projects`	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.

- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`partitionLabel` string, The partition label used to run this object.

`runningAsId` integer, The ID of the runner of this script.

`dbtProject` list, A list containing the following elements:

- target string, Which profile target to use. Ignored when used in conjunction with generate_profiles.
- schema string, The output schema for dbt to use.
- projectDir string, The path to dbt_project.yml. Defaults to the root of the repository. Generates 'DBT_PROJECT_DIR' environment variable.
- profilesDir string, The path to the profiles.yml file to be used by dbt. Ignored when used in conjunction with generate_profiles. Generates 'DBT_PROFILES_DIR' environment variable.
- dbtVersion string, The version of dbt to use. Generates 'DBT_VERSION' environment variable.
- dbtCommand string, The primary dbt command to run. Valid commands are build, run, test, compile, and retry.
- dbtCommandLineArgs string, Additional command line arguments to pass to dbt. Ignored when dbt retry command is selected.
- docsReportId string, The ID of the HTML report hosting the static dbt docs for this job. Updates every time a run succeeds. This report will be automatically shared with all users who are shared on the job.
- skipDocsGeneration boolean, Whether to skip dbt docs generation. If true, the linked docs report will not be updated when the script runs. Defaults to false.
- generateProfiles boolean, Whether to generate the profiles.yml file when running the script. Defaults to false.

`repoHttpUri` string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`repoRef` string, A git reference specifying an unambiguous version of the file. Can be a branch name, a tag, or the full or shortened SHA of a commit. Defaults to 'main'.

`targetDatabase` list, A list containing the following elements:

- remoteHostId integer, The remote host ID that this script will connect to.
- credentialId integer, The credential that this script will use.

scripts_put_dbt_projects *Add a dbt Script to a project*

Description

Add a dbt Script to a project

Usage

```
scripts_put_dbt_projects(id, project_id)
```

Arguments

id	integer required. The ID of the dbt Script.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

scripts_put_dbt_shares_groups *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
scripts_put_dbt_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
`totalGroupShares`	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_put_dbt_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
scripts_put_dbt_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

scripts_put_dbt_transfer

Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
scripts_put_dbt_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. ID of target user
include_dependencies	boolean required. Whether or not to give manage permissions on all dependencies
email_body	string optional. Custom body text for e-mail sent on transfer.
send_email	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies — array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

scripts_put_javascript

Replace all attributes of this JavaScript Script

Description

Replace all attributes of this JavaScript Script

Usage

```
scripts_put_javascript(
  id,
  name,
  source,
  remote_host_id,
  credential_id,
  parent_id = NULL,
  user_context = NULL,
```

```
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  target_project_id = NULL,
  running_as_id = NULL
)
```

Arguments

id	integer required. The ID for the script.
name	string required. The name of the script.
source	string required. The body/text of the script.
remote_host_id	integer required. The remote host ID that this script will connect to.
credential_id	integer required. The credential that this script will use.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
arguments	list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth

- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`target_project_id` integer optional. Target project to which script outputs will be added.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.

- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
hidden	boolean, The hidden status of the item.
targetProjectId	integer, Target project to which script outputs will be added.
archived	string, The archival status of the requested item(s).
source	string, The body/text of the script.
remoteHostId	integer, The remote host ID that this script will connect to.
credentialId	integer, The credential that this script will use.
runningAsId	integer, The ID of the runner of this script.

`scripts_put_javascript_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
scripts_put_javascript_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ID for the script.
name	string, The name of the script.
type	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
createdAt	string, The time this script was created.
updatedAt	string, The time the script was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.

- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

hidden	boolean, The hidden status of the item.
targetProjectId	integer, Target project to which script outputs will be added.
archived	string, The archival status of the requested item(s).
source	string, The body/text of the script.
remoteHostId	integer, The remote host ID that this script will connect to.
credentialId	integer, The credential that this script will use.
runningAsId	integer, The ID of the runner of this script.

`scripts_put_javascript_git` *Attach an item to a file in a git repo*

Description

Attach an item to a file in a git repo

Usage

```
scripts_put_javascript_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,
gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

---

scripts_put_javascript_projects

Add a JavaScript Script to a project

Description

Add a JavaScript Script to a project

Usage

```
scripts_put_javascript_projects(id, project_id)
```

Arguments

id	integer required. The ID of the JavaScript Script.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

`scripts_put_javascript_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
scripts_put_javascript_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_put_javascript_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
scripts_put_javascript_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_ids`	array required. An array of one or more user IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_put_javascript_transfer`
Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
scripts_put_javascript_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies`
: array, An array containing the following fields:
  - objectType string, Dependent object type
  - fcoType string, Human readable dependent object type
  - id integer, Dependent object ID
  - name string, Dependent object name, or nil if the requesting user cannot read this object

- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`scripts_put_python3` *Replace all attributes of this Python Script*

Description

Replace all attributes of this Python Script

Usage

```
scripts_put_python3(
  id,
  name,
  source,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  instance_type = NULL,
  cancel_timeout = NULL,
  docker_image_tag = NULL,
  partition_label = NULL,
  running_as_id = NULL
)
```

Arguments

`id`	integer required. The ID for the script.
`name`	string required. The name of the script.
`source`	string required. The body/text of the script.
`parent_id`	integer optional. The ID of the parent job that will trigger this script
`user_context`	string optional. "runner" or "author", who to execute the script as when run as a template.

`params` array optional. An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`schedule` list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.

- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`target_project_id`
integer optional. Target project to which script outputs will be added.

`required_resources`
list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instance_type` string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancel_timeout` integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`docker_image_tag`
string optional. The tag of the docker image to pull from DockerHub.

`partition_label`
string optional. The partition label used to run this object.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.

- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.

- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

hidden boolean, The hidden status of the item.

targetProjectId
integer, Target project to which script outputs will be added.

archived string, The archival status of the requested item(s).

requiredResources
list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

dockerImageTag string, The tag of the docker image to pull from DockerHub.

partitionLabel string, The partition label used to run this object.

runningAsId integer, The ID of the runner of this script.

source string, The body/text of the script.

`scripts_put_python3_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
scripts_put_python3_archive(id, status)
```

Arguments

id integer required. The ID of the object.

status boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ID for the script.
name	string, The name of the script.
type	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
createdAt	string, The time this script was created.
updatedAt	string, The time the script was last updated.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The status of the script's last run.
finishedAt	string, The time that the script's last run finished.
category	string, The category of the script.
projects	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
parentId	integer, The ID of the parent job that will trigger this script
userContext	string, "runner" or "author", who to execute the script as when run as a template.
params	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
arguments	list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.

- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`partitionLabel` string, The partition label used to run this object.

`runningAsId` integer, The ID of the runner of this script.

`source` string, The body/text of the script.

scripts_put_python3_git

Attach an item to a file in a git repo

Description

Attach an item to a file in a git repo

Usage

```
scripts_put_python3_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string,

- updatedAt string,

gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

`scripts_put_python3_projects`
Add a Python Script to a project

Description

Add a Python Script to a project

Usage

```
scripts_put_python3_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Python Script.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

`scripts_put_python3_shares_groups`
Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
scripts_put_python3_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
`totalGroupShares`	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_put_python3_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
scripts_put_python3_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

scripts_put_python3_transfer

Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
scripts_put_python3_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. ID of target user
include_dependencies	boolean required. Whether or not to give manage permissions on all dependencies
email_body	string optional. Custom body text for e-mail sent on transfer.
send_email	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies — array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

scripts_put_r *Replace all attributes of this R Script*

Description

Replace all attributes of this R Script

Usage

```
scripts_put_r(
  id,
  name,
  source,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
```

```
  next_run_at = NULL,
  time_zone = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  instance_type = NULL,
  cancel_timeout = NULL,
  docker_image_tag = NULL,
  partition_label = NULL,
  running_as_id = NULL
)
```

Arguments

id	integer required. The ID for the script.
name	string required. The name of the script.
source	string required. The body/text of the script.
parent_id	integer optional. The ID of the parent job that will trigger this script
user_context	string optional. "runner" or "author", who to execute the script as when run as a template.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
arguments	list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on.

- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`target_project_id` integer optional. Target project to which script outputs will be added.

`required_resources` list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instance_type` string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancel_timeout` integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`docker_image_tag`
: string optional. The tag of the docker image to pull from DockerHub.

`partition_label`
: string optional. The partition label used to run this object.

`running_as_id`
: integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id`
: integer, The ID for the script.

`name`
: string, The name of the script.

`type`
: string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt`
: string, The time this script was created.

`updatedAt`
: string, The time the script was last updated.

`author`
: list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

`state`
: string, The status of the script's last run.

`finishedAt`
: string, The time that the script's last run finished.

`category`
: string, The category of the script.

`projects`
: array, An array containing the following fields:
  - id integer, The ID for the project.
  - name string, The name of the project.

`parentId`
: integer, The ID of the parent job that will trigger this script

`userContext`
: string, "runner" or "author", who to execute the script as when run as a template.

`params`
: array, An array containing the following fields:
  - name string, The variable's name as used within your code.
  - label string, The label to present to users when asking them for the value.
  - description string, A short sentence or fragment describing this parameter to the end user.
  - type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - required boolean, Whether this param is required.
  - value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.

- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.

- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

`cancelTimeout` integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub.

`partitionLabel` string, The partition label used to run this object.

`runningAsId` integer, The ID of the runner of this script.

`source` string, The body/text of the script.

`scripts_put_r_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
scripts_put_r_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id`	integer, The ID for the script.
`name`	string, The name of the script.
`type`	string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)
`createdAt`	string, The time this script was created.
`updatedAt`	string, The time the script was last updated.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`state`	string, The status of the script's last run.
`finishedAt`	string, The time that the script's last run finished.
`category`	string, The category of the script.
`projects`	array, An array containing the following fields: • id integer, The ID for the project. • name string, The name of the project.
`parentId`	integer, The ID of the parent job that will trigger this script
`userContext`	string, "runner" or "author", who to execute the script as when run as a template.
`params`	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value.

- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.

- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.

- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType	string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
cancelTimeout	integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
dockerImageTag	string, The tag of the docker image to pull from DockerHub.
partitionLabel	string, The partition label used to run this object.
runningAsId	integer, The ID of the runner of this script.
source	string, The body/text of the script.

`scripts_put_r_git` *Attach an item to a file in a git repo*

Description

Attach an item to a file in a git repo

Usage

```
scripts_put_r_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,
gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

`scripts_put_r_projects`

Add an R Script to a project

Description

Add an R Script to a project

Usage

```
scripts_put_r_projects(id, project_id)
```

Arguments

id	integer required. The ID of the R Script.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

scripts_put_r_shares_groups
Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
scripts_put_r_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_put_r_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
scripts_put_r_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers
: list, A list containing the following elements:
- users array,
- groups array,

writers
: list, A list containing the following elements:
- users array,
- groups array,

owners
: list, A list containing the following elements:
- users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`scripts_put_r_transfer`
Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
scripts_put_r_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies`
: array, An array containing the following fields:
  - objectType string, Dependent object type
  - fcoType string, Human readable dependent object type
  - id integer, Dependent object ID
  - name string, Dependent object name, or nil if the requesting user cannot read this object

- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`scripts_put_sql` *Replace all attributes of this SQL Script*

Description

Replace all attributes of this SQL Script

Usage

```
scripts_put_sql(
  id,
  name,
  sql,
  remote_host_id,
  credential_id,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  target_project_id = NULL,
  csv_settings = NULL,
  running_as_id = NULL
)
```

Arguments

`id`	integer required. The ID for the script.
`name`	string required. The name of the script.
`sql`	string required. The raw SQL query for the script.
`remote_host_id`	integer required. The remote host ID that this script will connect to.
`credential_id`	integer required. The credential that this script will use.
`parent_id`	integer optional. The ID of the parent job that will trigger this script
`user_context`	string optional. "runner" or "author", who to execute the script as when run as a template.

`params` array optional. An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`schedule` list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.

- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`next_run_at` string optional. The time of the next scheduled run.

`time_zone` string optional. The time zone of this script.

`target_project_id`
integer optional. Target project to which script outputs will be added.

`csv_settings` list optional. A list containing the following elements:

- includeHeader boolean, Whether or not to include headers in the output data. Default: true
- compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
- columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
- unquoted boolean, Whether or not to quote fields. Default: false
- forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false
- filenamePrefix string, A user specified filename prefix for the output file to have. Default: null
- maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

`running_as_id` integer optional. The ID of the runner of this script.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId` integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount` integer, How many other scripts use this one as a template.

`templateScriptName` string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.

- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

archived	string, The archival status of the requested item(s).
sql	string, The raw SQL query for the script.
expandedArguments	list, Expanded arguments for use in injecting into different environments.
remoteHostId	integer, The remote host ID that this script will connect to.
credentialId	integer, The credential that this script will use.
codePreview	string, The code that this script will run with arguments inserted.
csvSettings	list, A list containing the following elements: • includeHeader boolean, Whether or not to include headers in the output data. Default: true • compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip • columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma • unquoted boolean, Whether or not to quote fields. Default: false • forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false • filenamePrefix string, A user specified filename prefix for the output file to have. Default: null • maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.
runningAsId	integer, The ID of the runner of this script.

`scripts_put_sql_archive`

Update the archive status of this object

Description

Update the archive status of this object

Usage

```
scripts_put_sql_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id` integer, The ID for the script.

`name` string, The name of the script.

`type` string, The type of the script (e.g SQL, Container, Python, R, JavaScript, dbt)

`createdAt` string, The time this script was created.

`updatedAt` string, The time the script was last updated.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`state` string, The status of the script's last run.

`finishedAt` string, The time that the script's last run finished.

`category` string, The category of the script.

`projects` array, An array containing the following fields:

- id integer, The ID for the project.
- name string, The name of the project.

`parentId` integer, The ID of the parent job that will trigger this script

`userContext` string, "runner" or "author", who to execute the script as when run as a template.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`arguments` list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

`isTemplate` boolean, Whether others scripts use this one as a template.

`publishedAsTemplateId`
integer, The ID of the template that this script is backing.

`fromTemplateId` integer, The ID of the template this script uses, if any.

`templateDependentsCount`
integer, How many other scripts use this one as a template.

`templateScriptName`
string, The name of the template script.

`links` list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on. Defaults to user's preferences.
- failureOn boolean, If failure email notifications are on. Defaults to user's preferences.

`runningAs` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.

- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`sql` string, The raw SQL query for the script.

`expandedArguments` list, Expanded arguments for use in injecting into different environments.

`remoteHostId` integer, The remote host ID that this script will connect to.

`credentialId` integer, The credential that this script will use.

`codePreview` string, The code that this script will run with arguments inserted.

`csvSettings` list, A list containing the following elements:

- includeHeader boolean, Whether or not to include headers in the output data. Default: true
- compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
- columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
- unquoted boolean, Whether or not to quote fields. Default: false
- forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false
- filenamePrefix string, A user specified filename prefix for the output file to have. Default: null
- maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

`runningAsId` integer, The ID of the runner of this script.

scripts_put_sql_git *Attach an item to a file in a git repo*

Description

Attach an item to a file in a git repo

Usage

```
scripts_put_sql_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,

gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

`scripts_put_sql_projects` *Add a SQL Script to a project*

Description

Add a SQL Script to a project

Usage

```
scripts_put_sql_projects(id, project_id)
```

Arguments

id	integer required. The ID of the SQL Script.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

`scripts_put_sql_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
scripts_put_sql_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

scripts_put_sql_shares_users

Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

```
scripts_put_sql_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

scripts_put_sql_transfer

Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
scripts_put_sql_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. ID of target user
include_dependencies	boolean required. Whether or not to give manage permissions on all dependencies
email_body	string optional. Custom body text for e-mail sent on transfer.
send_email	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies — array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

search_list *Perform a search*

Description

Perform a search

Usage

```
search_list(
  query = NULL,
  type = NULL,
  offset = NULL,
  order = NULL,
  owner = NULL,
  limit = NULL,
  archived = NULL,
  last_run_state = NULL
)
```

Arguments

query	string optional. The search query.
type	string optional. The type for the search. It accepts a comma-separated list. Valid arguments are listed on the "GET /search/types" endpoint.
offset	integer optional. The offset for the search results.
order	string optional. The field on which to order the result set.
owner	string optional. The owner for the search.
limit	integer optional. Defaults to 10. Maximum allowed is 1000.
archived	string optional. If specified, return only results with the chosen archived status; either 'true', 'false', or 'all'. Defaults to 'false'.
last_run_state	string optional. The last run state of the job being searched for; either: 'queued', 'running', 'succeeded', 'failed', or 'cancelled'.

Value

An array containing the following fields:

totalResults	integer, The number of items matching the search query.
aggregations	list, Aggregations by owner and type for the search results.
results	array, An array containing the following fields:

- score number, The relevance score from the search request.
- type string, The type of the item.
- id integer, The ID of the item.
- name string, The name of the item.
- typeName string, The verbose name of the type.
- updatedAt string, The time the item was last updated.
- owner string, The owner of the item.
- useCount integer, The use count of the item, if the item is a template.
- lastRunId integer, The last run id of the item, if the item is a job.
- lastRunState string, The last run state of the item, if the item is a job.
- lastRunStart string, The last run start time of the item, if the item is a job.
- lastRunFinish string, The last run finish time of the item, if the item is a job.
- public boolean, The flag that indicates a template is available to all users.
- lastRunException string, The exception of the item after the last run, if the item is a job.
- autoShare boolean, The flag that indicates if a project has Auto-Share enabled.

search_list_queries *Search queries that are not hidden*

Description

Search queries that are not hidden

Usage

```
search_list_queries(
  search_string = NULL,
  database_id = NULL,
  credential_id = NULL,
  author_id = NULL,
  archived = NULL,
  state = NULL,
  started_before = NULL,
  started_after = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

search_string	string optional. Space delimited search terms for searching queries by their SQL. Supports wild card characters "?" for any single character, and "*" for zero or more characters.
database_id	integer optional. The database ID.
credential_id	integer optional. The credential ID.
author_id	integer optional. The author of the query.
archived	boolean optional. The archival status of the requested item(s). Defaults to false.
state	array optional. The state of the last run. One or more of queued, running, succeeded, failed, and cancelled.
started_before	string optional. An upper bound for the start date of the last run.
started_after	string optional. A lower bound for the start date of the last run.
limit	integer optional. Number of results to return. Defaults to 10. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to last_run_started_at. Must be one of: last_run_started_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

`id`	integer, The query ID.
`database`	integer, The database ID.
`credential`	integer, The credential ID.
`sql`	string, The SQL executed by the query.
`authorId`	integer, The author of the query.
`archived`	boolean, The archival status of the requested item(s).
`createdAt`	string,
`updatedAt`	string,
`lastRun`	list, A list containing the following elements: • id integer, • state string, The state of the run. One of queued, running, succeeded, failed, and cancelled. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.

`search_list_types` *List available search types*

Description

List available search types

Usage

```
search_list_types()
```

Value

An array containing the following fields:

`type`	string, The name of the item type.

services_delete *Archive a Service (deprecated, use archiving endpoints instead)*

Description

Archive a Service (deprecated, use archiving endpoints instead)

Usage

```
services_delete(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

services_delete_deployments *Delete a Service deployment*

Description

Delete a Service deployment

Usage

```
services_delete_deployments(service_id, deployment_id)
```

Arguments

service_id	integer required. The ID of the owning Service
deployment_id	integer required. The ID for this deployment

Value

An empty HTTP response

`services_delete_projects`
Remove a Service from a project

Description

Remove a Service from a project

Usage

```
services_delete_projects(id, project_id)
```

Arguments

`id`	integer required. The ID of the Service.
`project_id`	integer required. The ID of the project.

Value

An empty HTTP response

`services_delete_shares_groups`
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
services_delete_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

`services_delete_shares_users` *Revoke the permissions a user has on this object*

Description

Revoke the permissions a user has on this object

Usage

```
services_delete_shares_users(id, user_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. The ID of the user.

Value

An empty HTTP response

`services_delete_tokens` *Revoke a token by id*

Description

Revoke a token by id

Usage

```
services_delete_tokens(id, token_id)
```

Arguments

`id`	integer required. The ID of the service.
`token_id`	integer required. The ID of the token.

Value

An empty HTTP response

services_get *Get a Service*

Description

Get a Service

Usage

```
services_get(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for this Service.
name	string, The name of this Service.
description	string, The description of this Service.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
type	string, The type of this Service
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
schedule	list, A list containing the following elements: • runtimePlan string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on. • recurrences array, List of day-hour combinations this item is scheduled for
timeZone	string,
replicas	integer, The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.

`maxReplicas` integer, The maximum number of Service replicas to deploy. Defining this field enables autoscaling.

`instanceType` string, The EC2 instance type to deploy to.

`memory` integer, The amount of memory allocated to each replica of the Service.

`cpu` integer, The amount of cpu allocated to each replica of the the Service.

`createdAt` string,

`updatedAt` string,

`credentials` array, The credentials attached to the service. Accepts a list of credential IDs and returns a list of id, name pairs.

`permissionSetId` integer, The ID of the associated permission set, if any.

`gitRepoUrl` string, The url for the git repo where the Service code lives.

`gitRepoRef` string, The git reference to use when pulling code from the repo.

`gitPathDir` string, The path to the Service code within the git repo. If unspecified, the root directory will be used.

`reportId` integer, The ID of the associated report.

`currentDeployment` list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- serviceId integer, The ID of owning Service

`currentUrl` string, The URL that the service is hosted at.

`environmentVariables` list, Environment Variables to be passed into the Service.

notifications	list, A list containing the following elements: • failureEmailAddresses array, Addresses to notify by e-mail when the service fails. • failureOn boolean, If failure email notifications are on
partitionLabel	string, The partition label used to run this object.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
archived	string, The archival status of the requested item(s).
hidden	boolean, The hidden status of the item.

`services_get_deployments` *Get details about a Service deployment*

Description

Get details about a Service deployment

Usage

```
services_get_deployments(service_id, deployment_id)
```

Arguments

service_id	integer required. The ID of the owning Service
deployment_id	integer required. The ID for this deployment

Value

A list containing the following elements:

deploymentId	integer, The ID for this deployment.
userId	integer, The ID of the owner.
host	string, Domain of the deployment.
name	string, Name of the deployment.
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
displayUrl	string, A signed URL for viewing the deployed item.
instanceType	string, The EC2 instance type requested for the deployment.
memory	integer, The memory allocated to the deployment, in MB.
cpu	integer, The cpu allocated to the deployment, in millicores.
state	string, The state of the deployment.

stateMessage	string, A detailed description of the state.
maxMemoryUsage	number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
maxCpuUsage	number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
createdAt	string,
updatedAt	string,
serviceId	integer, The ID of owning Service

`services_list` *List Services*

Description

List Services

Usage

```
services_list(
  hidden = NULL,
  archived = NULL,
  author = NULL,
  status = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
archived	string optional. The archival status of the requested item(s).
author	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.
status	string optional. If specified, returns Services with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'idle'.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

`id` integer, The ID for this Service.

`name` string, The name of this Service.

`description` string, The description of this Service.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`type` string, The type of this Service

`createdAt` string,

`updatedAt` string,

`gitRepoUrl` string, The url for the git repo where the Service code lives.

`gitRepoRef` string, The git reference to use when pulling code from the repo.

`gitPathDir` string, The path to the Service code within the git repo. If unspecified, the root directory will be used.

`currentDeployment` list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- serviceId integer, The ID of owning Service

`archived` string, The archival status of the requested item(s).

`services_list_dependencies` *List dependent objects for this object*

Description

List dependent objects for this object

Usage

```
services_list_dependencies(id, user_id = NULL)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer optional. ID of target user

Value

An array containing the following fields:

`objectType`	string, Dependent object type
`fcoType`	string, Human readable dependent object type
`id`	integer, Dependent object ID
`name`	string, Dependent object name, or nil if the requesting user cannot read this object
`permissionLevel`	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
`description`	string, Additional information about the dependency, if relevant
`shareable`	boolean, Whether or not the requesting user can share this object.

`services_list_deployments` *List deployments for a Service*

Description

List deployments for a Service

Usage

```
services_list_deployments(
  service_id,
  deployment_id = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

service_id	integer required. The ID of the owning Service
deployment_id	integer optional. The ID for this deployment
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

deploymentId	integer, The ID for this deployment.
userId	integer, The ID of the owner.
host	string, Domain of the deployment.
name	string, Name of the deployment.
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
instanceType	string, The EC2 instance type requested for the deployment.
memory	integer, The memory allocated to the deployment, in MB.
cpu	integer, The cpu allocated to the deployment, in millicores.
state	string, The state of the deployment.
stateMessage	string, A detailed description of the state.
maxMemoryUsage	number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
maxCpuUsage	number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.

createdAt	string,
updatedAt	string,
serviceId	integer, The ID of owning Service

`services_list_deployments_logs` *Get the logs for a Service deployment*

Description

Get the logs for a Service deployment

Usage

```
services_list_deployments_logs(
  id,
  deployment_id,
  start_at = NULL,
  end_at = NULL,
  limit = NULL
)
```

Arguments

id	integer required. The ID of the owning Service.
deployment_id	integer required. The ID for this deployment.
start_at	string optional. Log entries with a lower timestamp will be omitted.
end_at	string optional. Log entries with a higher timestamp will be omitted.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

message	string, The log message.
stream	string, The stream of the log. One of "stdout", "stderr".
createdAt	string, The time the log was created.
source	string, The source of the log. One of "system", "user".

`services_list_projects` *List the projects a Service belongs to*

Description

List the projects a Service belongs to

Usage

```
services_list_projects(id, hidden = NULL)
```

Arguments

`id`	integer required. The ID of the Service.
`hidden`	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

`id`	integer, The ID for this project.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The name of this project.
`description`	string, A description of the project.
`users`	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`autoShare`	boolean,
`createdAt`	string,
`updatedAt`	string,
`archived`	string, The archival status of the requested item(s).

`services_list_shares` *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
services_list_shares(id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

services_list_tokens *List tokens*

Description

List tokens

Usage

```
services_list_tokens(id)
```

Arguments

id	integer required. The ID of the service.

Value

An array containing the following fields:

id	integer, The ID of the token.
name	string, The name of the token.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
machineToken	boolean, If true, this token is not tied to a particular user.
expiresAt	string, The date and time when the token expires.
createdAt	string, The date and time when the token was created.

services_patch *Update some attributes of this Service*

Description

Update some attributes of this Service

Usage

```
services_patch(
  id,
  name = NULL,
  description = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  schedule = NULL,
  replicas = NULL,
  max_replicas = NULL,
  instance_type = NULL,
  memory = NULL,
  cpu = NULL,
  credentials = NULL,
  permission_set_id = NULL,
  git_repo_url = NULL,
  git_repo_ref = NULL,
  git_path_dir = NULL,
  environment_variables = NULL,
  notifications = NULL,
  partition_label = NULL
)
```

Arguments

id	integer required. The ID for this Service.
name	string optional. The name of this Service.
description	string optional. The description of this Service.
docker_image_name	string optional. The name of the docker image to pull from DockerHub.
docker_image_tag	string optional. The tag of the docker image to pull from DockerHub (default: latest).
schedule	list optional. A list containing the following elements: • runtimePlan string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on. • recurrences array, List of day-hour combinations this item is scheduled for
replicas	integer optional. The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.
max_replicas	integer optional. The maximum number of Service replicas to deploy. Defining this field enables autoscaling.
instance_type	string optional. The EC2 instance type to deploy to.

`memory` integer optional. The amount of memory allocated to each replica of the Service.

`cpu` integer optional. The amount of cpu allocated to each replica of the the Service.

`credentials` array optional. The credentials attached to the service. Accepts a list of credential IDs and returns a list of id, name pairs.

`permission_set_id` integer optional. The ID of the associated permission set, if any.

`git_repo_url` string optional. The url for the git repo where the Service code lives.

`git_repo_ref` string optional. The git reference to use when pulling code from the repo.

`git_path_dir` string optional. The path to the Service code within the git repo. If unspecified, the root directory will be used.

`environment_variables` list optional. Environment Variables to be passed into the Service.

`notifications` list optional. A list containing the following elements:

- failureEmailAddresses array, Addresses to notify by e-mail when the service fails.
- failureOn boolean, If failure email notifications are on

`partition_label` string optional. The partition label used to run this object.

Value

A list containing the following elements:

`id` integer, The ID for this Service.

`name` string, The name of this Service.

`description` string, The description of this Service.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`type` string, The type of this Service

`dockerImageName` string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).

`schedule` list, A list containing the following elements:

- runtimePlan string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on.
- recurrences array, List of day-hour combinations this item is scheduled for

`timeZone` string,

`replicas` integer, The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.

`maxReplicas` integer, The maximum number of Service replicas to deploy. Defining this field enables autoscaling.

`instanceType` string, The EC2 instance type to deploy to.

`memory` integer, The amount of memory allocated to each replica of the Service.

`cpu` integer, The amount of cpu allocated to each replica of the the Service.

`createdAt` string,

`updatedAt` string,

`credentials` array, The credentials attached to the service. Accepts a list of credential IDs and returns a list of id, name pairs.

`permissionSetId` integer, The ID of the associated permission set, if any.

`gitRepoUrl` string, The url for the git repo where the Service code lives.

`gitRepoRef` string, The git reference to use when pulling code from the repo.

`gitPathDir` string, The path to the Service code within the git repo. If unspecified, the root directory will be used.

`reportId` integer, The ID of the associated report.

`currentDeployment` list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- serviceId integer, The ID of owning Service

currentUrl string, The URL that the service is hosted at.

environmentVariables
list, Environment Variables to be passed into the Service.

notifications list, A list containing the following elements:

- failureEmailAddresses array, Addresses to notify by e-mail when the service fails.
- failureOn boolean, If failure email notifications are on

partitionLabel string, The partition label used to run this object.

myPermissionLevel
string, Your permission level on the object. One of "read", "write", or "manage".

archived string, The archival status of the requested item(s).

hidden boolean, The hidden status of the item.

---

services_post *Create a Service*

Description

Create a Service

Usage

```
services_post(
  name = NULL,
  description = NULL,
  type = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  schedule = NULL,
  replicas = NULL,
  max_replicas = NULL,
  instance_type = NULL,
  memory = NULL,
  cpu = NULL,
  credentials = NULL,
  permission_set_id = NULL,
  git_repo_url = NULL,
  git_repo_ref = NULL,
  git_path_dir = NULL,
  environment_variables = NULL,
  notifications = NULL,
  partition_label = NULL,
  hidden = NULL
)
```

Arguments

name	string optional. The name of this Service.
description	string optional. The description of this Service.
type	string optional. The type of this Service
docker_image_name	string optional. The name of the docker image to pull from DockerHub.
docker_image_tag	string optional. The tag of the docker image to pull from DockerHub (default: latest).
schedule	list optional. A list containing the following elements: • runtimePlan string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on. • recurrences array, List of day-hour combinations this item is scheduled for
replicas	integer optional. The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.
max_replicas	integer optional. The maximum number of Service replicas to deploy. Defining this field enables autoscaling.
instance_type	string optional. The EC2 instance type to deploy to.
memory	integer optional. The amount of memory allocated to each replica of the Service.
cpu	integer optional. The amount of cpu allocated to each replica of the the Service.
credentials	array optional. The credentials attached to the service. Accepts a list of credential IDs and returns a list of id, name pairs.
permission_set_id	integer optional. The ID of the associated permission set, if any.
git_repo_url	string optional. The url for the git repo where the Service code lives.
git_repo_ref	string optional. The git reference to use when pulling code from the repo.
git_path_dir	string optional. The path to the Service code within the git repo. If unspecified, the root directory will be used.
environment_variables	list optional. Environment Variables to be passed into the Service.
notifications	list optional. A list containing the following elements: • failureEmailAddresses array, Addresses to notify by e-mail when the service fails. • failureOn boolean, If failure email notifications are on
partition_label	string optional. The partition label used to run this object.
hidden	boolean optional. The hidden status of the item.

Value

A list containing the following elements:

`id` integer, The ID for this Service.

`name` string, The name of this Service.

`description` string, The description of this Service.

`user` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`type` string, The type of this Service

`dockerImageName` string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).

`schedule` list, A list containing the following elements:

- runtimePlan string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on.
- recurrences array, List of day-hour combinations this item is scheduled for

`timeZone` string,

`replicas` integer, The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.

`maxReplicas` integer, The maximum number of Service replicas to deploy. Defining this field enables autoscaling.

`instanceType` string, The EC2 instance type to deploy to.

`memory` integer, The amount of memory allocated to each replica of the Service.

`cpu` integer, The amount of cpu allocated to each replica of the the Service.

`createdAt` string,

`updatedAt` string,

`credentials` array, The credentials attached to the service. Accepts a list of credential IDs and returns a list of id, name pairs.

`permissionSetId` integer, The ID of the associated permission set, if any.

`gitRepoUrl` string, The url for the git repo where the Service code lives.

`gitRepoRef` string, The git reference to use when pulling code from the repo.

`gitPathDir` string, The path to the Service code within the git repo. If unspecified, the root directory will be used.

`reportId` integer, The ID of the associated report.

`currentDeployment`
list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- serviceId integer, The ID of owning Service

`currentUrl` string, The URL that the service is hosted at.

`environmentVariables`
list, Environment Variables to be passed into the Service.

`notifications` list, A list containing the following elements:

- failureEmailAddresses array, Addresses to notify by e-mail when the service fails.
- failureOn boolean, If failure email notifications are on

`partitionLabel` string, The partition label used to run this object.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`hidden` boolean, The hidden status of the item.

services_post_clone *Clone this Service*

Description

Clone this Service

Usage

```
services_post_clone(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for this Service.
name	string, The name of this Service.
description	string, The description of this Service.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
type	string, The type of this Service
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
schedule	list, A list containing the following elements: • runtimePlan string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on. • recurrences array, List of day-hour combinations this item is scheduled for
timeZone	string,
replicas	integer, The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.

`maxReplicas` integer, The maximum number of Service replicas to deploy. Defining this field enables autoscaling.

`instanceType` string, The EC2 instance type to deploy to.

`memory` integer, The amount of memory allocated to each replica of the Service.

`cpu` integer, The amount of cpu allocated to each replica of the the Service.

`createdAt` string,

`updatedAt` string,

`credentials` array, The credentials attached to the service. Accepts a list of credential IDs and returns a list of id, name pairs.

`permissionSetId` integer, The ID of the associated permission set, if any.

`gitRepoUrl` string, The url for the git repo where the Service code lives.

`gitRepoRef` string, The git reference to use when pulling code from the repo.

`gitPathDir` string, The path to the Service code within the git repo. If unspecified, the root directory will be used.

`reportId` integer, The ID of the associated report.

`currentDeployment` list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- serviceId integer, The ID of owning Service

`currentUrl` string, The URL that the service is hosted at.

`environmentVariables` list, Environment Variables to be passed into the Service.

notifications	list, A list containing the following elements: • failureEmailAddresses array, Addresses to notify by e-mail when the service fails. • failureOn boolean, If failure email notifications are on
partitionLabel	string, The partition label used to run this object.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
archived	string, The archival status of the requested item(s).
hidden	boolean, The hidden status of the item.

`services_post_deployments` *Deploy a Service*

Description

Deploy a Service

Usage

```
services_post_deployments(service_id, deployment_id = NULL)
```

Arguments

service_id	integer required. The ID of the owning Service
deployment_id	integer optional. The ID for this deployment

Value

A list containing the following elements:

deploymentId	integer, The ID for this deployment.
userId	integer, The ID of the owner.
host	string, Domain of the deployment.
name	string, Name of the deployment.
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
displayUrl	string, A signed URL for viewing the deployed item.
instanceType	string, The EC2 instance type requested for the deployment.
memory	integer, The memory allocated to the deployment, in MB.
cpu	integer, The cpu allocated to the deployment, in millicores.
state	string, The state of the deployment.

stateMessage	string, A detailed description of the state.
maxMemoryUsage	number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
maxCpuUsage	number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
createdAt	string,
updatedAt	string,
serviceId	integer, The ID of owning Service

`services_post_redeploy` *Redeploy a Service*

Description

Redeploy a Service

Usage

```
services_post_redeploy(service_id, deployment_id = NULL)
```

Arguments

service_id	integer required. The ID of the owning Service
deployment_id	integer optional. The ID for this deployment

Value

A list containing the following elements:

deploymentId	integer, The ID for this deployment.
userId	integer, The ID of the owner.
host	string, Domain of the deployment.
name	string, Name of the deployment.
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
displayUrl	string, A signed URL for viewing the deployed item.
instanceType	string, The EC2 instance type requested for the deployment.
memory	integer, The memory allocated to the deployment, in MB.
cpu	integer, The cpu allocated to the deployment, in millicores.
state	string, The state of the deployment.
stateMessage	string, A detailed description of the state.

maxMemoryUsage	number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
maxCpuUsage	number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
createdAt	string,
updatedAt	string,
serviceId	integer, The ID of owning Service

`services_post_tokens` *Create a new long-lived service token*

Description

Create a new long-lived service token

Usage

```
services_post_tokens(id, name, machine_token = NULL, expires_in = NULL)
```

Arguments

id	integer required. The ID of the service.
name	string required. The name of the token.
machine_token	boolean optional. If true, create a compact token with no user information.
expires_in	integer optional. The number of seconds until the token should expire

Value

A list containing the following elements:

id	integer, The ID of the token.
name	string, The name of the token.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
machineToken	boolean, If true, this token is not tied to a particular user.
expiresAt	string, The date and time when the token expires.
createdAt	string, The date and time when the token was created.
token	string, The value of the token. Only returned when the token is first created.

services_put *Replace all attributes of this Service*

Description

Replace all attributes of this Service

Usage

```
services_put(
  id,
  name = NULL,
  description = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  schedule = NULL,
  replicas = NULL,
  max_replicas = NULL,
  instance_type = NULL,
  memory = NULL,
  cpu = NULL,
  credentials = NULL,
  permission_set_id = NULL,
  git_repo_url = NULL,
  git_repo_ref = NULL,
  git_path_dir = NULL,
  environment_variables = NULL,
  notifications = NULL,
  partition_label = NULL
)
```

Arguments

id	integer required. The ID for this Service.
name	string optional. The name of this Service.
description	string optional. The description of this Service.
docker_image_name	string optional. The name of the docker image to pull from DockerHub.
docker_image_tag	string optional. The tag of the docker image to pull from DockerHub (default: latest).
schedule	list optional. A list containing the following elements: • runtimePlan string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will

be on when scheduled. Always On means the deployed service will always be on.
- recurrences array, List of day-hour combinations this item is scheduled for

`replicas` integer optional. The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.

`max_replicas` integer optional. The maximum number of Service replicas to deploy. Defining this field enables autoscaling.

`instance_type` string optional. The EC2 instance type to deploy to.

`memory` integer optional. The amount of memory allocated to each replica of the Service.

`cpu` integer optional. The amount of cpu allocated to each replica of the the Service.

`credentials` array optional. The credentials attached to the service. Accepts a list of credential IDs and returns a list of id, name pairs.

`permission_set_id` integer optional. The ID of the associated permission set, if any.

`git_repo_url` string optional. The url for the git repo where the Service code lives.

`git_repo_ref` string optional. The git reference to use when pulling code from the repo.

`git_path_dir` string optional. The path to the Service code within the git repo. If unspecified, the root directory will be used.

`environment_variables` list optional. Environment Variables to be passed into the Service.

`notifications` list optional. A list containing the following elements:
- failureEmailAddresses array, Addresses to notify by e-mail when the service fails.
- failureOn boolean, If failure email notifications are on

`partition_label` string optional. The partition label used to run this object.

Value

A list containing the following elements:

`id` integer, The ID for this Service.

`name` string, The name of this Service.

`description` string, The description of this Service.

`user` list, A list containing the following elements:
- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`type` string, The type of this Service

`dockerImageName` string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).

`schedule` list, A list containing the following elements:

- runtimePlan string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on.
- recurrences array, List of day-hour combinations this item is scheduled for

`timeZone` string,

`replicas` integer, The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.

`maxReplicas` integer, The maximum number of Service replicas to deploy. Defining this field enables autoscaling.

`instanceType` string, The EC2 instance type to deploy to.

`memory` integer, The amount of memory allocated to each replica of the Service.

`cpu` integer, The amount of cpu allocated to each replica of the the Service.

`createdAt` string,

`updatedAt` string,

`credentials` array, The credentials attached to the service. Accepts a list of credential IDs and returns a list of id, name pairs.

`permissionSetId` integer, The ID of the associated permission set, if any.

`gitRepoUrl` string, The url for the git repo where the Service code lives.

`gitRepoRef` string, The git reference to use when pulling code from the repo.

`gitPathDir` string, The path to the Service code within the git repo. If unspecified, the root directory will be used.

`reportId` integer, The ID of the associated report.

`currentDeployment` list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.

- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- serviceId integer, The ID of owning Service

`currentUrl` string, The URL that the service is hosted at.

`environmentVariables` list, Environment Variables to be passed into the Service.

`notifications` list, A list containing the following elements:

- failureEmailAddresses array, Addresses to notify by e-mail when the service fails.
- failureOn boolean, If failure email notifications are on

`partitionLabel` string, The partition label used to run this object.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`hidden` boolean, The hidden status of the item.

`services_put_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
services_put_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id	integer, The ID for this Service.
name	string, The name of this Service.
description	string, The description of this Service.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
type	string, The type of this Service
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
schedule	list, A list containing the following elements: • runtimePlan string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on. • recurrences array, List of day-hour combinations this item is scheduled for
timeZone	string,
replicas	integer, The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.
maxReplicas	integer, The maximum number of Service replicas to deploy. Defining this field enables autoscaling.
instanceType	string, The EC2 instance type to deploy to.
memory	integer, The amount of memory allocated to each replica of the Service.
cpu	integer, The amount of cpu allocated to each replica of the the Service.
createdAt	string,
updatedAt	string,
credentials	array, The credentials attached to the service. Accepts a list of credential IDs and returns a list of id, name pairs.
permissionSetId	integer, The ID of the associated permission set, if any.
gitRepoUrl	string, The url for the git repo where the Service code lives.
gitRepoRef	string, The git reference to use when pulling code from the repo.
gitPathDir	string, The path to the Service code within the git repo. If unspecified, the root directory will be used.

`reportId` integer, The ID of the associated report.

`currentDeployment`
list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- serviceId integer, The ID of owning Service

`currentUrl` string, The URL that the service is hosted at.

`environmentVariables`
list, Environment Variables to be passed into the Service.

`notifications` list, A list containing the following elements:

- failureEmailAddresses array, Addresses to notify by e-mail when the service fails.
- failureOn boolean, If failure email notifications are on

`partitionLabel` string, The partition label used to run this object.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`hidden` boolean, The hidden status of the item.

services_put_projects *Add a Service to a project*

Description

Add a Service to a project

Usage

```
services_put_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Service.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

services_put_shares_groups *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
services_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

services_put_shares_users

Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

```
services_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

services_put_transfer *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
services_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. ID of target user
include_dependencies	boolean required. Whether or not to give manage permissions on all dependencies
email_body	string optional. Custom body text for e-mail sent on transfer.
send_email	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies — array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

sql *SQL escaping*

Description

SQL escaping

Usage

```
sql(...)
```

Arguments

...	Character vectors that will be combined into a single SQL expression.

stash_local_dataframe *Stash a data frame in feather or csv format, depending on CivisML version.*

Description

Stash a data frame in feather or csv format, depending on CivisML version.

Usage

```
stash_local_dataframe(x, tmpl_id)
```

Arguments

x	data.frame to stash
tmpl_id	CivisML training template id

Value

file id where dataframe is stored

storage_hosts_delete *Delete a storage host (deprecated)*

Description

Delete a storage host (deprecated)

Usage

```
storage_hosts_delete(id)
```

Arguments

id	integer required. The ID of the storage host.

Value

An empty HTTP response

`storage_hosts_delete_shares_groups`
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
storage_hosts_delete_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

`storage_hosts_delete_shares_users`
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
storage_hosts_delete_shares_users(id, user_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. The ID of the user.

Value

An empty HTTP response

storage_hosts_get *Get a storage host*

Description

Get a storage host

Usage

```
storage_hosts_get(id)
```

Arguments

id	integer required. The ID of the storage host.

Value

A list containing the following elements:

id	integer, The ID of the storage host.
owner	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The human readable name for the storage host.
provider	string, The storage provider.One of: s3.
bucket	string, The bucket for this storage host.
s3Options	list, A list containing the following elements: • region string, The region for this storage host (ex. "us-east-1")

storage_hosts_list *List the storage hosts*

Description

List the storage hosts

Usage

```
storage_hosts_list()
```

Value

An array containing the following fields:

id	integer, The ID of the storage host.
owner	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The human readable name for the storage host.
provider	string, The storage provider.One of: s3.
bucket	string, The bucket for this storage host.
s3Options	list, A list containing the following elements: • region string, The region for this storage host (ex. "us-east-1")

---

storage_hosts_list_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
storage_hosts_list_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object

`permissionLevel`
: string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).

`description`
: string, Additional information about the dependency, if relevant

`shareable`
: boolean, Whether or not the requesting user can share this object.

`storage_hosts_list_shares`
List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
storage_hosts_list_shares(id)
```

Arguments

`id`
: integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers`
: list, A list containing the following elements:

- users array,
- groups array,

`writers`
: list, A list containing the following elements:

- users array,
- groups array,

`owners`
: list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`storage_hosts_patch` *Update some attributes of this storage host*

Description

Update some attributes of this storage host

Usage

```
storage_hosts_patch(
  id,
  name = NULL,
  provider = NULL,
  bucket = NULL,
  s3_options = NULL
)
```

Arguments

`id`	integer required. The ID of the storage host.
`name`	string optional. The human readable name for the storage host.
`provider`	string optional. The storage provider.One of: s3.
`bucket`	string optional. The bucket for this storage host.
`s3_options`	list optional. A list containing the following elements: • region string, The region for this storage host (ex. "us-east-1")

Value

A list containing the following elements:

`id`	integer, The ID of the storage host.
`owner`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The human readable name for the storage host.
`provider`	string, The storage provider.One of: s3.
`bucket`	string, The bucket for this storage host.
`s3Options`	list, A list containing the following elements: • region string, The region for this storage host (ex. "us-east-1")

`storage_hosts_post` *Create a new storage host*

Description

Create a new storage host

Usage

```
storage_hosts_post(provider, bucket, name, s3_options = NULL)
```

Arguments

`provider`	string required. The storage provider.One of: s3.
`bucket`	string required. The bucket for this storage host.
`name`	string required. The human readable name for the storage host.
`s3_options`	list optional. A list containing the following elements: • region string, The region for this storage host (ex. "us-east-1")

Value

A list containing the following elements:

`id`	integer, The ID of the storage host.
`owner`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The human readable name for the storage host.
`provider`	string, The storage provider.One of: s3.
`bucket`	string, The bucket for this storage host.
`s3Options`	list, A list containing the following elements: • region string, The region for this storage host (ex. "us-east-1")

`storage_hosts_put` *Replace all attributes of this storage host*

Description

Replace all attributes of this storage host

Usage

```
storage_hosts_put(id, name, provider, bucket, s3_options = NULL)
```

Arguments

`id`	integer required. The ID of the storage host.
`name`	string required. The human readable name for the storage host.
`provider`	string required. The storage provider.One of: s3.
`bucket`	string required. The bucket for this storage host.
`s3_options`	list optional. A list containing the following elements: • region string, The region for this storage host (ex. "us-east-1")

Value

A list containing the following elements:

`id`	integer, The ID of the storage host.
`owner`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The human readable name for the storage host.
`provider`	string, The storage provider.One of: s3.
`bucket`	string, The bucket for this storage host.
`s3Options`	list, A list containing the following elements: • region string, The region for this storage host (ex. "us-east-1")

`storage_hosts_put_shares_groups` *Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```
storage_hosts_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_ids`	array required. An array of one or more group IDs.
`permission_level`	string required. Options are: "read", "write", or "manage".
`share_email_body`	string optional. Custom body text for e-mail sent on a share.
`send_shared_email`	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

`readers`	list, A list containing the following elements: • users array, • groups array,
`writers`	list, A list containing the following elements: • users array, • groups array,
`owners`	list, A list containing the following elements: • users array, • groups array,
`totalUserShares`	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

storage_hosts_put_shares_users
Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

```
storage_hosts_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers
: list, A list containing the following elements:
- users array,
- groups array,

writers
: list, A list containing the following elements:
- users array,
- groups array,

owners
: list, A list containing the following elements:
- users array,

- groups array,

`totalUserShares`
: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`storage_hosts_put_transfer` *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
storage_hosts_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. ID of target user
`include_dependencies`	boolean required. Whether or not to give manage permissions on all dependencies
`email_body`	string optional. Custom body text for e-mail sent on transfer.
`send_email`	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies` array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object

- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`studios_delete_deployments` *Delete a Studio deployment*

Description

Delete a Studio deployment

Usage

```
studios_delete_deployments(studio_id, deployment_id)
```

Arguments

`studio_id`	integer required. The ID of the owning Studio
`deployment_id`	integer required. The ID for this deployment

Value

An empty HTTP response

`studios_get` *Get a Studio*

Description

Get a Studio

Usage

```
studios_get(id)
```

Arguments

`id`	integer required.

Value

A list containing the following elements:

`id` integer, The ID of the studio.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of the studio.

`dockerImageName` string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).

`instanceType` string, The EC2 instance type to deploy to.

`requiredResources` list, A list containing the following elements:

- memory integer, The amount of memory allocated to the studio.
- diskSpace number, The amount of disk space, in GB, to allocate for the studio. Fractional values (e.g. 0.25) are supported.
- cpu integer, The amount of cpu allocated to the studio.

`createdAt` string,

`updatedAt` string,

`mostRecentDeployment` list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.

- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- studioId integer, The ID of the owning Studio

`gitCredentialId` integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- description string, A short sentence or fragment describing this parameter.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- value string, The value you would like to set this param to.

`gitRepoId` integer, The ID of the git repository.

`gitRepoUrl` string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`gitRef` string, The git reference if git repo is specified

`partitionLabel` string, The partition label used to run this object.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`studios_get_deployments` *Get details about a Studio deployment*

Description

Get details about a Studio deployment

Usage

```
studios_get_deployments(studio_id, deployment_id)
```

Arguments

`studio_id`	integer required. The ID of the owning Studio
`deployment_id`	integer required. The ID for this deployment

Value

A list containing the following elements:

deploymentId	integer, The ID for this deployment.
userId	integer, The ID of the owner.
host	string, Domain of the deployment.
name	string, Name of the deployment.
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
displayUrl	string, A signed URL for viewing the deployed item.
instanceType	string, The EC2 instance type requested for the deployment.
memory	integer, The memory allocated to the deployment, in MB.
cpu	integer, The cpu allocated to the deployment, in millicores.
state	string, The state of the deployment.
stateMessage	string, A detailed description of the state.
maxMemoryUsage	number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
maxCpuUsage	number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
createdAt	string,
updatedAt	string,
studioId	integer, The ID of the owning Studio

---

studios_list_deployments

List deployments for a Studio

Description

List deployments for a Studio

Usage

```
studios_list_deployments(
  studio_id,
  deployment_id = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

studio_id	integer required. The ID of the owning Studio
deployment_id	integer optional. The ID for this deployment
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

deploymentId	integer, The ID for this deployment.
userId	integer, The ID of the owner.
host	string, Domain of the deployment.
name	string, Name of the deployment.
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
instanceType	string, The EC2 instance type requested for the deployment.
memory	integer, The memory allocated to the deployment, in MB.
cpu	integer, The cpu allocated to the deployment, in millicores.
state	string, The state of the deployment.
stateMessage	string, A detailed description of the state.
maxMemoryUsage	number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
maxCpuUsage	number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
createdAt	string,
updatedAt	string,
studioId	integer, The ID of the owning Studio

studios_list_deployments_logs
Get the logs for a Studio deployment

Description

Get the logs for a Studio deployment

Usage

```
studios_list_deployments_logs(
  id,
  deployment_id,
  start_at = NULL,
  end_at = NULL,
  limit = NULL
)
```

Arguments

id	integer required. The ID of the owning Studio.
deployment_id	integer required. The ID for this deployment.
start_at	string optional. Log entries with a lower timestamp will be omitted.
end_at	string optional. Log entries with a higher timestamp will be omitted.
limit	integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

message	string, The log message.
stream	string, The stream of the log. One of "stdout", "stderr".
createdAt	string, The time the log was created.
source	string, The source of the log. One of "system", "user".

studios_patch *Update some attributes of this Studio*

Description

Update some attributes of this Studio

Usage

```
studios_patch(
  id,
  name = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  instance_type = NULL,
  required_resources = NULL,
  git_credential_id = NULL,
  params = NULL,
  git_repo_url = NULL,
  git_ref = NULL,
  partition_label = NULL
)
```

Arguments

id	integer required. The ID of the studio.
name	string optional. The name of the studio.
docker_image_name	string optional. The name of the docker image to pull from DockerHub.
docker_image_tag	string optional. The tag of the docker image to pull from DockerHub (default: latest).
instance_type	string optional. The EC2 instance type to deploy to.
required_resources	list optional. A list containing the following elements: • memory integer, The amount of memory allocated to the studio. • diskSpace number, The amount of disk space, in GB, to allocate for the studio. Fractional values (e.g. 0.25) are supported. • cpu integer, The amount of cpu allocated to the studio.
git_credential_id	integer optional. The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used.
params	array optional. An array containing the following fields:

- name string, The variable's name as used within your code.
- description string, A short sentence or fragment describing this parameter.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- value string, The value you would like to set this param to.

`git_repo_url` string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`git_ref` string optional. The git reference if git repo is specified

`partition_label` string optional. The partition label used to run this object.

Value

A list containing the following elements:

`id` integer, The ID of the studio.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of the studio.

`dockerImageName` string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).

`instanceType` string, The EC2 instance type to deploy to.

`requiredResources` list, A list containing the following elements:

- memory integer, The amount of memory allocated to the studio.
- diskSpace number, The amount of disk space, in GB, to allocate for the studio. Fractional values (e.g. 0.25) are supported.
- cpu integer, The amount of cpu allocated to the studio.

`createdAt` string,

`updatedAt` string,

`mostRecentDeployment` list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.

- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- studioId integer, The ID of the owning Studio

`gitCredentialId` integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- description string, A short sentence or fragment describing this parameter.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- value string, The value you would like to set this param to.

`gitRepoId` integer, The ID of the git repository.

`gitRepoUrl` string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`gitRef` string, The git reference if git repo is specified

`partitionLabel` string, The partition label used to run this object.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`studios_post` *Create a Studio*

Description

Create a Studio

Usage

```
studios_post(
  name = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  instance_type = NULL,
  required_resources = NULL,
  git_credential_id = NULL,
  params = NULL,
  git_repo_url = NULL,
  git_ref = NULL,
  partition_label = NULL
)
```

Arguments

name	string optional. The name of the studio.
docker_image_name	string optional. The name of the docker image to pull from DockerHub.
docker_image_tag	string optional. The tag of the docker image to pull from DockerHub (default: latest).
instance_type	string optional. The EC2 instance type to deploy to.
required_resources	list optional. A list containing the following elements: • memory integer, The amount of memory allocated to the studio. • diskSpace number, The amount of disk space, in GB, to allocate for the studio. Fractional values (e.g. 0.25) are supported. • cpu integer, The amount of cpu allocated to the studio.
git_credential_id	integer optional. The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • description string, A short sentence or fragment describing this parameter. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • value string, The value you would like to set this param to.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref	string optional. The git reference if git repo is specified
partition_label	string optional. The partition label used to run this object.

Value

A list containing the following elements:

`id` integer, The ID of the studio.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of the studio.

`dockerImageName` string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).

`instanceType` string, The EC2 instance type to deploy to.

`requiredResources` list, A list containing the following elements:

- memory integer, The amount of memory allocated to the studio.
- diskSpace number, The amount of disk space, in GB, to allocate for the studio. Fractional values (e.g. 0.25) are supported.
- cpu integer, The amount of cpu allocated to the studio.

`createdAt` string,

`updatedAt` string,

`mostRecentDeployment` list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.

- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- studioId integer, The ID of the owning Studio

`gitCredentialId`
: integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used.

`params`
: array, An array containing the following fields:

- name string, The variable's name as used within your code.
- description string, A short sentence or fragment describing this parameter.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- value string, The value you would like to set this param to.

`gitRepoId`
: integer, The ID of the git repository.

`gitRepoUrl`
: string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`gitRef`
: string, The git reference if git repo is specified

`partitionLabel`
: string, The partition label used to run this object.

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`archived`
: string, The archival status of the requested item(s).

`studios_post_deployments` *Deploy a Studio*

Description

Deploy a Studio

Usage

```
studios_post_deployments(studio_id, deployment_id = NULL)
```

Arguments

`studio_id`
: integer required. The ID of the owning Studio

`deployment_id`
: integer optional. The ID for this deployment

Value

A list containing the following elements:

deploymentId	integer, The ID for this deployment.
userId	integer, The ID of the owner.
host	string, Domain of the deployment.
name	string, Name of the deployment.
dockerImageName	string, The name of the docker image to pull from DockerHub.
dockerImageTag	string, The tag of the docker image to pull from DockerHub (default: latest).
displayUrl	string, A signed URL for viewing the deployed item.
instanceType	string, The EC2 instance type requested for the deployment.
memory	integer, The memory allocated to the deployment, in MB.
cpu	integer, The cpu allocated to the deployment, in millicores.
state	string, The state of the deployment.
stateMessage	string, A detailed description of the state.
maxMemoryUsage	number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
maxCpuUsage	number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
createdAt	string,
updatedAt	string,
studioId	integer, The ID of the owning Studio

---

studios_put *Replace all attributes of this Studio*

Description

Replace all attributes of this Studio

Usage

```
studios_put(
  id,
  name = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  instance_type = NULL,
  required_resources = NULL,
  git_credential_id = NULL,
  params = NULL,
  git_repo_url = NULL,
  git_ref = NULL,
  partition_label = NULL
)
```

Arguments

id	integer required. The ID of the studio.
name	string optional. The name of the studio.
docker_image_name	string optional. The name of the docker image to pull from DockerHub.
docker_image_tag	string optional. The tag of the docker image to pull from DockerHub (default: latest).
instance_type	string optional. The EC2 instance type to deploy to.
required_resources	list optional. A list containing the following elements: • memory integer, The amount of memory allocated to the studio. • diskSpace number, The amount of disk space, in GB, to allocate for the studio. Fractional values (e.g. 0.25) are supported. • cpu integer, The amount of cpu allocated to the studio.
git_credential_id	integer optional. The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used.
params	array optional. An array containing the following fields: • name string, The variable's name as used within your code. • description string, A short sentence or fragment describing this parameter. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • value string, The value you would like to set this param to.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref	string optional. The git reference if git repo is specified
partition_label	string optional. The partition label used to run this object.

Value

A list containing the following elements:

id	integer, The ID of the studio.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of the studio.

dockerImageName
: string, The name of the docker image to pull from DockerHub.

dockerImageTag
: string, The tag of the docker image to pull from DockerHub (default: latest).

instanceType
: string, The EC2 instance type to deploy to.

requiredResources
: list, A list containing the following elements:

- memory integer, The amount of memory allocated to the studio.
- diskSpace number, The amount of disk space, in GB, to allocate for the studio. Fractional values (e.g. 0.25) are supported.
- cpu integer, The amount of cpu allocated to the studio.

createdAt
: string,

updatedAt
: string,

mostRecentDeployment
: list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- studioId integer, The ID of the owning Studio

gitCredentialId
: integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used.

params
: array, An array containing the following fields:

- name string, The variable's name as used within your code.
- description string, A short sentence or fragment describing this parameter.

- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- value string, The value you would like to set this param to.

`gitRepoId` integer, The ID of the git repository.

`gitRepoUrl` string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).

`gitRef` string, The git reference if git repo is specified

`partitionLabel` string, The partition label used to run this object.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`archived` string, The archival status of the requested item(s).

`studios_put_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
studios_put_archive(id, status)
```

Arguments

`id`	integer required. The ID of the object.
`status`	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

`id` integer, The ID of the studio.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`name` string, The name of the studio.

`dockerImageName` string, The name of the docker image to pull from DockerHub.

`dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).

`instanceType` string, The EC2 instance type to deploy to.

`requiredResources`
list, A list containing the following elements:

- memory integer, The amount of memory allocated to the studio.
- diskSpace number, The amount of disk space, in GB, to allocate for the studio. Fractional values (e.g. 0.25) are supported.
- cpu integer, The amount of cpu allocated to the studio.

`createdAt` string,

`updatedAt` string,

`mostRecentDeployment`
list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment, in MB.
- cpu integer, The cpu allocated to the deployment, in millicores.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- maxMemoryUsage number, If the deployment has finished, the maximum amount of memory used during the deployment, in MB.
- maxCpuUsage number, If the deployment has finished, the maximum amount of cpu used during the deployment, in millicores.
- createdAt string,
- updatedAt string,
- studioId integer, The ID of the owning Studio

`gitCredentialId`
integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used.

`params` array, An array containing the following fields:

- name string, The variable's name as used within your code.
- description string, A short sentence or fragment describing this parameter.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- value string, The value you would like to set this param to.

`gitRepoId` integer, The ID of the git repository.

gitRepoUrl	string, The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
gitRef	string, The git reference if git repo is specified
partitionLabel	string, The partition label used to run this object.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".
archived	string, The archival status of the requested item(s).

`tables_delete_projects` *Remove a Table from a project*

Description

Remove a Table from a project

Usage

```
tables_delete_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Table.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

`tables_delete_tags` *Add a tag to a table*

Description

Add a tag to a table

Usage

```
tables_delete_tags(id, table_tag_id)
```

Arguments

id	integer required. The ID of the table.
table_tag_id	integer required. The ID of the tag.

Value

An empty HTTP response

tables_get *Show basic table info*

Description

Show basic table info

Usage

```
tables_get(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID of the table.
databaseId	integer, The ID of the database.
schema	string, The name of the schema containing the table.
name	string, Name of the table.
description	string, The description of the table, as specified by the table owner
isView	boolean, True if this table represents a view. False if it represents a regular table.
rowCount	integer, The number of rows in the table.
columnCount	integer, The number of columns in the table.
sizeMb	number, The size of the table in megabytes.
owner	string, The database username of the table's owner.
distkey	string, The column used as the Amazon Redshift distkey.
sortkeys	string, The column used as the Amazon Redshift sortkey.
refreshStatus	string, How up-to-date the table's statistics on row counts, null counts, distinct counts, and values distributions are. One of: refreshing, stale, or current.
lastRefresh	string, The time of the last statistics refresh.
dataUpdatedAt	string, The last time that Civis Platform captured a change in this table.Only applicable for Redshift tables; please see the Civis help desk for more info.
schemaUpdatedAt	string, The last time that Civis Platform captured a change to the table attributes/structure.Only applicable for Redshift tables; please see the Civis help desk for more info.
refreshId	string, The ID of the most recent statistics refresh.
lastRun	list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`primaryKeys` array, The primary keys for this table.

`lastModifiedKeys`
array, The columns indicating an entry's modification status for this table.

`tableTags` array, An array containing the following fields:

- id integer, Table Tag ID
- name string, Table Tag Name

`ontologyMapping`
list, The ontology-key to column-name mapping. See /ontology for the list of valid ontology keys.

`columns` array, An array containing the following fields:

- name string, Name of the column.
- civisDataType string, The generic data type of the column (ex. "string"). Since this is database-agnostic, it may be helpful when loading data to R/Python.
- sqlType string, The database-specific SQL type of the column (ex. "varchar(30)").
- sampleValues array, A sample of values from the column.
- encoding string, The compression encoding for this columnSee: http://docs.aws.amazon.com/redshift
- description string, The description of the column, as specified by the table owner
- order integer, Relative position of the column in the table.
- minValue string, Smallest value in the column.
- maxValue string, Largest value in the column.
- avgValue number, This parameter is deprecated.
- stddev number, This parameter is deprecated.
- valueDistributionPercent object, A mapping between each value in the column and the percentage of rows with that value.Only present for tables with fewer than approximately 25,000,000 rows and for columns with fewer than twenty distinct values.
- coverageCount integer, Number of non-null values in the column.
- nullCount integer, Number of null values in the column.
- possibleDependentVariableTypes array, Possible dependent variable types the column may be used to model. Null if it may not be used as a dependent variable.
- useableAsIndependentVariable boolean, Whether the column may be used as an independent variable to train a model.
- useableAsPrimaryKey boolean, Whether the column may be used as an primary key to identify table rows.

- valueDistribution object, An object mapping distinct values in the column to the number of times they appear in the column
- distinctCount integer, Number of distinct values in the column. NULL values are counted and treated as a single distinct value.

`joins` array, An array containing the following fields:

- id integer,
- leftTableId integer,
- leftIdentifier string,
- rightTableId integer,
- rightIdentifier string,
- on string,
- leftJoin boolean,
- createdAt string,
- updatedAt string,

`multipartKey` array,

`enhancements` array, An array containing the following fields:

- type string,
- createdAt string,
- updatedAt string,
- joinId integer,

`viewDef` string,

`tableDef` string,

`outgoingTableMatches`
array, An array containing the following fields:

- sourceTableId integer, Source table
- targetType string, Target type
- targetId integer, Target ID
- target object,
- job object,

`tables_get_enhancements_cass_ncoa`
View the status of a CASS / NCOA table enhancement

Description

View the status of a CASS / NCOA table enhancement

Usage

```
tables_get_enhancements_cass_ncoa(id, source_table_id)
```

Arguments

id	integer required. The ID of the enhancement.
source_table_id	integer required. The ID of the table that was enhanced.

Value

A list containing the following elements:

id	integer, The ID of the enhancement.
sourceTableId	integer, The ID of the table that was enhanced.
state	string, The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
enhancedTableSchema	string, The schema name of the table created by the enhancement.
enhancedTableName	string, The name of the table created by the enhancement.
performNcoa	boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.
ncoaCredentialId	integer, Credential to use when performing NCOA updates. Required if 'performNcoa' is true.
outputLevel	string, The set of fields persisted by a CASS or NCOA enhancement.For CASS enhancements, one of 'cass' or 'all.'For NCOA enhancements, one of 'cass', 'ncoa' , 'coalesced' or 'all'.By default, all fields will be returned.
batchSize	integer, The maximum number of records processed at a time. Note that this parameter is not available to all users.

---

tables_get_enhancements_geocodings

View the status of a geocoding table enhancement

Description

View the status of a geocoding table enhancement

Usage

```
tables_get_enhancements_geocodings(id, source_table_id)
```

Arguments

id	integer required. The ID of the enhancement.
source_table_id	integer required. The ID of the table that was enhanced.

Value

A list containing the following elements:

id	integer, The ID of the enhancement.
sourceTableId	integer, The ID of the table that was enhanced.
state	string, The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
enhancedTableSchema	string, The schema name of the table created by the enhancement.
enhancedTableName	string, The name of the table created by the enhancement.

---

tables_list *List tables*

Description

List tables

Usage

```
tables_list(
  database_id = NULL,
  schema = NULL,
  name = NULL,
  search = NULL,
  table_tag_ids = NULL,
  credential_id = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

database_id	integer optional. The ID of the database.
schema	string optional. If specified, will be used to filter the tables returned. Substring matching is supported with "%" and "*" wildcards (e.g., "schema=%census%" will return both "client_census.table" and "census_2010.table").
name	string optional. If specified, will be used to filter the tables returned. Substring matching is supported with "%" and "*" wildcards (e.g., "name=%table%" will return both "table1" and "my table").
search	string optional. If specified, will be used to filter the tables returned. Will search across schema and name (in the full form schema.name) and will return any full name containing the search string.

table_tag_ids	array optional. If specified, will be used to filter the tables returned. Will search across Table Tags and will return any tables that have one of the matching Table Tags.
credential_id	integer optional. If specified, will be used instead of the default credential to filter the tables returned.
limit	integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to schema. Must be one of: schema, name, search, table_tag_ids, credential_id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id	integer, The ID of the table.
databaseId	integer, The ID of the database.
schema	string, The name of the schema containing the table.
name	string, Name of the table.
description	string, The description of the table, as specified by the table owner
isView	boolean, True if this table represents a view. False if it represents a regular table.
rowCount	integer, The number of rows in the table.
columnCount	integer, The number of columns in the table.
sizeMb	number, The size of the table in megabytes.
owner	string, The database username of the table's owner.
distkey	string, The column used as the Amazon Redshift distkey.
sortkeys	string, The column used as the Amazon Redshift sortkey.
refreshStatus	string, How up-to-date the table's statistics on row counts, null counts, distinct counts, and values distributions are. One of: refreshing, stale, or current.
lastRefresh	string, The time of the last statistics refresh.
refreshId	string, The ID of the most recent statistics refresh.
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed. • error string, The error message for this run, if present.
tableTags	array, An array containing the following fields: • id integer, Table Tag ID • name string, Table Tag Name

tables_list_columns *List columns in the specified table*

Description

List columns in the specified table

Usage

```
tables_list_columns(
  id,
  name = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	integer required.
name	string optional. Search for columns with the given name, within the specified table.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to name. Must be one of: name, order.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

name	string, Name of the column.
civisDataType	string, The generic data type of the column (ex. "string"). Since this is database-agnostic, it may be helpful when loading data to R/Python.
sqlType	string, The database-specific SQL type of the column (ex. "varchar(30)").
sampleValues	array, A sample of values from the column.
encoding	string, The compression encoding for this columnSee: http://docs.aws.amazon.com/redshift/latest/dg/c_C
description	string, The description of the column, as specified by the table owner
order	integer, Relative position of the column in the table.
minValue	string, Smallest value in the column.

maxValue string, Largest value in the column.

avgValue number, This parameter is deprecated.

stddev number, This parameter is deprecated.

valueDistributionPercent
list, A mapping between each value in the column and the percentage of rows with that value.Only present for tables with fewer than approximately 25,000,000 rows and for columns with fewer than twenty distinct values.

coverageCount integer, Number of non-null values in the column.

nullCount integer, Number of null values in the column.

possibleDependentVariableTypes
array, Possible dependent variable types the column may be used to model. Null if it may not be used as a dependent variable.

useableAsIndependentVariable
boolean, Whether the column may be used as an independent variable to train a model.

useableAsPrimaryKey
boolean, Whether the column may be used as an primary key to identify table rows.

valueDistribution
list, An object mapping distinct values in the column to the number of times they appear in the column

distinctCount integer, Number of distinct values in the column. NULL values are counted and treated as a single distinct value.

---

tables_list_projects *List the projects a Table belongs to*

Description

List the projects a Table belongs to

Usage

```
tables_list_projects(id, hidden = NULL)
```

Arguments

id integer required. The ID of the Table.

hidden boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

`id`	integer, The ID for this project.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The name of this project.
`description`	string, A description of the project.
`users`	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`autoShare`	boolean,
`createdAt`	string,
`updatedAt`	string,
`archived`	string, The archival status of the requested item(s).

`tables_patch` *Update a table*

Description

Update a table

Usage

```
tables_patch(
  id,
  ontology_mapping = NULL,
  description = NULL,
  primary_keys = NULL,
  last_modified_keys = NULL
)
```

Arguments

id	integer required. The ID of the table.
ontology_mapping	list optional. The ontology-key to column-name mapping. See /ontology for the list of valid ontology keys.
description	string optional. The user-defined description of the table.
primary_keys	array optional. A list of column(s) which together uniquely identify a row in the data.These columns must not contain NULL values.
last_modified_keys	array optional. The columns indicating when a row was last modified.

Value

A list containing the following elements:

id	integer, The ID of the table.
databaseId	integer, The ID of the database.
schema	string, The name of the schema containing the table.
name	string, Name of the table.
description	string, The description of the table, as specified by the table owner
isView	boolean, True if this table represents a view. False if it represents a regular table.
rowCount	integer, The number of rows in the table.
columnCount	integer, The number of columns in the table.
sizeMb	number, The size of the table in megabytes.
owner	string, The database username of the table's owner.
distkey	string, The column used as the Amazon Redshift distkey.
sortkeys	string, The column used as the Amazon Redshift sortkey.
refreshStatus	string, How up-to-date the table's statistics on row counts, null counts, distinct counts, and values distributions are. One of: refreshing, stale, or current.
lastRefresh	string, The time of the last statistics refresh.
dataUpdatedAt	string, The last time that Civis Platform captured a change in this table.Only applicable for Redshift tables; please see the Civis help desk for more info.
schemaUpdatedAt	string, The last time that Civis Platform captured a change to the table attributes/structure.Only applicable for Redshift tables; please see the Civis help desk for more info.
refreshId	string, The ID of the most recent statistics refresh.
lastRun	list, A list containing the following elements: • id integer, • state string, • createdAt string, The time that the run was queued. • startedAt string, The time that the run started.

- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`primaryKeys` array, The primary keys for this table.

`lastModifiedKeys`
array, The columns indicating an entry's modification status for this table.

`tableTags` array, An array containing the following fields:

- id integer, Table Tag ID
- name string, Table Tag Name

`ontologyMapping`
list, The ontology-key to column-name mapping. See /ontology for the list of valid ontology keys.

`tables_post_enhancements_cass_ncoa`
Standardize addresses in a table

Description

Standardize addresses in a table

Usage

```
tables_post_enhancements_cass_ncoa(
  source_table_id,
  perform_ncoa = NULL,
  ncoa_credential_id = NULL,
  output_level = NULL,
  batch_size = NULL
)
```

Arguments

`source_table_id`
integer required. The ID of the table to be enhanced.

`perform_ncoa` boolean optional. Whether to update addresses for records matching the National Change of Address (NCOA) database.

`ncoa_credential_id`
integer optional. Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

`output_level` string optional. The set of fields persisted by a CASS or NCOA enhancement.For CASS enhancements, one of 'cass' or 'all.'For NCOA enhancements, one of 'cass', 'ncoa' , 'coalesced' or 'all'.By default, all fields will be returned.

`batch_size` integer optional. The maximum number of records processed at a time. Note that this parameter is not available to all users.

Value

A list containing the following elements:

id	integer, The ID of the enhancement.
sourceTableId	integer, The ID of the table that was enhanced.
state	string, The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
enhancedTableSchema	string, The schema name of the table created by the enhancement.
enhancedTableName	string, The name of the table created by the enhancement.
performNcoa	boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.
ncoaCredentialId	integer, Credential to use when performing NCOA updates. Required if 'performNcoa' is true.
outputLevel	string, The set of fields persisted by a CASS or NCOA enhancement.For CASS enhancements, one of 'cass' or 'all.'For NCOA enhancements, one of 'cass', 'ncoa' , 'coalesced' or 'all'.By default, all fields will be returned.
batchSize	integer, The maximum number of records processed at a time. Note that this parameter is not available to all users.

---

tables_post_enhancements_geocodings
Geocode a table

Description

Geocode a table

Usage

```
tables_post_enhancements_geocodings(source_table_id)
```

Arguments

source_table_id	integer required. The ID of the table to be enhanced.

Value

A list containing the following elements:

id	integer, The ID of the enhancement.
sourceTableId	integer, The ID of the table that was enhanced.
state	string, The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
enhancedTableSchema	string, The schema name of the table created by the enhancement.
enhancedTableName	string, The name of the table created by the enhancement.

---

tables_post_refresh *Request a refresh for column and table statistics*

Description

Request a refresh for column and table statistics

Usage

```
tables_post_refresh(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID of the table.
databaseId	integer, The ID of the database.
schema	string, The name of the schema containing the table.
name	string, Name of the table.
description	string, The description of the table, as specified by the table owner
isView	boolean, True if this table represents a view. False if it represents a regular table.
rowCount	integer, The number of rows in the table.
columnCount	integer, The number of columns in the table.
sizeMb	number, The size of the table in megabytes.
owner	string, The database username of the table's owner.
distkey	string, The column used as the Amazon Redshift distkey.
sortkeys	string, The column used as the Amazon Redshift sortkey.

`refreshStatus` string, How up-to-date the table's statistics on row counts, null counts, distinct counts, and values distributions are. One of: refreshing, stale, or current.

`lastRefresh` string, The time of the last statistics refresh.

`dataUpdatedAt` string, The last time that Civis Platform captured a change in this table.Only applicable for Redshift tables; please see the Civis help desk for more info.

`schemaUpdatedAt` string, The last time that Civis Platform captured a change to the table attributes/structure.Only applicable for Redshift tables; please see the Civis help desk for more info.

`refreshId` string, The ID of the most recent statistics refresh.

`lastRun` list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

`primaryKeys` array, The primary keys for this table.

`lastModifiedKeys` array, The columns indicating an entry's modification status for this table.

`tableTags` array, An array containing the following fields:

- id integer, Table Tag ID
- name string, Table Tag Name

`ontologyMapping` list, The ontology-key to column-name mapping. See /ontology for the list of valid ontology keys.

`columns` array, An array containing the following fields:

- name string, Name of the column.
- civisDataType string, The generic data type of the column (ex. "string"). Since this is database-agnostic, it may be helpful when loading data to R/Python.
- sqlType string, The database-specific SQL type of the column (ex. "varchar(30)").
- sampleValues array, A sample of values from the column.
- encoding string, The compression encoding for this columnSee: http://docs.aws.amazon.com/redshift
- description string, The description of the column, as specified by the table owner
- order integer, Relative position of the column in the table.
- minValue string, Smallest value in the column.
- maxValue string, Largest value in the column.
- avgValue number, This parameter is deprecated.
- stddev number, This parameter is deprecated.

- valueDistributionPercent object, A mapping between each value in the column and the percentage of rows with that value.Only present for tables with fewer than approximately 25,000,000 rows and for columns with fewer than twenty distinct values.
- coverageCount integer, Number of non-null values in the column.
- nullCount integer, Number of null values in the column.
- possibleDependentVariableTypes array, Possible dependent variable types the column may be used to model. Null if it may not be used as a dependent variable.
- useableAsIndependentVariable boolean, Whether the column may be used as an independent variable to train a model.
- useableAsPrimaryKey boolean, Whether the column may be used as an primary key to identify table rows.
- valueDistribution object, An object mapping distinct values in the column to the number of times they appear in the column
- distinctCount integer, Number of distinct values in the column. NULL values are counted and treated as a single distinct value.

`joins` array, An array containing the following fields:

- id integer,
- leftTableId integer,
- leftIdentifier string,
- rightTableId integer,
- rightIdentifier string,
- on string,
- leftJoin boolean,
- createdAt string,
- updatedAt string,

`multipartKey` array,

`enhancements` array, An array containing the following fields:

- type string,
- createdAt string,
- updatedAt string,
- joinId integer,

`viewDef` string,

`tableDef` string,

`outgoingTableMatches`
array, An array containing the following fields:

- sourceTableId integer, Source table
- targetType string, Target type
- targetId integer, Target ID
- target object,
- job object,

tables_post_scan *Creates and enqueues a single table scanner job on a new table*

Description

Creates and enqueues a single table scanner job on a new table

Usage

```
tables_post_scan(database_id, schema, table_name, stats_priority = NULL)
```

Arguments

database_id	integer required. The ID of the database.
schema	string required. The name of the schema containing the table.
table_name	string required. The name of the table.
stats_priority	string optional. When to sync table statistics. Valid Options are the following. Option: 'flag' means to flag stats for the next scheduled run of a full table scan on the database. Option: 'block' means to block this job on stats syncing. Option: 'queue' means to queue a separate job for syncing stats and do not block this job on the queued job. Defaults to 'flag'

Value

A list containing the following elements:

jobId	integer, The ID of the job created.
runId	integer, The ID of the run created.

tables_put_projects *Add a Table to a project*

Description

Add a Table to a project

Usage

```
tables_put_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Table.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

---

tables_put_tags *Add a tag to a table*

Description

Add a tag to a table

Usage

```
tables_put_tags(id, table_tag_id)
```

Arguments

id	integer required. The ID of the table.
table_tag_id	integer required. The ID of the tag.

Value

A list containing the following elements:

id	integer, The ID of the table.
tableTagId	integer, The ID of the tag.

---

table_tags_delete *Delete a Table Tag*

Description

Delete a Table Tag

Usage

```
table_tags_delete(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

table_tags_get *Get a Table Tag*

Description

Get a Table Tag

Usage

```
table_tags_get(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, Table Tag ID
name	string, Table Tag Name
createdAt	string, The date the tag was created.
updatedAt	string, The date the tag was recently updated on.
tableCount	integer, The total number of tables associated with the tag.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.

table_tags_list *List Table Tags*

Description

List Table Tags

Usage

```
table_tags_list(
  name = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

name	string optional. Name of the tag. If it is provided, the results will be filtered by name
limit	integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to name. Must be one of: name, user, table_count.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id	integer, Table Tag ID
name	string, Table Tag Name
tableCount	integer, The total number of tables associated with the tag.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.

---

table_tags_post *Create a Table Tag*

Description

Create a Table Tag

Usage

```
table_tags_post(name)
```

Arguments

name	string required. Table Tag Name

Value

A list containing the following elements:

id	integer, Table Tag ID
name	string, Table Tag Name
createdAt	string, The date the tag was created.
updatedAt	string, The date the tag was recently updated on.
tableCount	integer, The total number of tables associated with the tag.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.

---

templates_delete_reports

Archive a Report Template (deprecated, use archiving endpoints instead)

Description

Archive a Report Template (deprecated, use archiving endpoints instead)

Usage

```
templates_delete_reports(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

templates_delete_reports_shares_groups
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
templates_delete_reports_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

---

templates_delete_reports_shares_users
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
templates_delete_reports_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

templates_delete_scripts *Archive a Script Template (deprecated, use archiving endpoints instead)*

Description

Archive a Script Template (deprecated, use archiving endpoints instead)

Usage

```
templates_delete_scripts(id)
```

Arguments

id	integer required.

Value

An empty HTTP response

templates_delete_scripts_projects *Remove a Script Template from a project*

Description

Remove a Script Template from a project

Usage

```
templates_delete_scripts_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Script Template.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

templates_delete_scripts_shares_groups
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
templates_delete_scripts_shares_groups(id, group_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_id	integer required. The ID of the group.

Value

An empty HTTP response

templates_delete_scripts_shares_users
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
templates_delete_scripts_shares_users(id, user_id)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. The ID of the user.

Value

An empty HTTP response

templates_get_reports *Get a Report Template*

Description

Get a Report Template

Usage

```
templates_get_reports(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer,
name	string, The name of the template.
category	string, The category of this report template. Can be left blank. Acceptable values are: dataset-viz
createdAt	string,
updatedAt	string,
useCount	integer, The number of uses of this template.
archived	boolean, Whether the template has been archived.
techReviewed	boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
authCodeUrl	string, A URL to the template's stored code body.
provideAPIKey	boolean, Whether reports based on this template request an API Key from the report viewer.
hidden	boolean, The hidden status of the item.

`templates_get_scripts` *Get a Script Template*

Description

Get a Script Template

Usage

```
templates_get_scripts(id)
```

Arguments

`id`	integer required.

Value

A list containing the following elements:

`id`	integer,
`public`	boolean, If the template is public or not.
`scriptId`	integer, The id of the script that this template uses.
`scriptType`	string, The type of the template's backing script (e.g SQL, Container, Python, R, JavaScript)
`userContext`	string, The user context of the script that this template uses.
`params`	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user. • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom • required boolean, Whether this param is required. • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param. • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type. • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
`name`	string, The name of the template.

category	string, The category of this template.
note	string, A note describing what this template is used for; custom scripts created off this template will display this description.
createdAt	string,
updatedAt	string,
useCount	integer, The number of uses of this template.
uiReportId	integer, The id of the report that this template uses.
techReviewed	boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
archived	boolean, Whether the template has been archived.
hidden	boolean, The hidden status of the item.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
myPermissionLevel	string, Your permission level on the object. One of "read", "write", or "manage".

`templates_list_reports` *List Report Templates*

Description

List Report Templates

Usage

```
templates_list_reports(
  hidden = NULL,
  author = NULL,
  category = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
author	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.
category	string optional. A category to filter results by, one of: dataset-viz
limit	integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to name. Must be one of: name, updated_at, created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id	integer,
name	string, The name of the template.
category	string, The category of this report template. Can be left blank. Acceptable values are: dataset-viz
createdAt	string,
updatedAt	string,
useCount	integer, The number of uses of this template.
archived	boolean, Whether the template has been archived.
techReviewed	boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
author	list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

templates_list_reports_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
templates_list_reports_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

templates_list_reports_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
templates_list_reports_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

templates_list_scripts

List Script Templates

Description

List Script Templates

Usage

```
templates_list_scripts(
  hidden = NULL,
  author = NULL,
  category = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
author	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.
category	string optional. A category to filter results by, one of: import, export, enhancement, model, and script
limit	integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to name. Must be one of: name, updated_at, created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id	integer,
public	boolean, If the template is public or not.
scriptId	integer, The id of the script that this template uses.
userContext	string, The user context of the script that this template uses.
name	string, The name of the template.
category	string, The category of this template.
createdAt	string,
updatedAt	string,
useCount	integer, The number of uses of this template.
uiReportId	integer, The id of the report that this template uses.
techReviewed	boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
archived	boolean, Whether the template has been archived.
author	list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

templates_list_scripts_dependencies

List dependent objects for this object

Description

List dependent objects for this object

Usage

```
templates_list_scripts_dependencies(id, user_id = NULL)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer optional. ID of target user

Value

An array containing the following fields:

objectType	string, Dependent object type
fcoType	string, Human readable dependent object type
id	integer, Dependent object ID
name	string, Dependent object name, or nil if the requesting user cannot read this object
permissionLevel	string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
description	string, Additional information about the dependency, if relevant
shareable	boolean, Whether or not the requesting user can share this object.

templates_list_scripts_projects

List the projects a Script Template belongs to

Description

List the projects a Script Template belongs to

Usage

```
templates_list_scripts_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the Script Template.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id	integer, The ID for this project.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
name	string, The name of this project.
description	string, A description of the project.
users	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
autoShare	boolean,
createdAt	string,
updatedAt	string,
archived	string, The archival status of the requested item(s).

---

templates_list_scripts_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

```
templates_list_scripts_shares(id)
```

Arguments

id	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

templates_patch_reports

Update some attributes of this Report Template

Description

Update some attributes of this Report Template

Usage

```
templates_patch_reports(
  id,
  name = NULL,
  category = NULL,
  archived = NULL,
  code_body = NULL,
  provide_api_key = NULL
)
```

Arguments

id	integer required.
name	string optional. The name of the template.
category	string optional. The category of this report template. Can be left blank. Acceptable values are: dataset-viz
archived	boolean optional. Whether the template has been archived.
code_body	string optional. The code for the Template body.
provide_api_key	boolean optional. Whether reports based on this template request an API Key from the report viewer.

Value

A list containing the following elements:

id	integer,
name	string, The name of the template.
category	string, The category of this report template. Can be left blank. Acceptable values are: dataset-viz
createdAt	string,
updatedAt	string,
useCount	integer, The number of uses of this template.
archived	boolean, Whether the template has been archived.
techReviewed	boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
authCodeUrl	string, A URL to the template's stored code body.
provideAPIKey	boolean, Whether reports based on this template request an API Key from the report viewer.
hidden	boolean, The hidden status of the item.

`templates_patch_scripts` *Update some attributes of this Script Template*

Description

Update some attributes of this Script Template

Usage

```
templates_patch_scripts(
  id,
  name = NULL,
  note = NULL,
  ui_report_id = NULL,
  archived = NULL
)
```

Arguments

`id`	integer required.
`name`	string optional. The name of the template.
`note`	string optional. A note describing what this template is used for; custom scripts created off this template will display this description.
`ui_report_id`	integer optional. The id of the report that this template uses.
`archived`	boolean optional. Whether the template has been archived.

Value

A list containing the following elements:

`id`	integer,
`public`	boolean, If the template is public or not.
`scriptId`	integer, The id of the script that this template uses.
`scriptType`	string, The type of the template's backing script (e.g SQL, Container, Python, R, JavaScript)
`userContext`	string, The user context of the script that this template uses.
`params`	array, An array containing the following fields: • name string, The variable's name as used within your code. • label string, The label to present to users when asking them for the value. • description string, A short sentence or fragment describing this parameter to the end user.

- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`name` string, The name of the template.

`category` string, The category of this template.

`note` string, A note describing what this template is used for; custom scripts created off this template will display this description.

`createdAt` string,

`updatedAt` string,

`useCount` integer, The number of uses of this template.

`uiReportId` integer, The id of the report that this template uses.

`techReviewed` boolean, Whether this template has been audited by Civis for security vulnerability and correctness.

`archived` boolean, Whether the template has been archived.

`hidden` boolean, The hidden status of the item.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`templates_post_reports` *Create a Report Template*

Description

Create a Report Template

Usage

```
templates_post_reports(
  name,
  code_body,
  category = NULL,
  archived = NULL,
  provide_api_key = NULL,
  hidden = NULL
)
```

Arguments

name	string required. The name of the template.
code_body	string required. The code for the Template body.
category	string optional. The category of this report template. Can be left blank. Acceptable values are: dataset-viz
archived	boolean optional. Whether the template has been archived.
provide_api_key	boolean optional. Whether reports based on this template request an API Key from the report viewer.
hidden	boolean optional. The hidden status of the item.

Value

A list containing the following elements:

id	integer,
name	string, The name of the template.
category	string, The category of this report template. Can be left blank. Acceptable values are: dataset-viz
createdAt	string,
updatedAt	string,
useCount	integer, The number of uses of this template.
archived	boolean, Whether the template has been archived.
techReviewed	boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
authCodeUrl	string, A URL to the template's stored code body.

provideAPIKey	boolean, Whether reports based on this template request an API Key from the report viewer.
hidden	boolean, The hidden status of the item.

`templates_post_scripts` *Create a Script Template*

Description

Create a Script Template

Usage

```
templates_post_scripts(
  script_id,
  name,
  note = NULL,
  ui_report_id = NULL,
  archived = NULL,
  hidden = NULL
)
```

Arguments

script_id	integer required. The id of the script that this template uses.
name	string required. The name of the template.
note	string optional. A note describing what this template is used for; custom scripts created off this template will display this description.
ui_report_id	integer optional. The id of the report that this template uses.
archived	boolean optional. Whether the template has been archived.
hidden	boolean optional. The hidden status of the item.

Value

A list containing the following elements:

id	integer,
public	boolean, If the template is public or not.
scriptId	integer, The id of the script that this template uses.
scriptType	string, The type of the template's backing script (e.g SQL, Container, Python, R, JavaScript)
userContext	string, The user context of the script that this template uses.
params	array, An array containing the following fields:

- name string, The variable's name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'

`name` string, The name of the template.

`category` string, The category of this template.

`note` string, A note describing what this template is used for; custom scripts created off this template will display this description.

`createdAt` string,

`updatedAt` string,

`useCount` integer, The number of uses of this template.

`uiReportId` integer, The id of the report that this template uses.

`techReviewed` boolean, Whether this template has been audited by Civis for security vulnerability and correctness.

`archived` boolean, Whether the template has been archived.

`hidden` boolean, The hidden status of the item.

`author` list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

templates_put_reports *Replace all attributes of this Report Template*

Description

Replace all attributes of this Report Template

Usage

```
templates_put_reports(
  id,
  name,
  code_body,
  category = NULL,
  archived = NULL,
  provide_api_key = NULL
)
```

Arguments

id	integer required.
name	string required. The name of the template.
code_body	string required. The code for the Template body.
category	string optional. The category of this report template. Can be left blank. Acceptable values are: dataset-viz
archived	boolean optional. Whether the template has been archived.
provide_api_key	boolean optional. Whether reports based on this template request an API Key from the report viewer.

Value

A list containing the following elements:

id	integer,
name	string, The name of the template.
category	string, The category of this report template. Can be left blank. Acceptable values are: dataset-viz
createdAt	string,
updatedAt	string,
useCount	integer, The number of uses of this template.
archived	boolean, Whether the template has been archived.
techReviewed	boolean, Whether this template has been audited by Civis for security vulnerability and correctness.

author	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
authCodeUrl	string, A URL to the template's stored code body.
provideAPIKey	boolean, Whether reports based on this template request an API Key from the report viewer.
hidden	boolean, The hidden status of the item.

`templates_put_reports_shares_groups`
Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
templates_put_reports_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- `readers` list, A list containing the following elements:
  - users array,
  - groups array,
- `writers` list, A list containing the following elements:
  - users array,
  - groups array,
- `owners` list, A list containing the following elements:
  - users array,
  - groups array,
- `totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- `totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`templates_put_reports_shares_users` *Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```
templates_put_reports_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

- `id` integer required. The ID of the resource that is shared.
- `user_ids` array required. An array of one or more user IDs.
- `permission_level` string required. Options are: "read", "write", or "manage".
- `share_email_body` string optional. Custom body text for e-mail sent on a share.
- `send_shared_email` boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- `readers` list, A list containing the following elements:
  - users array,
  - groups array,
- `writers` list, A list containing the following elements:
  - users array,
  - groups array,
- `owners` list, A list containing the following elements:
  - users array,
  - groups array,
- `totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- `totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

`templates_put_reports_transfer` *Transfer ownership of this object to another user*

Description

Transfer ownership of this object to another user

Usage

```
templates_put_reports_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

- `id` integer required. The ID of the resource that is shared.
- `user_id` integer required. ID of target user
- `include_dependencies` boolean required. Whether or not to give manage permissions on all dependencies
- `email_body` string optional. Custom body text for e-mail sent on transfer.
- `send_email` boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

`dependencies` array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

`templates_put_scripts` *Replace all attributes of this Script Template*

Description

Replace all attributes of this Script Template

Usage

```
templates_put_scripts(
  id,
  name,
  note = NULL,
  ui_report_id = NULL,
  archived = NULL
)
```

Arguments

`id`	integer required.
`name`	string required. The name of the template.
`note`	string optional. A note describing what this template is used for; custom scripts created off this template will display this description.
`ui_report_id`	integer optional. The id of the report that this template uses.
`archived`	boolean optional. Whether the template has been archived.

Value

A list containing the following elements:

- `id` integer,
- `public` boolean, If the template is public or not.
- `scriptId` integer, The id of the script that this template uses.
- `scriptType` string, The type of the template's backing script (e.g SQL, Container, Python, R, JavaScript)
- `userContext` string, The user context of the script that this template uses.
- `params` array, An array containing the following fields:
  - name string, The variable's name as used within your code.
  - label string, The label to present to users when asking them for the value.
  - description string, A short sentence or fragment describing this parameter to the end user.
  - type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - required boolean, Whether this param is required.
  - value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
  - allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: '{label: 'Import', 'value': 'import'}'
- `name` string, The name of the template.
- `category` string, The category of this template.
- `note` string, A note describing what this template is used for; custom scripts created off this template will display this description.
- `createdAt` string,
- `updatedAt` string,
- `useCount` integer, The number of uses of this template.
- `uiReportId` integer, The id of the report that this template uses.
- `techReviewed` boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
- `archived` boolean, Whether the template has been archived.
- `hidden` boolean, The hidden status of the item.
- `author` list, A list containing the following elements:
  - id integer, The ID of this user.

- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`myPermissionLevel`
string, Your permission level on the object. One of "read", "write", or "manage".

`templates_put_scripts_projects`
Add a Script Template to a project

Description

Add a Script Template to a project

Usage

```
templates_put_scripts_projects(id, project_id)
```

Arguments

`id`	integer required. The ID of the Script Template.
`project_id`	integer required. The ID of the project.

Value

An empty HTTP response

`templates_put_scripts_shares_groups`
Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
templates_put_scripts_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

templates_put_scripts_shares_users

Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

```
templates_put_scripts_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

templates_put_scripts_transfer

Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
templates_put_scripts_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. ID of target user
include_dependencies	boolean required. Whether or not to give manage permissions on all dependencies
email_body	string optional. Custom body text for e-mail sent on transfer.
send_email	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies — array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

transfer_table *Transfer a table from one location to another.*

Description

Transfer a table from one location to another.

Usage

```
transfer_table(
  source_db,
  dest_db,
  source_table,
  dest_table,
  job_name = NULL,
  source_credential_id = NULL,
  dest_credential_id = NULL,
  interval = NULL,
  verbose = FALSE,
  advanced_options = NULL
)
```

Arguments

source_db	string or int, The name of the database where the source table is located. Optionally, could be the database ID.
dest_db	string or int, The name of the database where the table will be transferred. Optionally, could be the database ID.
source_table	string, Full name of the table to transfer, e.g., "schema.table".
dest_table	string, Full name of the table in the destination database, e.g., "schema.table".
job_name	string, optional, A name to give the job. If omitted, a random job name will be used.
source_credential_id	string or int, Optional credential ID for the source database. If NULL, the default credential will be used.
dest_credential_id	string or int, Optional credential ID for the source database. If NULL, the default credential will be used.
interval	Number of seconds to wait between checks for job completion. If NULL, the default exponential backoff from await will be used.
verbose	bool, Set to TRUE to print intermediate progress indicators.
advanced_options	A list of advanced options for the sync. See imports_post_syncs for details.

Value

A civis_api object.

See Also

Other tables: get_table_id(), refresh_table()

Examples

```
## Not run:
transfer_table(source_db='Cluster A', dest_db='Cluster B',
               source_table='schma.tbl', dest_table='schma.tbl')

## End(Not run)
```

---

usage_get_llm *Get an individual usage statistic for a given organization*

Description

Get an individual usage statistic for a given organization

Usage

```
usage_get_llm(id)
```

Arguments

id	integer required. The ID of the usage statistic to get.

Value

A list containing the following elements:

id	integer, The ID of the usage statistic to get.
runId	integer, The ID of the run which contributed this usage.
jobId	integer, The ID of the job which contributed this usage.
userId	integer, The ID of the user who contributed this usage.
organizationId	integer, The organization of the user who contributed this usage.
runCreatedAt	string, When the run was created at.
runTime	integer, The duration of the run in seconds.
credits	number, The number of credits used.
inputTokens	integer, The number of tokens input to the run.
outputTokens	integer, The number of tokens output from the run.
modelId	string, The ID of the LLM model used.

---

usage_limits_get_llm *Get a LLM Usage Limit*

Description

Get a LLM Usage Limit

Usage

```
usage_limits_get_llm(id)
```

Arguments

id	integer required. The ID for the limit.

Value

A list containing the following elements:

id	integer, The ID for the limit.
organizationId	integer, The ID of the organization to which this limit belongs.
createdAt	string, The time this limit was created.
updatedAt	string, The time the limit was last updated.
hardLimit	integer, The limit value. One of 1000, 10000, 50000, and 100000.

usage_limits_get_matching *Get a Matching Usage Limit*

Description

Get a Matching Usage Limit

Usage

```
usage_limits_get_matching(id)
```

Arguments

id	integer required. The ID for the limit.

Value

A list containing the following elements:

id	integer, The ID for the limit.
organizationId	integer, The ID of the organization to which this limit belongs.
createdAt	string, The time this limit was created.
updatedAt	string, The time the limit was last updated.
hardLimit	integer, The limit value. One of 50000000, 200000000, 500000000, 1000000000, and 2000000000.
task	string, The category of this limit. One of 'IDR' or 'CDM'.
notificationEmails	array, Addresses to notify by e-mail when the limit is reached.

usage_limits_list_llm *List LLM Usage Limits*

Description

List LLM Usage Limits

Usage

```
usage_limits_list_llm(organization_id = NULL)
```

Arguments

organization_id	integer optional. If specified, return limits for this organization only.

Value

An array containing the following fields:

id	integer, The ID for the limit.
organizationId	integer, The ID of the organization to which this limit belongs.
createdAt	string, The time this limit was created.
updatedAt	string, The time the limit was last updated.
hardLimit	integer, The limit value. One of 1000, 10000, 50000, and 100000.

---

usage_limits_list_matching

List Matching Usage Limits

Description

List Matching Usage Limits

Usage

```
usage_limits_list_matching(task = NULL)
```

Arguments

task	string optional. If specified, return limits for this task type only. One of 'IDR' or 'CDM'.

Value

An array containing the following fields:

id	integer, The ID for the limit.
organizationId	integer, The ID of the organization to which this limit belongs.
createdAt	string, The time this limit was created.
updatedAt	string, The time the limit was last updated.
hardLimit	integer, The limit value. One of 50000000, 200000000, 500000000, 1000000000, and 2000000000.
task	string, The category of this limit. One of 'IDR' or 'CDM'.
notificationEmails	array, Addresses to notify by e-mail when the limit is reached.

usage_list_llm *Get a list of usage statistics for a given organization*

Description

Get a list of usage statistics for a given organization

Usage

```
usage_list_llm(org_id = NULL, start_date = NULL, end_date = NULL)
```

Arguments

org_id	integer optional. The ID of the organization to get usage statistics for.
start_date	string optional. The start date of the range to get usage statistics for."\ "Defaults to the start of the current month if neither start_date nor end_date is specified.
end_date	string optional. The end date of the range to get usage statistics for."\ "Defaults to the end of the current day if neither start_date nor end_date is specified.

Value

An array containing the following fields:

id	integer, The ID of the usage statistic to get.
runId	integer, The ID of the run which contributed this usage.
jobId	integer, The ID of the job which contributed this usage.
userId	integer, The ID of the user who contributed this usage.
organizationId	integer, The organization of the user who contributed this usage.
runCreatedAt	string, When the run was created at.
runTime	integer, The duration of the run in seconds.
credits	number, The number of credits used.
inputTokens	integer, The number of tokens input to the run.
outputTokens	integer, The number of tokens output from the run.
modelId	string, The ID of the LLM model used.

usage_list_llm_organization_summary *Get summarized usage statistics for a given organization*

Description

Get summarized usage statistics for a given organization

Usage

```
usage_list_llm_organization_summary(org_id, start_date = NULL, end_date = NULL)
```

Arguments

org_id	integer required. The ID of the organization to get usage statistics for.
start_date	string optional. The start date of the range to get usage statistics for."\ "Defaults to the start of the current month if neither start_date nor end_date is specified.
end_date	string optional. The end date of the range to get usage statistics for."\ "Defaults to the end of the current day if neither start_date nor end_date is specified.

Value

A list containing the following elements:

credits	number, The number of credits used.
organizationId	integer, The organization for which LLM usage statistics are summarized.

usage_list_matching *Get usage statistics for a given organization*

Description

Get usage statistics for a given organization

Usage

```
usage_list_matching(
  org_id = NULL,
  task = NULL,
  start_date = NULL,
  end_date = NULL
)
```

Arguments

org_id	integer optional. The ID of the organization to get usage statistics for.
task	string optional. The type of matching job contributing to this usage. One of ["IDR", "CDM"].
start_date	string optional. The start date of the range to get usage statistics for.
end_date	string optional. The end date of the range to get usage statistics for.

Value

An array containing the following fields:

runId	integer, The ID of the run which contributed this usage.
jobId	integer, The ID of the job which contributed this usage.
userId	integer, The ID of the user who contributed this usage.
organizationId	integer, The organization of the user who contributed this usage.
runCreatedAt	string, When the run was created at.
runTime	integer, The duration of the run in seconds.
numRecords	integer, The number of records matched by the run.
task	string, The type of matching job contributing to this usage. One of ["IDR", "CDM"].

---

users_delete_2fa *Wipes the user's current 2FA settings so that they must reset them upon next login*

Description

Wipes the user's current 2FA settings so that they must reset them upon next login

Usage

```
users_delete_2fa(id)
```

Arguments

id	integer required. The ID of this user.

Value

A list containing the following elements:

`id`	integer, The ID of this user.
`user`	string, The username of this user.
`name`	string, The name of this user.
`email`	string, The email of this user.
`active`	boolean, Whether this user account is active or deactivated.
`primaryGroupId`	integer, The ID of the primary group of this user.
`groups`	array, An array containing the following fields: • id integer, The ID of this group. • name string, The name of this group. • slug string, The slug of this group. • organizationId integer, The ID of the organization associated with this group. • organizationName string, The name of the organization associated with this group.
`city`	string, The city of this user.
`state`	string, The state of this user.
`timeZone`	string, The time zone of this user.
`initials`	string, The initials of this user.
`department`	string, The department of this user.
`title`	string, The title of this user.
`githubUsername`	string, The GitHub username of this user.
`prefersSmsOtp`	boolean, The preference for phone authorization of this user
`vpnEnabled`	boolean, The availability of vpn for this user.
`ssoDisabled`	boolean, The availability of SSO for this user.
`otpRequiredForLogin`	boolean, The two factor authentication requirement for this user.
`exemptFromOrgSmsOtpDisabled`	boolean, Whether the user has SMS OTP enabled on an individual level. This field does not matter if the org does not have SMS OTP disabled.
`smsOtpAllowed`	boolean, Whether the user is allowed to receive two factor authentication codes via SMS.
`robot`	boolean, Whether the user is a robot.
`phone`	string, The phone number of this user.
`organizationSlug`	string, The slug of the organization the user belongs to.
`organizationSSODisableCapable`	boolean, The user's organization's ability to disable sso for their users.
`organizationLoginType`	string, The user's organization's login type.

`organizationSmsOtpDisabled`
: boolean, Whether the user's organization has SMS OTP disabled.

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`createdAt`
: string, The date and time when the user was created.

`updatedAt`
: string, The date and time when the user was last updated.

`lastSeenAt`
: string, The date and time when the user last visited Platform.

`suspended`
: boolean, Whether the user is suspended due to inactivity.

`createdById`
: integer, The ID of the user who created this user.

`lastUpdatedById`
: integer, The ID of the user who last updated this user.

`unconfirmedEmail`
: string, The new email address awaiting confirmation from the user.

`accountStatus`
: string, Account status of this user. One of: "Active", "Deactivated", "Suspended", "Unsuspended"

`users_delete_api_keys` *Revoke the specified API key*

Description

Revoke the specified API key

Usage

```
users_delete_api_keys(id, key_id)
```

Arguments

`id`
: string required. The ID of the user or 'me'.

`key_id`
: integer required. The ID of the API key.

Value

A list containing the following elements:

`id`
: integer, The ID of the API key.

`name`
: string, The name of the API key.

`expiresAt`
: string, The date and time when the key expired.

`createdAt`
: string, The date and time when the key was created.

`revokedAt`
: string, The date and time when the key was revoked.

`lastUsedAt`
: string, The date and time when the key was last used.

`scopes`
: array, The scopes which the key is permissioned on.

useCount	integer, The number of times the key has been used.
expired	boolean, True if the key has expired.
active	boolean, True if the key has neither expired nor been revoked.
constraints	array, An array containing the following fields: • constraint string, The path matcher of the constraint. • constraintType string, The type of constraint (exact/prefix/regex/verb). • getAllowed boolean, Whether the constraint allows GET requests. • headAllowed boolean, Whether the constraint allows HEAD requests. • postAllowed boolean, Whether the constraint allows POST requests. • putAllowed boolean, Whether the constraint allows PUT requests. • patchAllowed boolean, Whether the constraint allows PATCH requests. • deleteAllowed boolean, Whether the constraint allows DELETE requests.

`users_delete_me_favorites`
Unfavorite an item

Description

Unfavorite an item

Usage

```
users_delete_me_favorites(id)
```

Arguments

id	integer required. The id of the favorite.

Value

An empty HTTP response

`users_delete_sessions` *Terminate all of the user's active sessions (must be a team or org admin)*

Description

Terminate all of the user's active sessions (must be a team or org admin)

Usage

```
users_delete_sessions(id)
```

Arguments

`id`	integer required. The ID of this user.

Value

A list containing the following elements:

`id`	integer, The ID of this user.
`user`	string, The username of this user.
`name`	string, The name of this user.
`email`	string, The email of this user.
`active`	boolean, Whether this user account is active or deactivated.
`primaryGroupId`	integer, The ID of the primary group of this user.
`groups`	array, An array containing the following fields: • id integer, The ID of this group. • name string, The name of this group. • slug string, The slug of this group. • organizationId integer, The ID of the organization associated with this group. • organizationName string, The name of the organization associated with this group.
`city`	string, The city of this user.
`state`	string, The state of this user.
`timeZone`	string, The time zone of this user.
`initials`	string, The initials of this user.
`department`	string, The department of this user.
`title`	string, The title of this user.
`githubUsername`	string, The GitHub username of this user.
`prefersSmsOtp`	boolean, The preference for phone authorization of this user
`vpnEnabled`	boolean, The availability of vpn for this user.

ssoDisabled boolean, The availability of SSO for this user.

otpRequiredForLogin
boolean, The two factor authentication requirement for this user.

exemptFromOrgSmsOtpDisabled
boolean, Whether the user has SMS OTP enabled on an individual level. This field does not matter if the org does not have SMS OTP disabled.

smsOtpAllowed boolean, Whether the user is allowed to receive two factor authentication codes via SMS.

robot boolean, Whether the user is a robot.

phone string, The phone number of this user.

organizationSlug
string, The slug of the organization the user belongs to.

organizationSSODisableCapable
boolean, The user's organization's ability to disable sso for their users.

organizationLoginType
string, The user's organization's login type.

organizationSmsOtpDisabled
boolean, Whether the user's organization has SMS OTP disabled.

myPermissionLevel
string, Your permission level on the object. One of "read", "write", or "manage".

createdAt string, The date and time when the user was created.

updatedAt string, The date and time when the user was last updated.

lastSeenAt string, The date and time when the user last visited Platform.

suspended boolean, Whether the user is suspended due to inactivity.

createdById integer, The ID of the user who created this user.

lastUpdatedById
integer, The ID of the user who last updated this user.

unconfirmedEmail
string, The new email address awaiting confirmation from the user.

accountStatus string, Account status of this user. One of: "Active", "Deactivated", "Suspended", "Unsuspended"

---

users_get *Show info about a user*

Description

Show info about a user

Usage

```
users_get(id)
```

Arguments

id	integer required. The ID of this user.

Value

A list containing the following elements:

id	integer, The ID of this user.
user	string, The username of this user.
name	string, The name of this user.
email	string, The email of this user.
active	boolean, Whether this user account is active or deactivated.
primaryGroupId	integer, The ID of the primary group of this user.
groups	array, An array containing the following fields: • id integer, The ID of this group. • name string, The name of this group. • slug string, The slug of this group. • organizationId integer, The ID of the organization associated with this group. • organizationName string, The name of the organization associated with this group.
city	string, The city of this user.
state	string, The state of this user.
timeZone	string, The time zone of this user.
initials	string, The initials of this user.
department	string, The department of this user.
title	string, The title of this user.
githubUsername	string, The GitHub username of this user.
prefersSmsOtp	boolean, The preference for phone authorization of this user
vpnEnabled	boolean, The availability of vpn for this user.
ssoDisabled	boolean, The availability of SSO for this user.
otpRequiredForLogin	boolean, The two factor authentication requirement for this user.
exemptFromOrgSmsOtpDisabled	boolean, Whether the user has SMS OTP enabled on an individual level. This field does not matter if the org does not have SMS OTP disabled.
smsOtpAllowed	boolean, Whether the user is allowed to receive two factor authentication codes via SMS.
robot	boolean, Whether the user is a robot.
phone	string, The phone number of this user.
organizationSlug	string, The slug of the organization the user belongs to.

`organizationSSODisableCapable`
: boolean, The user's organization's ability to disable sso for their users.

`organizationLoginType`
: string, The user's organization's login type.

`organizationSmsOtpDisabled`
: boolean, Whether the user's organization has SMS OTP disabled.

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`createdAt` string, The date and time when the user was created.

`updatedAt` string, The date and time when the user was last updated.

`lastSeenAt` string, The date and time when the user last visited Platform.

`suspended` boolean, Whether the user is suspended due to inactivity.

`createdById` integer, The ID of the user who created this user.

`lastUpdatedById`
: integer, The ID of the user who last updated this user.

`unconfirmedEmail`
: string, The new email address awaiting confirmation from the user.

`accountStatus` string, Account status of this user. One of: "Active", "Deactivated", "Suspended", "Unsuspended"

`users_get_api_keys` *Show the specified API key*

Description

Show the specified API key

Usage

```
users_get_api_keys(id, key_id)
```

Arguments

`id`	string required. The ID of the user or 'me'.
`key_id`	integer required. The ID of the API key.

Value

A list containing the following elements:

`id`	integer, The ID of the API key.
`name`	string, The name of the API key.
`expiresAt`	string, The date and time when the key expired.
`createdAt`	string, The date and time when the key was created.

revokedAt	string, The date and time when the key was revoked.
lastUsedAt	string, The date and time when the key was last used.
scopes	array, The scopes which the key is permissioned on.
useCount	integer, The number of times the key has been used.
expired	boolean, True if the key has expired.
active	boolean, True if the key has neither expired nor been revoked.
constraints	array, An array containing the following fields: • constraint string, The path matcher of the constraint. • constraintType string, The type of constraint (exact/prefix/regex/verb). • getAllowed boolean, Whether the constraint allows GET requests. • headAllowed boolean, Whether the constraint allows HEAD requests. • postAllowed boolean, Whether the constraint allows POST requests. • putAllowed boolean, Whether the constraint allows PUT requests. • patchAllowed boolean, Whether the constraint allows PATCH requests. • deleteAllowed boolean, Whether the constraint allows DELETE requests.

`users_list` *List users*

Description

List users

Usage

```
users_list(
  feature_flag = NULL,
  account_status = NULL,
  query = NULL,
  group_id = NULL,
  group_ids = NULL,
  organization_id = NULL,
  exclude_groups = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

feature_flag	string optional. Return users that have a feature flag enabled.
account_status	string optional. The account status by which to filter users. May be one of "active", "inactive", or "all". Defaults to active.
query	string optional. Return users who match the given query, based on name, user, email, and id.
group_id	integer optional. The ID of the group by which to filter users. Cannot be present if group_ids is.
group_ids	array optional. The IDs of the groups by which to filter users. Cannot be present if group_id is.
organization_id	integer optional. The ID of the organization by which to filter users.
exclude_groups	boolean optional. Whether or to exclude users' groups. Default: false.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 10000.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to name. Must be one of: name, user.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id	integer, The ID of this user.
user	string, The username of this user.
name	string, The name of this user.
email	string, The email of this user.
active	boolean, Whether this user account is active or deactivated.
primaryGroupId	integer, The ID of the primary group of this user.
groups	array, An array containing the following fields: • id integer, The ID of this group. • name string, The name of this group. • slug string, The slug of this group. • organizationId integer, The ID of the organization associated with this group. • organizationName string, The name of the organization associated with this group.
createdAt	string, The date and time when the user was created.
currentSignInAt	string, The date and time when the user's current session began.

updatedAt	string, The date and time when the user was last updated.
lastSeenAt	string, The date and time when the user last visited Platform.
suspended	boolean, Whether the user is suspended due to inactivity.
createdById	integer, The ID of the user who created this user.
lastUpdatedById	integer, The ID of the user who last updated this user.

`users_list_api_keys` *Show API keys belonging to the specified user*

Description

Show API keys belonging to the specified user

Usage

```
users_list_api_keys(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

id	string required. The ID of the user or 'me'.
limit	integer optional. Number of results to return. Defaults to its maximum of 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID of the API key.
name	string, The name of the API key.
expiresAt	string, The date and time when the key expired.
createdAt	string, The date and time when the key was created.
revokedAt	string, The date and time when the key was revoked.

lastUsedAt	string, The date and time when the key was last used.
scopes	array, The scopes which the key is permissioned on.
useCount	integer, The number of times the key has been used.
expired	boolean, True if the key has expired.
active	boolean, True if the key has neither expired nor been revoked.
constraintCount	integer, The number of constraints on the created key

---

`users_list_me` *Show info about the logged-in user*

Description

Show info about the logged-in user

Usage

```
users_list_me()
```

Value

A list containing the following elements:

id	integer, The ID of this user.
name	string, This user's name.
email	string, This user's email address.
username	string, This user's username.
initials	string, This user's initials.
lastCheckedAnnouncements	string, The date and time at which the user last checked their announcements.
featureFlags	list, The feature flag settings for this user.
roles	array, The roles this user has, listed by slug.
preferences	list, This user's preferences.
customBranding	string, The branding of Platform for this user.
primaryGroupId	integer, The ID of the primary group of this user.
groups	array, An array containing the following fields: • id integer, The ID of this group. • name string, The name of this group. • slug string, The slug of this group. • organizationId integer, The ID of the organization associated with this group. • organizationName string, The name of the organization associated with this group.

`organizationName`
: string, The name of the organization the user belongs to.

`organizationSlug`
: string, The slug of the organization the user belongs to.

`organizationDefaultThemeId`
: integer, The ID of the organizations's default theme.

`createdAt`
: string, The date and time when the user was created.

`signInCount`
: integer, The number of times the user has signed in.

`assumingRole`
: boolean, Whether the user is assuming this role or not.

`roleAssumer`
: list, Details about the role assumer.

`assumingAdmin`
: boolean, Whether the user is assuming admin.

`assumingAdminExpiration`
: string, When the user's admin role is set to expire.

`superadminModeExpiration`
: string, The user is in superadmin mode when set to a DateTime. The user is not in superadmin mode when set to null.

`disableNonCompliantFedrampFeatures`
: boolean, Whether to disable non-compliant fedramp features.

`personaRole`
: string, The high-level role representing the current user's main permissions.

`createdById`
: integer, The ID of the user who created this user.

`lastUpdatedById`
: integer, The ID of the user who last updated this user.

`users_list_me_activity`
Get recent activity for logged-in user

Description

Get recent activity for logged-in user

Usage

```
users_list_me_activity(status = NULL, author = NULL, order = NULL)
```

Arguments

`status`
: string optional. The status to filter objects by. One of "all", "succeeded", "failed", or "running".

`author`
: string optional. A comma separated list of author IDs to filter objects by.

`order`
: string optional. The order of the jobs. If set to "name", the order is DESC alphabetically. If set to "newest", the order is DESC by most recently updated.

Value

An array containing the following fields:

id	string, The ID of the object.
name	string, The name of the object.
type	string, The type of the object.
user	string, The user associated with the object.
category	string, The job category, if the object is a job.
state	string, The state of the object. One of "succeeded", "failed", or "running".
updatedAt	string, When the object was last updated.
nextRunAt	string, When the job is next scheduled to run, if the object is a job.
lastRunId	string, The ID of the last run, if the object is a job.
lastRunState	string, The state of the last run, if the object is a job. One of "succeeded", "failed", or "running".

`users_list_me_favorites`
List Favorites

Description

List Favorites

Usage

```
users_list_me_favorites(
  object_id = NULL,
  object_type = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

object_id	integer optional. The id of the object. If specified as a query parameter, must also specify object_type parameter.
object_type	string optional. The type of the object that is favorited. Valid options: Container Script, Identity Resolution, Import, Python Script, R Script, dbt Script, JavaScript Script, SQL Script, Template Script, Project, Workflow, Table, Tableau Report, Service Report, HTML Report, SQL Report
limit	integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.

page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to position. Must be one of: position, created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id	integer, The id of the favorite.
objectId	integer, The id of the object. If specified as a query parameter, must also specify object_type parameter.
objectType	string, The type of the object that is favorited. Valid options: Container Script, Identity Resolution, Import, Python Script, R Script, dbt Script, JavaScript Script, SQL Script, Template Script, Project, Workflow, Table, Tableau Report, Service Report, HTML Report, SQL Report
objectName	string, The name of the object that is favorited.
createdAt	string, The time this favorite was created.
objectUpdatedAt	string, The time the object that is favorited was last updated
objectAuthor	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
position	integer, The rank position of this favorite. Use the patch users/me/favorites/:id/ranking/ endpoints to update.

`users_list_me_organization_admins`
Get list of organization admins for logged-in user

Description

Get list of organization admins for logged-in user

Usage

```
users_list_me_organization_admins()
```

Value

An array containing the following fields:

`id`	integer, The ID of this user.
`name`	string, This user's name.
`username`	string, This user's username.
`initials`	string, This user's initials.
`online`	boolean, Whether this user is online.
`email`	string, This user's email address.

`users_patch` *Update info about a user (must be a team or org admin)*

Description

Update info about a user (must be a team or org admin)

Usage

```
users_patch(
  id,
  name = NULL,
  email = NULL,
  active = NULL,
  primary_group_id = NULL,
  city = NULL,
  state = NULL,
  time_zone = NULL,
  initials = NULL,
  department = NULL,
  title = NULL,
  prefers_sms_otp = NULL,
  group_ids = NULL,
  vpn_enabled = NULL,
  sso_disabled = NULL,
  otp_required_for_login = NULL,
  exempt_from_org_sms_otp_disabled = NULL,
  robot = NULL,
  phone = NULL,
  password = NULL,
  account_status = NULL
)
```

Arguments

id	integer required. The ID of this user.
name	string optional. The name of this user.
email	string optional. The email of this user.
active	boolean optional. Whether this user account is active or deactivated.
primary_group_id	integer optional. The ID of the primary group of this user.
city	string optional. The city of this user.
state	string optional. The state of this user.
time_zone	string optional. The time zone of this user.
initials	string optional. The initials of this user.
department	string optional. The department of this user.
title	string optional. The title of this user.
prefers_sms_otp	boolean optional. The preference for phone authorization of this user
group_ids	array optional. An array of ids of all the groups this user is in.
vpn_enabled	boolean optional. The availability of vpn for this user.
sso_disabled	boolean optional. The availability of SSO for this user.
otp_required_for_login	boolean optional. The two factor authentication requirement for this user.
exempt_from_org_sms_otp_disabled	boolean optional. Whether the user has SMS OTP enabled on an individual level. This field does not matter if the org does not have SMS OTP disabled.
robot	boolean optional. Whether the user is a robot.
phone	string optional. The phone number of this user.
password	string optional. The password of this user.
account_status	string optional. Account status of this user. One of: "Active", "Deactivated", "Suspended", "Unsuspended"

Value

A list containing the following elements:

id	integer, The ID of this user.
user	string, The username of this user.
name	string, The name of this user.
email	string, The email of this user.
active	boolean, Whether this user account is active or deactivated.
primaryGroupId	integer, The ID of the primary group of this user.
groups	array, An array containing the following fields: • id integer, The ID of this group.

- name string, The name of this group.
- slug string, The slug of this group.
- organizationId integer, The ID of the organization associated with this group.
- organizationName string, The name of the organization associated with this group.

`city` string, The city of this user.

`state` string, The state of this user.

`timeZone` string, The time zone of this user.

`initials` string, The initials of this user.

`department` string, The department of this user.

`title` string, The title of this user.

`githubUsername` string, The GitHub username of this user.

`prefersSmsOtp` boolean, The preference for phone authorization of this user

`vpnEnabled` boolean, The availability of vpn for this user.

`ssoDisabled` boolean, The availability of SSO for this user.

`otpRequiredForLogin` boolean, The two factor authentication requirement for this user.

`exemptFromOrgSmsOtpDisabled` boolean, Whether the user has SMS OTP enabled on an individual level. This field does not matter if the org does not have SMS OTP disabled.

`smsOtpAllowed` boolean, Whether the user is allowed to receive two factor authentication codes via SMS.

`robot` boolean, Whether the user is a robot.

`phone` string, The phone number of this user.

`organizationSlug` string, The slug of the organization the user belongs to.

`organizationSSODisableCapable` boolean, The user's organization's ability to disable sso for their users.

`organizationLoginType` string, The user's organization's login type.

`organizationSmsOtpDisabled` boolean, Whether the user's organization has SMS OTP disabled.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`createdAt` string, The date and time when the user was created.

`updatedAt` string, The date and time when the user was last updated.

`lastSeenAt` string, The date and time when the user last visited Platform.

`suspended` boolean, Whether the user is suspended due to inactivity.

`createdById` integer, The ID of the user who created this user.

`lastUpdatedById` integer, The ID of the user who last updated this user.

unconfirmedEmail
: string, The new email address awaiting confirmation from the user.

accountStatus
: string, Account status of this user. One of: "Active", "Deactivated", "Suspended", "Unsuspended"

`users_patch_me` *Update info about the logged-in user*

Description

Update info about the logged-in user

Usage

```
users_patch_me(preferences = NULL, last_checked_announcements = NULL)
```

Arguments

preferences
: list optional. A list containing the following elements:

- appIndexOrderField string, This attribute is deprecated
- appIndexOrderDir string, This attribute is deprecated
- resultIndexOrderField string, Order field for the reports index page.
- resultIndexOrderDir string, Order direction for the reports index page.
- resultIndexTypeFilter string, Type filter for the reports index page.
- resultIndexAuthorFilter string, Author filter for the reports index page.
- resultIndexArchivedFilter string, Archived filter for the reports index page.
- importIndexOrderField string, Order field for the imports index page.
- importIndexOrderDir string, Order direction for the imports index page.
- importIndexTypeFilter string, Type filter for the imports index page.
- importIndexAuthorFilter string, Author filter for the imports index page.
- importIndexDestFilter string, Destination filter for the imports index page.
- importIndexStatusFilter string, Status filter for the imports index page.
- importIndexArchivedFilter string, Archived filter for the imports index page.
- exportIndexOrderField string, Order field for the exports index page.
- exportIndexOrderDir string, Order direction for the exports index page.
- exportIndexTypeFilter string, Type filter for the exports index page.
- exportIndexAuthorFilter string, Author filter for the exports index page.
- exportIndexStatusFilter string, Status filter for the exports index page.
- modelIndexOrderField string, Order field for the models index page.
- modelIndexOrderDir string, Order direction for the models index page.
- modelIndexAuthorFilter string, Author filter for the models index page.
- modelIndexStatusFilter string, Status filter for the models index page.
- modelIndexArchivedFilter string, Archived filter for the models index page.

- modelIndexThumbnailView string, Thumbnail view for the models index page.
- scriptIndexOrderField string, Order field for the scripts index page.
- scriptIndexOrderDir string, Order direction for the scripts index page.
- scriptIndexTypeFilter string, Type filter for the scripts index page.
- scriptIndexAuthorFilter string, Author filter for the scripts index page.
- scriptIndexStatusFilter string, Status filter for the scripts index page.
- scriptIndexArchivedFilter string, Archived filter for the scripts index page.
- projectIndexOrderField string, Order field for the projects index page.
- projectIndexOrderDir string, Order direction for the projects index page.
- projectIndexAuthorFilter string, Author filter for the projects index page.
- projectIndexArchivedFilter string, Archived filter for the projects index page.
- reportIndexThumbnailView string, Thumbnail view for the reports index page.
- projectDetailOrderField string, Order field for projects detail pages.
- projectDetailOrderDir string, Order direction for projects detail pages.
- projectDetailAuthorFilter string, Author filter for projects detail pages.
- projectDetailTypeFilter string, Type filter for projects detail pages.
- projectDetailArchivedFilter string, Archived filter for the projects detail pages.
- enhancementIndexOrderField string, Order field for the enhancements index page.
- enhancementIndexOrderDir string, Order direction for the enhancements index page.
- enhancementIndexAuthorFilter string, Author filter for the enhancements index page.
- enhancementIndexArchivedFilter string, Archived filter for the enhancements index page.
- preferredServerId integer, ID of preferred server.
- civisExploreSkipIntro boolean, Whether the user is shown steps for each exploration.
- registrationIndexOrderField string, Order field for the registrations index page.
- registrationIndexOrderDir string, Order direction for the registrations index page.
- registrationIndexStatusFilter string, Status filter for the registrations index page.
- upgradeRequested string, Whether a free trial upgrade has been requested.
- welcomeOrderField string, Order direction for the welcome page.
- welcomeOrderDir string, Order direction for the welcome page.
- welcomeAuthorFilter string, Status filter for the welcome page.
- welcomeStatusFilter string, Status filter for the welcome page.
- welcomeArchivedFilter string, Status filter for the welcome page.
- dataPaneWidth string, Width of the data pane when expanded.

- dataPaneCollapsed string, Whether the data pane is collapsed.
- notebookOrderField string, Order field for the notebooks page.
- notebookOrderDir string, Order direction for the notebooks page.
- notebookAuthorFilter string, Author filter for the notebooks page.
- notebookArchivedFilter string, Archived filter for the notebooks page.
- notebookStatusFilter string, Status filter for the notebooks page.
- workflowIndexOrderField string, Order field for the workflows page.
- workflowIndexOrderDir string, Order direction for the workflows page.
- workflowIndexAuthorFilter string, Author filter for the workflows page.
- workflowIndexArchivedFilter string, Archived filter for the workflows page.
- serviceOrderField string, Order field for the services page.
- serviceOrderDir string, Order direction for the services page.
- serviceAuthorFilter string, Author filter for the services page.
- serviceArchivedFilter string, Archived filter for the services page.
- assumeRoleHistory string, JSON string of previously assumed roles.
- defaultSuccessNotificationsOn boolean, Whether email notifications for the success of all applicable jobs are on by default.
- defaultFailureNotificationsOn boolean, Whether email notifications for the failure of all applicable jobs are on by default.
- myActivityMetrics boolean, Whether the activity metrics are filtered to the current user.
- standardSQLAutocompleteDisabled boolean, Whether the query page includes standard SQL autocomplete.
- aiSQLAssistDisabled boolean, Whether the query page includes AI-powered SQL autocomplete.
- queryPreviewRows integer, Number of preview rows query should return in the UI.

`last_checked_announcements`
: string optional. The date and time at which the user last checked their announcements.

Value

A list containing the following elements:

`id` : integer, The ID of this user.

`name` : string, This user's name.

`email` : string, This user's email address.

`username` : string, This user's username.

`initials` : string, This user's initials.

`lastCheckedAnnouncements`
: string, The date and time at which the user last checked their announcements.

`featureFlags` : list, The feature flag settings for this user.

`roles` : array, The roles this user has, listed by slug.

`preferences` list, This user's preferences.

`customBranding` string, The branding of Platform for this user.

`primaryGroupId` integer, The ID of the primary group of this user.

`groups` array, An array containing the following fields:

- id integer, The ID of this group.
- name string, The name of this group.
- slug string, The slug of this group.
- organizationId integer, The ID of the organization associated with this group.
- organizationName string, The name of the organization associated with this group.

`organizationName` string, The name of the organization the user belongs to.

`organizationSlug` string, The slug of the organization the user belongs to.

`organizationDefaultThemeId` integer, The ID of the organizations's default theme.

`createdAt` string, The date and time when the user was created.

`signInCount` integer, The number of times the user has signed in.

`assumingRole` boolean, Whether the user is assuming this role or not.

`roleAssumer` list, Details about the role assumer.

`assumingAdmin` boolean, Whether the user is assuming admin.

`assumingAdminExpiration` string, When the user's admin role is set to expire.

`superadminModeExpiration` string, The user is in superadmin mode when set to a DateTime. The user is not in superadmin mode when set to null.

`disableNonCompliantFedrampFeatures` boolean, Whether to disable non-compliant fedramp features.

`personaRole` string, The high-level role representing the current user's main permissions.

`createdById` integer, The ID of the user who created this user.

`lastUpdatedById` integer, The ID of the user who last updated this user.

`users_patch_me_favorites_ranking_bottom` *Move a favorite to the bottom of the list*

Description

Move a favorite to the bottom of the list

Usage

```
users_patch_me_favorites_ranking_bottom(id)
```

Arguments

id	integer required. The id of the favorite.

Value

An empty HTTP response

`users_patch_me_favorites_ranking_higher`
Move a favorite one position closer to the top of the list

Description

Move a favorite one position closer to the top of the list

Usage

```
users_patch_me_favorites_ranking_higher(id)
```

Arguments

id	integer required. The id of the favorite.

Value

An empty HTTP response

`users_patch_me_favorites_ranking_lower`
Move a favorite one position closer to the bottom of the list

---

Description

Move a favorite one position closer to the bottom of the list

Usage

```
users_patch_me_favorites_ranking_lower(id)
```

Arguments

id	integer required. The id of the favorite.

Value

An empty HTTP response

`users_patch_me_favorites_ranking_top` *Move a favorite to the top of the list*

Description

Move a favorite to the top of the list

Usage

```
users_patch_me_favorites_ranking_top(id)
```

Arguments

`id`	integer required. The id of the favorite.

Value

An empty HTTP response

`users_post` *Create a new user (must be a team or org admin)*

Description

Create a new user (must be a team or org admin)

Usage

```
users_post(
  name,
  email,
  primary_group_id,
  user,
  active = NULL,
  city = NULL,
  state = NULL,
  time_zone = NULL,
  initials = NULL,
  department = NULL,
  title = NULL,
  prefers_sms_otp = NULL,
```

```
  group_ids = NULL,
  vpn_enabled = NULL,
  sso_disabled = NULL,
  otp_required_for_login = NULL,
  exempt_from_org_sms_otp_disabled = NULL,
  robot = NULL,
  send_email = NULL
)
```

Arguments

name	string required. The name of this user.
email	string required. The email of this user.
primary_group_id	integer required. The ID of the primary group of this user.
user	string required. The username of this user.
active	boolean optional. Whether this user account is active or deactivated.
city	string optional. The city of this user.
state	string optional. The state of this user.
time_zone	string optional. The time zone of this user.
initials	string optional. The initials of this user.
department	string optional. The department of this user.
title	string optional. The title of this user.
prefers_sms_otp	boolean optional. The preference for phone authorization of this user
group_ids	array optional. An array of ids of all the groups this user is in.
vpn_enabled	boolean optional. The availability of vpn for this user.
sso_disabled	boolean optional. The availability of SSO for this user.
otp_required_for_login	boolean optional. The two factor authentication requirement for this user.
exempt_from_org_sms_otp_disabled	boolean optional. Whether the user has SMS OTP enabled on an individual level. This field does not matter if the org does not have SMS OTP disabled.
robot	boolean optional. Whether the user is a robot.
send_email	boolean optional. Whether the user will receive a welcome email. Defaults to false.

Value

A list containing the following elements:

id	integer, The ID of this user.
user	string, The username of this user.

`name` string, The name of this user.

`email` string, The email of this user.

`active` boolean, Whether this user account is active or deactivated.

`primaryGroupId` integer, The ID of the primary group of this user.

`groups` array, An array containing the following fields:

- id integer, The ID of this group.
- name string, The name of this group.
- slug string, The slug of this group.
- organizationId integer, The ID of the organization associated with this group.
- organizationName string, The name of the organization associated with this group.

`city` string, The city of this user.

`state` string, The state of this user.

`timeZone` string, The time zone of this user.

`initials` string, The initials of this user.

`department` string, The department of this user.

`title` string, The title of this user.

`githubUsername` string, The GitHub username of this user.

`prefersSmsOtp` boolean, The preference for phone authorization of this user

`vpnEnabled` boolean, The availability of vpn for this user.

`ssoDisabled` boolean, The availability of SSO for this user.

`otpRequiredForLogin` boolean, The two factor authentication requirement for this user.

`exemptFromOrgSmsOtpDisabled` boolean, Whether the user has SMS OTP enabled on an individual level. This field does not matter if the org does not have SMS OTP disabled.

`smsOtpAllowed` boolean, Whether the user is allowed to receive two factor authentication codes via SMS.

`robot` boolean, Whether the user is a robot.

`phone` string, The phone number of this user.

`organizationSlug` string, The slug of the organization the user belongs to.

`organizationSSODisableCapable` boolean, The user's organization's ability to disable sso for their users.

`organizationLoginType` string, The user's organization's login type.

`organizationSmsOtpDisabled` boolean, Whether the user's organization has SMS OTP disabled.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`createdAt` string, The date and time when the user was created.

updatedAt	string, The date and time when the user was last updated.
lastSeenAt	string, The date and time when the user last visited Platform.
suspended	boolean, Whether the user is suspended due to inactivity.
createdById	integer, The ID of the user who created this user.
lastUpdatedById	integer, The ID of the user who last updated this user.
unconfirmedEmail	string, The new email address awaiting confirmation from the user.
accountStatus	string, Account status of this user. One of: "Active", "Deactivated", "Suspended", "Unsuspended"

`users_post_access_email` *Sends the target user a 'Reset Password' or 'Welcome to Platform' email depending on the their status - Only available to Org and Team Admins*

Description

Sends the target user a 'Reset Password' or 'Welcome to Platform' email depending on the their status - Only available to Org and Team Admins

Usage

```
users_post_access_email(id)
```

Arguments

id	integer required. The ID of this user.

Value

An empty HTTP response

`users_post_api_keys` *Create a new API key belonging to the logged-in user*

Description

Create a new API key belonging to the logged-in user

Usage

```
users_post_api_keys(id, expires_in, name, constraints = NULL)
```

Arguments

`id`	string required. The ID of the user or 'me'.
`expires_in`	integer required. The number of seconds the key should last for.
`name`	string required. The name of the API key.
`constraints`	array optional. An array containing the following fields: • constraint string, The path matcher of the constraint. • constraintType string, The type of constraint (exact/prefix/regex/verb). • getAllowed boolean, Whether the constraint allows GET requests. • headAllowed boolean, Whether the constraint allows HEAD requests. • postAllowed boolean, Whether the constraint allows POST requests. • putAllowed boolean, Whether the constraint allows PUT requests. • patchAllowed boolean, Whether the constraint allows PATCH requests. • deleteAllowed boolean, Whether the constraint allows DELETE requests.

Value

A list containing the following elements:

`id`	integer, The ID of the API key.
`name`	string, The name of the API key.
`expiresAt`	string, The date and time when the key expired.
`createdAt`	string, The date and time when the key was created.
`revokedAt`	string, The date and time when the key was revoked.
`lastUsedAt`	string, The date and time when the key was last used.
`scopes`	array, The scopes which the key is permissioned on.
`useCount`	integer, The number of times the key has been used.
`expired`	boolean, True if the key has expired.
`active`	boolean, True if the key has neither expired nor been revoked.
`constraints`	array, An array containing the following fields: • constraint string, The path matcher of the constraint.

- constraintType string, The type of constraint (exact/prefix/regex/verb).
- getAllowed boolean, Whether the constraint allows GET requests.
- headAllowed boolean, Whether the constraint allows HEAD requests.
- postAllowed boolean, Whether the constraint allows POST requests.
- putAllowed boolean, Whether the constraint allows PUT requests.
- patchAllowed boolean, Whether the constraint allows PATCH requests.
- deleteAllowed boolean, Whether the constraint allows DELETE requests.

`token` string, The API key.

`users_post_me_favorites` *Favorite an item*

Description

Favorite an item

Usage

```
users_post_me_favorites(object_id, object_type)
```

Arguments

`object_id`	integer required. The id of the object. If specified as a query parameter, must also specify object_type parameter.
`object_type`	string required. The type of the object that is favorited. Valid options: Container Script, Identity Resolution, Import, Python Script, R Script, dbt Script, JavaScript Script, SQL Script, Template Script, Project, Workflow, Table, Tableau Report, Service Report, HTML Report, SQL Report

Value

A list containing the following elements:

`id`	integer, The id of the favorite.
`objectId`	integer, The id of the object. If specified as a query parameter, must also specify object_type parameter.
`objectType`	string, The type of the object that is favorited. Valid options: Container Script, Identity Resolution, Import, Python Script, R Script, dbt Script, JavaScript Script, SQL Script, Template Script, Project, Workflow, Table, Tableau Report, Service Report, HTML Report, SQL Report
`objectName`	string, The name of the object that is favorited.
`createdAt`	string, The time this favorite was created.
`objectUpdatedAt`	string, The time the object that is favorited was last updated

objectAuthor	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.

`users_post_unsuspend` *Unsuspends user*

Description

Unsuspends user

Usage

```
users_post_unsuspend(id)
```

Arguments

id	integer required. The ID of this user.

Value

A list containing the following elements:

id	integer, The ID of this user.
user	string, The username of this user.
unlockedAt	string, The time the user's account was unsuspended

---

`workflows_delete_projects` *Remove a Workflow from a project*

Description

Remove a Workflow from a project

Usage

```
workflows_delete_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Workflow.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

`workflows_delete_shares_groups`
Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

```
workflows_delete_shares_groups(id, group_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`group_id`	integer required. The ID of the group.

Value

An empty HTTP response

`workflows_delete_shares_users`
Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

```
workflows_delete_shares_users(id, user_id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer required. The ID of the user.

Value

An empty HTTP response

workflows_get *Get a Workflow*

Description

Get a Workflow

Usage

```
workflows_get(id)
```

Arguments

id	integer required.

Value

A list containing the following elements:

id	integer, The ID for this workflow.
name	string, The name of this workflow.
description	string, A description of the workflow.
definition	string, The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain‘ is specified.
valid	boolean, The validity of the workflow definition.
validationErrors	string, The errors encountered when validating the workflow definition.
fileId	string, The file id for the s3 file containing the workflow configuration.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user’s name. • username string, This user’s username. • initials string, This user’s initials. • online boolean, Whether this user is online.
state	string, The state of the workflow. State is "running" if any execution is running, otherwise reflects most recent execution state.
lastExecutionId	integer, The ID of the most recent execution of this workflow, if any.
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on.

- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`allowConcurrentExecutions`
: boolean, Whether the workflow can execute when already running.

`timeZone`
: string, The time zone of this workflow.

`nextExecutionAt`
: string, The time of the next scheduled execution.

`notifications`
: list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on
- failureOn boolean, If failure email notifications are on

`archived`
: string, The archival status of the requested item(s).

`hidden`
: boolean, The hidden status of the item.

`myPermissionLevel`
: string, Your permission level on the object. One of "read", "write", or "manage".

`createdAt`
: string,

`updatedAt`
: string,

`workflows_get_executions`
Get a workflow execution

Description

Get a workflow execution

Usage

```
workflows_get_executions(id, execution_id)
```

Arguments

id	integer required. The ID for the workflow.
execution_id	integer required. The ID for the workflow execution.

Value

A list containing the following elements:

id	integer, The ID for this workflow execution.
state	string, The state of this workflow execution.
mistralState	string, The state of this workflow as reported by mistral. One of running, paused, success, error, or cancelled
mistralStateInfo	string, The state info of this workflow as reported by mistral.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
definition	string, The definition of the workflow for this execution.
input	list, Key-value pairs defined for this execution.
includedTasks	array, The subset of workflow tasks selected to execute.
tasks	array, An array containing the following fields: • name string, The name of the task. • mistralState string, The state of this task. One of idle, waiting, running, delayed, success, error, or cancelled • mistralStateInfo string, Extra info associated with the state of the task. • runs array, The runs associated with this task, in descending order by id. • executions array, The executions run by this task, in descending order by id.
startedAt	string, The time this execution started.
finishedAt	string, The time this execution finished.
createdAt	string, The time this execution was created.
updatedAt	string, The time this execution was last updated.

`workflows_get_executions_tasks` *Get a task of a workflow execution*

Description

Get a task of a workflow execution

Usage

```
workflows_get_executions_tasks(id, execution_id, task_name)
```

Arguments

`id`	integer required. The ID for the workflow.
`execution_id`	integer required. The ID for the workflow execution.
`task_name`	string required. The URL-encoded name of the task.

Value

A list containing the following elements:

`name`	string, The name of the task.
`mistralState`	string, The state of this task. One of idle, waiting, running, delayed, success, error, or cancelled
`mistralStateInfo`	string, Extra info associated with the state of the task.
`runs`	array, An array containing the following fields: • id integer, The ID of the run. • jobId integer, The ID of the job associated with the run. • myPermissionLevel string, Your permission level on the job. One of "read", "write", "manage", or "nil". • state string, The state of the run. • createdAt string, The time that the run was queued. • startedAt string, The time that the run started. • finishedAt string, The time that the run completed.
`executions`	array, An array containing the following fields: • id integer, The ID of the execution. • workflowId integer, The ID of the workflow associated with the execution. • myPermissionLevel string, Your permission level on the workflow. One of "read", "write", "manage", or "nil". • state string, The state of this workflow execution. • createdAt string, The time this execution was created. • startedAt string, The time this execution started. • finishedAt string, The time this execution finished.

workflows_get_git_commits *Get file contents at git ref*

Description

Get file contents at git ref

Usage

```
workflows_get_git_commits(id, commit_hash)
```

Arguments

id	integer required. The ID of the item.
commit_hash	string required. The SHA (full or shortened) of the desired git commit.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

workflows_list *List Workflows*

Description

List Workflows

Usage

```
workflows_list(
  hidden = NULL,
  archived = NULL,
  author = NULL,
  state = NULL,
  scheduled = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
archived	string optional. The archival status of the requested item(s).
author	string optional. If specified, return items from any of these authors. It accepts a comma-separated list of user IDs.
state	array optional. State of the most recent execution. One or more of queued, running, succeeded, failed, cancelled, idle, and scheduled. Note that the "scheduled" state applies only to scheduled workflows which have never been run. If you want to see all scheduled workflows, please use the "scheduled" filter instead.
scheduled	boolean optional. If the workflow is scheduled.
limit	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num	integer optional. Page number of the results to return. Defaults to the first page, 1.
order	string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at.
order_dir	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id	integer, The ID for this workflow.
name	string, The name of this workflow.
description	string, A description of the workflow.
valid	boolean, The validity of the workflow definition.
fileId	string, The file id for the s3 file containing the workflow configuration.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
state	string, The state of the workflow. State is "running" if any execution is running, otherwise reflects most recent execution state.
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth

- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`allowConcurrentExecutions`
: boolean, Whether the workflow can execute when already running.

`timeZone` : string, The time zone of this workflow.

`nextExecutionAt`
: string, The time of the next scheduled execution.

`archived` : string, The archival status of the requested item(s).

`createdAt` : string,

`updatedAt` : string,

`workflows_list_dependencies`
List dependent objects for this object

Description

List dependent objects for this object

Usage

```
workflows_list_dependencies(id, user_id = NULL)
```

Arguments

`id`	integer required. The ID of the resource that is shared.
`user_id`	integer optional. ID of target user

Value

An array containing the following fields:

`objectType` : string, Dependent object type

`fcoType` : string, Human readable dependent object type

`id` : integer, Dependent object ID

`name` : string, Dependent object name, or nil if the requesting user cannot read this object

`permissionLevel`
: string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).

`description` : string, Additional information about the dependency, if relevant

`shareable` : boolean, Whether or not the requesting user can share this object.

`workflows_list_executions` *List workflow executions*

Description

List workflow executions

Usage

```
workflows_list_executions(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

Arguments

`id`	integer required. The ID for this workflow.
`limit`	integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
`page_num`	integer optional. Page number of the results to return. Defaults to the first page, 1.
`order`	string optional. The field on which to order the result set. Defaults to id. Must be one of: id, updated_at, created_at.
`order_dir`	string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

`id`	integer, The ID for this workflow execution.
`state`	string, The state of this workflow execution.
`mistralState`	string, The state of this workflow as reported by mistral. One of running, paused, success, error, or cancelled
`mistralStateInfo`	string, The state info of this workflow as reported by mistral.
`user`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username.

- initials string, This user's initials.
- online boolean, Whether this user is online.

startedAt	string, The time this execution started.
finishedAt	string, The time this execution finished.
createdAt	string, The time this execution was created.
updatedAt	string, The time this execution was last updated.

`workflows_list_git` *Get the git metadata attached to an item*

Description

Get the git metadata attached to an item

Usage

```
workflows_list_git(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,
gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

workflows_list_git_commits

Get the git commits for an item on the current branch

Description

Get the git commits for an item on the current branch

Usage

```
workflows_list_git_commits(id)
```

Arguments

id	integer required. The ID of the item.

Value

A list containing the following elements:

commitHash	string, The SHA of the commit.
authorName	string, The name of the commit's author.
date	string, The commit's timestamp.
message	string, The commit message.

workflows_list_projects

List the projects a Workflow belongs to

Description

List the projects a Workflow belongs to

Usage

```
workflows_list_projects(id, hidden = NULL)
```

Arguments

id	integer required. The ID of the Workflow.
hidden	boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

`id`	integer, The ID for this project.
`author`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`name`	string, The name of this project.
`description`	string, A description of the project.
`users`	array, An array containing the following fields: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`autoShare`	boolean,
`createdAt`	string,
`updatedAt`	string,
`archived`	string, The archival status of the requested item(s).

`workflows_list_shares` *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
workflows_list_shares(id)
```

Arguments

`id`	integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

`readers` list, A list containing the following elements:

- users array,
- groups array,

`writers` list, A list containing the following elements:

- users array,
- groups array,

`owners` list, A list containing the following elements:

- users array,
- groups array,

`totalUserShares`
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

`totalGroupShares`
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

workflows_patch *Update some attributes of this Workflow*

Description

Update some attributes of this Workflow

Usage

```
workflows_patch(
  id,
  name = NULL,
  description = NULL,
  definition = NULL,
  schedule = NULL,
  allow_concurrent_executions = NULL,
  time_zone = NULL,
  notifications = NULL
)
```

Arguments

id	integer required. The ID for this workflow.
name	string optional. The name of this workflow.
description	string optional. A description of the workflow.
definition	string optional. The definition of the workflow in YAML format. Must not be specified if 'fromJobChain' is specified.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
allow_concurrent_executions	boolean optional. Whether the workflow can execute when already running.
time_zone	string optional. The time zone of this workflow.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully. • failureEmailAddresses array, Addresses to notify by e-mail when the job fails. • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes. • successOn boolean, If success email notifications are on • failureOn boolean, If failure email notifications are on

Value

A list containing the following elements:

id	integer, The ID for this workflow.
name	string, The name of this workflow.
description	string, A description of the workflow.
definition	string, The definition of the workflow in YAML format. Must not be specified if 'fromJobChain' is specified.
valid	boolean, The validity of the workflow definition.

validationErrors
: string, The errors encountered when validating the workflow definition.

fileId
: string, The file id for the s3 file containing the workflow configuration.

user
: list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

state
: string, The state of the workflow. State is "running" if any execution is running, otherwise reflects most recent execution state.

lastExecutionId
: integer, The ID of the most recent execution of this workflow, if any.

schedule
: list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
  - scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

allowConcurrentExecutions
: boolean, Whether the workflow can execute when already running.

timeZone
: string, The time zone of this workflow.

nextExecutionAt
: string, The time of the next scheduled execution.

notifications
: list, A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  - successOn boolean, If success email notifications are on
  - failureOn boolean, If failure email notifications are on

archived
: string, The archival status of the requested item(s).

hidden
: boolean, The hidden status of the item.

myPermissionLevel
: string, Your permission level on the object. One of "read", "write", or "manage".

createdAt
: string,

updatedAt
: string,

`workflows_patch_git` *Update an attached git file*

Description

Update an attached git file

Usage

```
workflows_patch_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements:

- id integer, The ID for this git repository.
- repoUrl string, The URL for this git repository.
- createdAt string,
- updatedAt string,

`gitRefType` string, Specifies if the file is versioned by branch or tag.

`pullFromGit` boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

`workflows_post` *Create a Workflow*

Description

Create a Workflow

Usage

```
workflows_post(
  name,
  description = NULL,
  from_job_chain = NULL,
  definition = NULL,
  schedule = NULL,
  allow_concurrent_executions = NULL,
  time_zone = NULL,
  notifications = NULL,
  hidden = NULL
)
```

Arguments

`name` string required. The name of this workflow.

`description` string optional. A description of the workflow.

`from_job_chain` integer optional. If specified, create a workflow from the job chain this job is in, and inherit the schedule from the root of the chain.

`definition` string optional. The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain‘ is specified.

`schedule` list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.

- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`allow_concurrent_executions`
: boolean optional. Whether the workflow can execute when already running.

`time_zone`
: string optional. The time zone of this workflow.

`notifications`
: list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on
- failureOn boolean, If failure email notifications are on

`hidden`
: boolean optional. The hidden status of the item.

Value

A list containing the following elements:

`id`
: integer, The ID for this workflow.

`name`
: string, The name of this workflow.

`description`
: string, A description of the workflow.

`definition`
: string, The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain‘ is specified.

`valid`
: boolean, The validity of the workflow definition.

`validationErrors`
: string, The errors encountered when validating the workflow definition.

`fileId`
: string, The file id for the s3 file containing the workflow configuration.

`user`
: list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user’s name.
- username string, This user’s username.
- initials string, This user’s initials.
- online boolean, Whether this user is online.

`state`
: string, The state of the workflow. State is "running" if any execution is running, otherwise reflects most recent execution state.

`lastExecutionId`
: integer, The ID of the most recent execution of this workflow, if any.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`allowConcurrentExecutions` boolean, Whether the workflow can execute when already running.

`timeZone` string, The time zone of this workflow.

`nextExecutionAt` string, The time of the next scheduled execution.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on
- failureOn boolean, If failure email notifications are on

`archived` string, The archival status of the requested item(s).

`hidden` boolean, The hidden status of the item.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`createdAt` string,

`updatedAt` string,

`workflows_post_clone` *Clone this Workflow*

Description

Clone this Workflow

Usage

```
workflows_post_clone(id, clone_schedule = NULL, clone_notifications = NULL)
```

Arguments

id	integer required. The ID for the workflow.
clone_schedule	boolean optional. If true, also copy the schedule to the new workflow.
clone_notifications	boolean optional. If true, also copy the notifications to the new workflow.

Value

A list containing the following elements:

id	integer, The ID for this workflow.
name	string, The name of this workflow.
description	string, A description of the workflow.
definition	string, The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain‘ is specified.
valid	boolean, The validity of the workflow definition.
validationErrors	string, The errors encountered when validating the workflow definition.
fileId	string, The file id for the s3 file containing the workflow configuration.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user’s name. • username string, This user’s username. • initials string, This user’s initials. • online boolean, Whether this user is online.
state	string, The state of the workflow. State is "running" if any execution is running, otherwise reflects most recent execution state.
lastExecutionId	integer, The ID of the most recent execution of this workflow, if any.
schedule	list, A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
allowConcurrentExecutions	boolean, Whether the workflow can execute when already running.

`timeZone` string, The time zone of this workflow.

`nextExecutionAt` string, The time of the next scheduled execution.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on
- failureOn boolean, If failure email notifications are on

`archived` string, The archival status of the requested item(s).

`hidden` boolean, The hidden status of the item.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`createdAt` string,

`updatedAt` string,

`workflows_post_executions` *Execute a workflow*

Description

Execute a workflow

Usage

```
workflows_post_executions(
  id,
  target_task = NULL,
  input = NULL,
  included_tasks = NULL
)
```

Arguments

id	integer required. The ID for the workflow.
target_task	string optional. For a reverse workflow, the name of the task to target.
input	list optional. Key-value pairs to send to this execution as inputs.
included_tasks	array optional. If specified, executes only the subset of workflow tasks included as specified by task name.

Value

A list containing the following elements:

id	integer, The ID for this workflow execution.
state	string, The state of this workflow execution.
mistralState	string, The state of this workflow as reported by mistral. One of running, paused, success, error, or cancelled
mistralStateInfo	string, The state info of this workflow as reported by mistral.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
definition	string, The definition of the workflow for this execution.
input	list, Key-value pairs defined for this execution.
includedTasks	array, The subset of workflow tasks selected to execute.
tasks	array, An array containing the following fields: • name string, The name of the task. • mistralState string, The state of this task. One of idle, waiting, running, delayed, success, error, or cancelled • mistralStateInfo string, Extra info associated with the state of the task. • runs array, The runs associated with this task, in descending order by id. • executions array, The executions run by this task, in descending order by id.
startedAt	string, The time this execution started.
finishedAt	string, The time this execution finished.
createdAt	string, The time this execution was created.
updatedAt	string, The time this execution was last updated.

`workflows_post_executions_cancel` *Cancel a workflow execution*

Description

Cancel a workflow execution

Usage

```
workflows_post_executions_cancel(id, execution_id)
```

Arguments

`id`	integer required. The ID for the workflow.
`execution_id`	integer required. The ID for the workflow execution.

Value

A list containing the following elements:

`id`	integer, The ID for this workflow execution.
`state`	string, The state of this workflow execution.
`mistralState`	string, The state of this workflow as reported by mistral. One of running, paused, success, error, or cancelled
`mistralStateInfo`	string, The state info of this workflow as reported by mistral.
`user`	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
`definition`	string, The definition of the workflow for this execution.
`input`	list, Key-value pairs defined for this execution.
`includedTasks`	array, The subset of workflow tasks selected to execute.
`tasks`	array, An array containing the following fields: • name string, The name of the task. • mistralState string, The state of this task. One of idle, waiting, running, delayed, success, error, or cancelled • mistralStateInfo string, Extra info associated with the state of the task. • runs array, The runs associated with this task, in descending order by id.

- executions array, The executions run by this task, in descending order by id.

startedAt	string, The time this execution started.
finishedAt	string, The time this execution finished.
createdAt	string, The time this execution was created.
updatedAt	string, The time this execution was last updated.

`workflows_post_executions_resume`
Resume a paused workflow execution

Description

Resume a paused workflow execution

Usage

```
workflows_post_executions_resume(id, execution_id)
```

Arguments

id	integer required. The ID for the workflow.
execution_id	integer required. The ID for the workflow execution.

Value

A list containing the following elements:

id	integer, The ID for this workflow execution.
state	string, The state of this workflow execution.
mistralState	string, The state of this workflow as reported by mistral. One of running, paused, success, error, or cancelled
mistralStateInfo	string, The state info of this workflow as reported by mistral.
user	list, A list containing the following elements: • id integer, The ID of this user. • name string, This user's name. • username string, This user's username. • initials string, This user's initials. • online boolean, Whether this user is online.
definition	string, The definition of the workflow for this execution.
input	list, Key-value pairs defined for this execution.
includedTasks	array, The subset of workflow tasks selected to execute.

tasks	array, An array containing the following fields: • name string, The name of the task. • mistralState string, The state of this task. One of idle, waiting, running, delayed, success, error, or cancelled • mistralStateInfo string, Extra info associated with the state of the task. • runs array, The runs associated with this task, in descending order by id. • executions array, The executions run by this task, in descending order by id.
startedAt	string, The time this execution started.
finishedAt	string, The time this execution finished.
createdAt	string, The time this execution was created.
updatedAt	string, The time this execution was last updated.

`workflows_post_executions_retry` *Retry a failed task, or all failed tasks in an execution*

Description

Retry a failed task, or all failed tasks in an execution

Usage

```
workflows_post_executions_retry(id, execution_id, task_name = NULL)
```

Arguments

id	integer required. The ID for the workflow.
execution_id	integer required. The ID for the workflow execution.
task_name	string optional. If specified, the name of the task to be retried. If not specified, all failed tasks in the execution will be retried.

Value

A list containing the following elements:

id	integer, The ID for this workflow execution.
state	string, The state of this workflow execution.
mistralState	string, The state of this workflow as reported by mistral. One of running, paused, success, error, or cancelled
mistralStateInfo	string, The state info of this workflow as reported by mistral.
user	list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

`definition`	string, The definition of the workflow for this execution.
`input`	list, Key-value pairs defined for this execution.
`includedTasks`	array, The subset of workflow tasks selected to execute.
`tasks`	array, An array containing the following fields: • name string, The name of the task. • mistralState string, The state of this task. One of idle, waiting, running, delayed, success, error, or cancelled • mistralStateInfo string, Extra info associated with the state of the task. • runs array, The runs associated with this task, in descending order by id. • executions array, The executions run by this task, in descending order by id.
`startedAt`	string, The time this execution started.
`finishedAt`	string, The time this execution finished.
`createdAt`	string, The time this execution was created.
`updatedAt`	string, The time this execution was last updated.

`workflows_post_git_checkout_latest`

Checkout latest commit on the current branch of a script or workflow

---

Description

Checkout latest commit on the current branch of a script or workflow

Usage

```
workflows_post_git_checkout_latest(id)
```

Arguments

`id`	integer required. The ID of the item.

Value

A list containing the following elements:

`content`	string, The file's contents.
`type`	string, The file's type.
`size`	integer, The file's size.
`fileHash`	string, The SHA of the file.

workflows_post_git_commits *Commit and push a new version of the file*

Description

Commit and push a new version of the file

Usage

```
workflows_post_git_commits(id, content, message, file_hash)
```

Arguments

id	integer required. The ID of the item.
content	string required. The contents to commit to the file.
message	string required. A commit message describing the changes being made.
file_hash	string required. The full SHA of the file being replaced.

Value

A list containing the following elements:

content	string, The file's contents.
type	string, The file's type.
size	integer, The file's size.
fileHash	string, The SHA of the file.

workflows_put *Replace all attributes of this Workflow*

Description

Replace all attributes of this Workflow

Usage

```
workflows_put(
  id,
  name,
  description = NULL,
  definition = NULL,
  schedule = NULL,
  allow_concurrent_executions = NULL,
  time_zone = NULL,
  notifications = NULL
)
```

Arguments

id	integer required. The ID for this workflow.
name	string required. The name of this workflow.
description	string optional. A description of the workflow.
definition	string optional. The definition of the workflow in YAML format. Must not be specified if 'fromJobChain' is specified.
schedule	list optional. A list containing the following elements: • scheduled boolean, If the item is scheduled. • scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth • scheduledHours array, Hours of the day it is scheduled on. • scheduledMinutes array, Minutes of the day it is scheduled on. • scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes. • scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.
allow_concurrent_executions	boolean optional. Whether the workflow can execute when already running.
time_zone	string optional. The time zone of this workflow.
notifications	list optional. A list containing the following elements: • urls array, URLs to receive a POST request at job completion • successEmailSubject string, Custom subject line for success e-mail. • successEmailBody string, Custom body text for success e-mail, written in Markdown. • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully. • failureEmailAddresses array, Addresses to notify by e-mail when the job fails. • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes. • successOn boolean, If success email notifications are on • failureOn boolean, If failure email notifications are on

Value

A list containing the following elements:

id	integer, The ID for this workflow.
name	string, The name of this workflow.
description	string, A description of the workflow.
definition	string, The definition of the workflow in YAML format. Must not be specified if 'fromJobChain' is specified.
valid	boolean, The validity of the workflow definition.

validationErrors
: string, The errors encountered when validating the workflow definition.

fileId
: string, The file id for the s3 file containing the workflow configuration.

user
: list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

state
: string, The state of the workflow. State is "running" if any execution is running, otherwise reflects most recent execution state.

lastExecutionId
: integer, The ID of the most recent execution of this workflow, if any.

schedule
: list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
  - scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

allowConcurrentExecutions
: boolean, Whether the workflow can execute when already running.

timeZone
: string, The time zone of this workflow.

nextExecutionAt
: string, The time of the next scheduled execution.

notifications
: list, A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  - successOn boolean, If success email notifications are on
  - failureOn boolean, If failure email notifications are on

archived
: string, The archival status of the requested item(s).

hidden
: boolean, The hidden status of the item.

myPermissionLevel
: string, Your permission level on the object. One of "read", "write", or "manage".

createdAt
: string,

updatedAt
: string,

`workflows_put_archive` *Update the archive status of this object*

Description

Update the archive status of this object

Usage

```
workflows_put_archive(id, status)
```

Arguments

id	integer required. The ID of the object.
status	boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id
: integer, The ID for this workflow.

name
: string, The name of this workflow.

description
: string, A description of the workflow.

definition
: string, The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain‘ is specified.

valid
: boolean, The validity of the workflow definition.

validationErrors
: string, The errors encountered when validating the workflow definition.

fileId
: string, The file id for the s3 file containing the workflow configuration.

user
: list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user’s name.
- username string, This user’s username.
- initials string, This user’s initials.
- online boolean, Whether this user is online.

state
: string, The state of the workflow. State is "running" if any execution is running, otherwise reflects most recent execution state.

lastExecutionId
: integer, The ID of the most recent execution of this workflow, if any.

`schedule` list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Days of the week, based on numeric value starting at 0 for Sunday. Mutually exclusive with scheduledDaysOfMonth
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Deprecated in favor of scheduled minutes.
- scheduledDaysOfMonth array, Days of the month it is scheduled on, mutually exclusive with scheduledDays.

`allowConcurrentExecutions` boolean, Whether the workflow can execute when already running.

`timeZone` string, The time zone of this workflow.

`nextExecutionAt` string, The time of the next scheduled execution.

`notifications` list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on
- failureOn boolean, If failure email notifications are on

`archived` string, The archival status of the requested item(s).

`hidden` boolean, The hidden status of the item.

`myPermissionLevel` string, Your permission level on the object. One of "read", "write", or "manage".

`createdAt` string,

`updatedAt` string,

`workflows_put_git` *Attach an item to a file in a git repo*

Description

Attach an item to a file in a git repo

Usage

```
workflows_put_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  git_ref_type = NULL,
  pull_from_git = NULL
)
```

Arguments

id	integer required. The ID of the item.
git_ref	string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch	string optional. The git branch that the file is on.
git_path	string optional. The path of the file in the repository.
git_repo_url	string optional. The URL of the git repository (e.g., https://github.com/organization/repo_name.git).
git_ref_type	string optional. Specifies if the file is versioned by branch or tag.
pull_from_git	boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef	string, A git reference specifying an unambiguous version of the file. Can be a branch name, tag or the full or shortened SHA of a commit.
gitBranch	string, The git branch that the file is on.
gitPath	string, The path of the file in the repository.
gitRepo	list, A list containing the following elements: • id integer, The ID for this git repository. • repoUrl string, The URL for this git repository. • createdAt string, • updatedAt string,
gitRefType	string, Specifies if the file is versioned by branch or tag.
pullFromGit	boolean, Automatically pull latest commit from git. Only works for scripts and workflows (assuming you have the feature enabled)

workflows_put_projects

Add a Workflow to a project

Description

Add a Workflow to a project

Usage

```
workflows_put_projects(id, project_id)
```

Arguments

id	integer required. The ID of the Workflow.
project_id	integer required. The ID of the project.

Value

An empty HTTP response

workflows_put_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
workflows_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
group_ids	array required. An array of one or more group IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

workflows_put_shares_users

Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

```
workflows_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_ids	array required. An array of one or more user IDs.
permission_level	string required. Options are: "read", "write", or "manage".
share_email_body	string optional. Custom body text for e-mail sent on a share.
send_shared_email	boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers	list, A list containing the following elements: • users array, • groups array,
writers	list, A list containing the following elements: • users array, • groups array,
owners	list, A list containing the following elements: • users array, • groups array,
totalUserShares	integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares	integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

workflows_put_transfer

Transfer ownership of this object to another user

Description

Transfer ownership of this object to another user

Usage

```
workflows_put_transfer(
  id,
  user_id,
  include_dependencies,
  email_body = NULL,
  send_email = NULL
)
```

Arguments

id	integer required. The ID of the resource that is shared.
user_id	integer required. ID of target user
include_dependencies	boolean required. Whether or not to give manage permissions on all dependencies
email_body	string optional. Custom body text for e-mail sent on transfer.
send_email	boolean optional. Send email to the target user of the transfer?

Value

A list containing the following elements:

dependencies — array, An array containing the following fields:

- objectType string, Dependent object type
- fcoType string, Human readable dependent object type
- id integer, Dependent object ID
- name string, Dependent object name, or nil if the requesting user cannot read this object
- permissionLevel string, Permission level of target user (not user's groups) for dependent object. Null if no target user or not shareable (e.g. a database table).
- description string, Additional information about the dependency, if relevant
- shared boolean, Whether dependent object was successfully shared with target user

---

write_chunks *Split a file into chunks of a given chunk size, returning a list of file names.*

Description

Split a file into chunks of a given chunk size, returning a list of file names.

Usage

```
write_chunks(file, chunk_size)
```

Arguments

file	name of the file
chunk_size	size of the chunk in bytes.

write_civis *Upload a local data frame or csv file to the Civis Platform (Redshift)*

Description

Uploads a data frame, a csv file, or file on S3 to Redshift based on the first argument.

A default database can be set using `options(civis.default_db = "my_database")`. If there is only one database available, this database will automatically be used as the default.

Usage

```
write_civis(x, ...)

## S3 method for class 'data.frame'
write_civis(
  x,
  tablename,
  database = NULL,
  if_exists = "fail",
  distkey = NULL,
  sortkey1 = NULL,
  sortkey2 = NULL,
  max_errors = NULL,
  verbose = FALSE,
  hidden = TRUE,
  diststyle = NULL,
  header = TRUE,
  credential_id = NULL,
  import_args = NULL,
  ...
)

## S3 method for class 'character'
write_civis(
  x,
  tablename,
  database = NULL,
  if_exists = "fail",
  distkey = NULL,
  sortkey1 = NULL,
  sortkey2 = NULL,
  max_errors = NULL,
  verbose = FALSE,
  hidden = TRUE,
  diststyle = NULL,
  header = TRUE,
```

```
  credential_id = NULL,
  import_args = NULL,
  ...
)

## S3 method for class 'numeric'
write_civis(
  x,
  tablename,
  database = NULL,
  if_exists = "fail",
  distkey = NULL,
  sortkey1 = NULL,
  sortkey2 = NULL,
  max_errors = NULL,
  verbose = FALSE,
  delimiter = ",",
  hidden = TRUE,
  diststyle = NULL,
  header = TRUE,
  credential_id = NULL,
  import_args = NULL,
  ...
)
```

Arguments

x	data frame, file path of a csv, or the id of a csv file on S3 to upload to platform.
...	arguments passed to `write.csv`.
tablename	string, Name of table and schema `"schema.tablename"`.
database	string, Name of database where data frame is to be uploaded. If no database is specified, uses `options(civis.default_db)`.
if_exists	string, optional, String indicating action to take if table already exists. Must be either "fail", "drop", "truncate" or "append". Defaults to "fail".
distkey	string, optional, Column name designating the distkey.
sortkey1	string, optional, Column name designating the first sortkey.
sortkey2	string, optional, Column name designating the second (compound) sortkey.
max_errors	int, optional, Maximum number of rows with errors to remove before failing.
verbose	bool, Set to TRUE to print intermediate progress indicators.
hidden	bool, if `TRUE` (default), this job will not appear in the Civis UI.
diststyle	string optional. The diststyle to use for the table. One of "even", "all", or "key".
header	bool, if `TRUE` (default) the first row is a header.
credential_id	integer, the id of the credential to be used when performing the database import. If `NULL` (default), the default credential of the current user will be used.
import_args	list of additional arguments for `imports_post_files`.
delimiter	string, optional. Which delimiter to use. One of `','`, `'\t'` or `'|'`.

Methods (by class)

- `write_civis(data.frame)`: Upload a data frame to Civis Platform (Redshift).
- `write_civis(character)`: Upload a csv to Civis Platform (Redshift).
- `write_civis(numeric)`: Upload a csv file from the files endpoint to Civis Platform (Redshift)

See Also

`refresh_table` to update table meta-data.

Other io: `download_civis()`, `query_civis()`, `query_civis_file()`, `read_civis()`, `write_civis_file()`

Examples

```
## Not run:
df <- read.csv(local_file)

# Create new table, fail if already exists
write_civis(df, "schema.my_table", "my_database")

# Create new table, append if already exists
write_civis(df, "schema.my_table", "my_database", if_exists="append")

# Create new table with additional options
write_civis(df, "schema.my_table", "my_database",
            distkey="id",
            sortkey1="added_date",
            credential_id = 1,
            header = FALSE)

# Create new table directly from a saved csv
write_civis("my/file/path.csv", "schema.my_table", "my_database")

# Create new table from a file_id
id <- write_civis_file("my/file/path.csv", name = "path.csv")
write_civis(id, "schema.my_table", "my_database")


## End(Not run)
```

`write_civis_file` *Upload a R object or file to Civis Platform (Files endpoint)*

Description

Uploads a data frame, R object or file to the files endpoint on Civis Platform (Amazon S3). It returns the id of the file for use with `read_civis` or `download_civis`.

Data frames are uploaded as CSVs with `write.csv`. R objects are serialized with `saveRDS`. Files are uploaded as-is. Objects or files larger than 50mb are chunked and can be uploaded in parallel if a `plan` has been set. Files larger than 5TB cannot be uploaded.

Usage

```
write_civis_file(x, ...)

## Default S3 method:
write_civis_file(x, name = "r-object.rds", expires_at = NULL, ...)

## S3 method for class 'data.frame'
write_civis_file(
  x,
  name = "data.csv",
  expires_at = NULL,
  row.names = FALSE,
  ...
)

## S3 method for class 'character'
write_civis_file(x, name = x, expires_at = NULL, ...)
```

Arguments

x	R object or path of file to upload.
...	arguments passed to saveRDS or write.csv for data frames.
name	string, Name of the file or object.
expires_at	string, The date and time the object will expire on in the format "YYYY-MM-DD HH:MM:SS". expires_at = NULL allows files to be kept indefinitely.
row.names	default FALSE. Either a logical value indicating whether the row names of x are to be written along with x, or a character vector of row names to be written.

Details

Data frames are uploaded as CSVs using write.csv, with row.names = FALSE by default. Additional arguments to write.csv can be passed through ....

By default, R objects are serialized using saveRDS before uploading the object to the files endpoint. If given a filepath, the file is uploaded as-is.

Value

The file id which can be used to later retrieve the file using read_civis.

Methods (by class)

- write_civis_file(default): Serialize R object
- write_civis_file(data.frame): Upload a data frame as a csv
- write_civis_file(character): Upload any file

See Also

Other io: download_civis(), query_civis(), query_civis_file(), read_civis(), write_civis()

Examples

```
## Not run:
data(iris)
file_id <- write_civis_file(iris)
read_civis(file_id)

file_id <- write_civis_file("path/to/my.csv")
read_civis(file_id)
read_civis(file_id, using = readr::read_csv)

file_id <- write_civis_file(list(a = 1))
read_civis(file_id, using = readRDS)

# Does not expire
file_id <- write_civis_file(iris, expires_at = NULL)

# Expires on a given date and time
file_id <- write_civis_file(iris, expires_at = "2030-01-01")
file_id <- write_civis_file(iris, expires_at = "12:00:00")
file_id <- write_civis_file(iris, expires_at = "2030-01-01 12:00:00")

# Upload a large file in parallel.
library(future)
plan(multisession)
file_id <- write_civis_file("my_large_file")

## End(Not run)
```

---

write_job_output *Add a file as a run output if called from a container job*

Description

Add a file as a run output if called from a container job

Usage

```
write_job_output(filename)
```

Arguments

filename	string, name of the file to add as a run output `civis::scripts_post_*_runs_outputs`.

Details

Only posts if running on Civis Platform.

Value

Returns the filename if not running on platform.

Index

∗ **datasets**
CIVIS_ML_CLASSIFIERS, 47
CIVIS_ML_REGRESSORS, 74
∗ **id**
get_database_id, 320
∗ **io**
download_civis, 144
query_civis, 614
query_civis_file, 616
read_civis, 617
write_civis, 1205
write_civis_file, 1207
∗ **pagination**
fetch_all, 295
fetch_until, 299
∗ **script_utils**
civis_script, 86
fetch_output_file_ids, 297
run_civis, 683
run_template, 684
∗ **tables**
get_table_id, 325
refresh_table, 619
transfer_table, 1132

admin_list_organizations, 22
aliases_delete, 23
aliases_delete_shares_groups, 24
aliases_delete_shares_users, 24
aliases_get, 25
aliases_get_object_type, 25
aliases_list, 26
aliases_list_dependencies, 27
aliases_list_shares, 28
aliases_patch, 29
aliases_post, 30
aliases_put, 31
aliases_put_shares_groups, 32
aliases_put_shares_users, 33
aliases_put_transfer, 34
announcements_list, 35
await, 36
await_all *(*await*)*, 36

cancel, 38
cancel.CivisFuture *(*CivisFuture*)*, 38
civis_file, 40, *47*
civis_file_manifest, 41, *46, 47*
civis_ml, *40, 41*, 41, *87*, *322, 323*
CIVIS_ML_CLASSIFIERS, 47
civis_ml_extra_trees_classifier, *45*, 48
civis_ml_extra_trees_regressor, *45*, 52
civis_ml_fetch_existing *(*civis_ml*)*, 41
civis_ml_gradient_boosting_classifier, *45*, 56
civis_ml_gradient_boosting_regressor, *45*, 61
civis_ml_random_forest_classifier, *45*, 65
civis_ml_random_forest_regressor, *45*, 70
CIVIS_ML_REGRESSORS, 74
civis_ml_sparse_linear_regressor, *45*, 74
civis_ml_sparse_logistic, *45*, 77
civis_ml_sparse_ridge_regressor, *45*, 82
civis_platform, 85, *683*
civis_script, 86, *298*, *617*, *683, 684*
civis_table, *47*, 87
civis_to_multifile_csv, *46*, 88, *145*
CivisFuture, 38, *85*, *683*
clusters_delete_kubernetes_partitions, 89
clusters_get_kubernetes, 90
clusters_get_kubernetes_instance_configs, 91
clusters_get_kubernetes_partitions, 92
clusters_list_kubernetes, 93
clusters_list_kubernetes_compute_hours, 95

clusters_list_kubernetes_compute_metrics, 95
clusters_list_kubernetes_deployment_stats, 98
clusters_list_kubernetes_deployments, 96
clusters_list_kubernetes_instance_configs_active_workloads, 98
clusters_list_kubernetes_instance_configs_historical_graphs, 99
clusters_list_kubernetes_instance_configs_historical_metrics, 100
clusters_list_kubernetes_instance_configs_user_statistics, 101
clusters_list_kubernetes_partitions, 102
clusters_patch_kubernetes_partitions, 103
clusters_post_kubernetes_partitions, 105
coef.civis_ml, 106
credentials_delete, 107
credentials_delete_shares_groups, 107
credentials_delete_shares_users, 108
credentials_get, 108
credentials_list, 109
credentials_list_dependencies, 111
credentials_list_shares, 112
credentials_list_types, 113
credentials_patch, 113
credentials_post, 115
credentials_post_authenticate, 117
credentials_post_temporary, 118
credentials_put, 119
credentials_put_shares_groups, 120
credentials_put_shares_users, 121
credentials_put_transfer, 122

databases_delete_schemas_tables_projects, 123
databases_get, 124
databases_get_schema_privileges, 128
databases_get_schemas_tables, 125
databases_get_status_graphs_timeframe, 128
databases_get_table_privilegesschema_name, 129
databases_get_whitelist_ips, 130
databases_list, 130
databases_list_advanced_settings, 131
databases_list_groups, 132
databases_list_schemas, 132
databases_list_schemas_tables, 133
databases_list_schemas_tables_projects, 133
databases_list_tables, 134
databases_list_tables_search, 136
databases_list_users, 137
databases_list_whitelist_ips, 138
databases_patch_advanced_settings, 138
databases_patch_schemas_tables, 139
databases_post_schemas_scan, 142
databases_put_advanced_settings, 143
databases_put_schemas_tables_projects, 143
default_credential, 144
download_civis, 144, *615–617*, *619*, *1207*, *1208*

endpoints_list, 146
enhancements_delete_cass_ncoa_projects, 147
enhancements_delete_cass_ncoa_runs, 147
enhancements_delete_cass_ncoa_shares_groups, 148
enhancements_delete_cass_ncoa_shares_users, 148
enhancements_delete_civis_data_match_projects, 149
enhancements_delete_civis_data_match_runs, 149
enhancements_delete_civis_data_match_shares_groups, 150
enhancements_delete_civis_data_match_shares_users, 150
enhancements_delete_geocode_projects, 151
enhancements_delete_geocode_runs, 151
enhancements_delete_geocode_shares_groups, 152
enhancements_delete_geocode_shares_users, 152
enhancements_delete_identity_resolution_projects, 153
enhancements_delete_identity_resolution_runs, 153

enhancements_delete_identity_resolution_shares_groups, 154
enhancements_delete_identity_resolution_shares_users, 154
enhancements_get_cass_ncoa, 155
enhancements_get_cass_ncoa_runs, 157
enhancements_get_civis_data_match, 158
enhancements_get_civis_data_match_runs, 160
enhancements_get_geocode, 161
enhancements_get_geocode_runs, 163
enhancements_get_identity_resolution, 164
enhancements_get_identity_resolution_runs, 167
enhancements_list, 168
enhancements_list_cass_ncoa_dependencies, 169
enhancements_list_cass_ncoa_projects, 170
enhancements_list_cass_ncoa_runs, 171
enhancements_list_cass_ncoa_runs_logs, 172
enhancements_list_cass_ncoa_runs_outputs, 173
enhancements_list_cass_ncoa_shares, 174
enhancements_list_civis_data_match_dependencies, 175
enhancements_list_civis_data_match_projects, 175
enhancements_list_civis_data_match_runs, 176
enhancements_list_civis_data_match_runs_logs, 177
enhancements_list_civis_data_match_runs_outputs, 178
enhancements_list_civis_data_match_shares, 179
enhancements_list_field_mapping, 180
enhancements_list_geocode_dependencies, 180
enhancements_list_geocode_projects, 181
enhancements_list_geocode_runs, 182
enhancements_list_geocode_runs_logs, 183
enhancements_list_geocode_runs_outputs, 184
enhancements_list_geocode_shares, 185
enhancements_list_identity_resolution, 186
enhancements_list_identity_resolution_dependencies, 187
enhancements_list_identity_resolution_projects, 188
enhancements_list_identity_resolution_runs, 189
enhancements_list_identity_resolution_runs_logs, 191
enhancements_list_identity_resolution_shares, 192
enhancements_list_types, 193
enhancements_patch_cass_ncoa, 193
enhancements_patch_civis_data_match, 197
enhancements_patch_geocode, 201
enhancements_patch_identity_resolution, 205
enhancements_post_cass_ncoa, 210
enhancements_post_cass_ncoa_cancel, 214
enhancements_post_cass_ncoa_runs, 215
enhancements_post_civis_data_match, 215
enhancements_post_civis_data_match_cancel, 219
enhancements_post_civis_data_match_clone, 220
enhancements_post_civis_data_match_runs, 222
enhancements_post_geocode, 223
enhancements_post_geocode_cancel, 226
enhancements_post_geocode_runs, 227
enhancements_post_identity_resolution, 227
enhancements_post_identity_resolution_cancel, 232
enhancements_post_identity_resolution_clone, 233
enhancements_post_identity_resolution_runs, 236
enhancements_put_cass_ncoa, 238
enhancements_put_cass_ncoa_archive, 242
enhancements_put_cass_ncoa_projects,

245
enhancements_put_cass_ncoa_shares_groups, 245
enhancements_put_cass_ncoa_shares_users, 246
enhancements_put_cass_ncoa_transfer, 247
enhancements_put_civis_data_match, 248
enhancements_put_civis_data_match_archive, 252
enhancements_put_civis_data_match_projects, 254
enhancements_put_civis_data_match_shares_groups, 255
enhancements_put_civis_data_match_shares_users, 256
enhancements_put_civis_data_match_transfer, 257
enhancements_put_geocode, 258
enhancements_put_geocode_archive, 261
enhancements_put_geocode_projects, 263
enhancements_put_geocode_shares_groups, 264
enhancements_put_geocode_shares_users, 265
enhancements_put_geocode_transfer, 266
enhancements_put_identity_resolution, 267
enhancements_put_identity_resolution_archive, 272
enhancements_put_identity_resolution_projects, 275
enhancements_put_identity_resolution_shares_groups, 276
enhancements_put_identity_resolution_shares_users, 277
enhancements_put_identity_resolution_transfer, 278
environment, *39*
exports_delete_files_csv_runs, 279
exports_get_files_csv, 279
exports_get_files_csv_runs, 281
exports_list, 281
exports_list_files_csv_runs, 283
exports_list_files_csv_runs_logs, 284
exports_list_files_csv_runs_outputs, 285
exports_patch_files_csv, 286
exports_post_files_csv, 288
exports_post_files_csv_runs, 290
exports_put_files_csv, 291
exports_put_files_csv_archive, 293
expression, *39*

fetch_all, 295, *299*
fetch_and_generate_client, 295
fetch_logs, *37*, *47*, 296
fetch_logs.CivisFuture *(*CivisFuture*)*, 38
fetch_oos_scores, *46*, *47*, 297
fetch_output *(*fetch_output_file_ids*)*, 297
fetch_output_file_ids, *87*, 297, *683*, *684*
fetch_predictions, *47*, 298
fetch_until, *295*, 299
files_delete_preprocess_csv, 300
files_delete_projects, 300
files_delete_shares_groups, 301
files_delete_shares_users, 301
files_get, 302
files_get_preprocess_csv, 303
files_list_dependencies, 304
files_list_projects, 304
files_list_shares, 305
files_patch, 306
files_patch_preprocess_csv, 307
files_post, 309
files_post_multipart, 310
files_post_multipart_complete, 311
files_post_preprocess_csv, 311
files_put, 313
files_put_preprocess_csv, 314
files_put_preprocess_csv_archive, 315
files_put_projects, 316
files_put_shares_groups, 317
files_put_shares_users, 318
files_put_transfer, 319
Future, *40*
future, *85*
future(), *39*

generate_client, 320
get_database_id, 320
get_error, *37*, 321
get_feature_importance, 322
get_job_type_version, 322
get_metric, *47*, 323
get_script_fun, 323

get_status, *37*, 324
get_table_id, 325, *620*, *1133*
get_template_ids_all_versions, 325
get_train_template_id, 326
git_repos_delete, 326
git_repos_get, 327
git_repos_list, 327
git_repos_list_refs, 328
git_repos_post, 328
groups_delete, 329
groups_delete_members, 329
groups_delete_shares_groups, 330
groups_delete_shares_users, 330
groups_get, 331
groups_list, 332
groups_list_child_groups, 333
groups_list_shares, 334
groups_patch, 335
groups_post, 337
groups_put, 339
groups_put_members, 341
groups_put_shares_groups, 342
groups_put_shares_users, 343

hist.civis_ml, 344

imports_delete_files_csv, 345
imports_delete_files_csv_runs, 345
imports_delete_files_runs, 346
imports_delete_projects, 346
imports_delete_shares_groups, 347
imports_delete_shares_users, 347
imports_delete_syncs, 348
imports_get, 348
imports_get_batches, 351
imports_get_files_csv, 351
imports_get_files_csv_runs, 353
imports_get_files_runs, 354
imports_list, 355
imports_list_batches, 357
imports_list_dependencies, 358
imports_list_files_csv_runs, 359
imports_list_files_csv_runs_logs, 360
imports_list_files_runs, 361
imports_list_files_runs_logs, 362
imports_list_projects, 362
imports_list_runs, 363
imports_list_runs_logs, 364
imports_list_shares, 365
imports_patch_files_csv, 366
imports_post, 369
imports_post_batches, 373
imports_post_cancel, 374
imports_post_files, 375, *1206*
imports_post_files_csv, 376
imports_post_files_csv_runs, 380
imports_post_files_runs, 381
imports_post_runs, 381
imports_post_syncs, 382, *1133*
imports_put, 387
imports_put_archive, 391
imports_put_files_csv, 393
imports_put_files_csv_archive, 397
imports_put_projects, 399
imports_put_shares_groups, 399
imports_put_shares_users, 401
imports_put_syncs, 402
imports_put_syncs_archive, 407
imports_put_transfer, 410

jobs_delete_projects, 411
jobs_delete_runs, 411
jobs_delete_shares_groups, 412
jobs_delete_shares_users, 412
jobs_get, 413
jobs_get_runs, 414
jobs_list, 415
jobs_list_children, 417
jobs_list_dependencies, 418
jobs_list_parents, 419
jobs_list_projects, 420
jobs_list_runs, 421
jobs_list_runs_logs, 422
jobs_list_runs_outputs, 423
jobs_list_shares, 424
jobs_list_workflows, 425
jobs_post_runs, 426
jobs_post_trigger_email, 427
jobs_put_archive, 427
jobs_put_projects, 429
jobs_put_shares_groups, 429
jobs_put_shares_users, 430
jobs_put_transfer, 431
json_values_delete_shares_groups, 432
json_values_delete_shares_users, 433
json_values_get, 433
json_values_list_dependencies, 434
json_values_list_shares, 434

json_values_patch, 435
json_values_post, 436
json_values_put_shares_groups, 436
json_values_put_shares_users, 438
json_values_put_transfer, 439

match_targets_delete_shares_groups, 440
match_targets_delete_shares_users, 440
match_targets_get, 441
match_targets_list, 441
match_targets_list_shares, 442
match_targets_patch, 443
match_targets_post, 443
match_targets_put_archive, 444
match_targets_put_shares_groups, 445
match_targets_put_shares_users, 446
media_delete_optimizations_runs, 447
media_delete_optimizations_shares_groups, 447
media_delete_optimizations_shares_users, 448
media_delete_ratecards_shares_groups, 448
media_delete_ratecards_shares_users, 449
media_delete_spot_orders_shares_groups, 449
media_delete_spot_orders_shares_users, 450
media_get_optimizations, 450
media_get_optimizations_runs, 452
media_get_ratecards, 452
media_get_spot_orders, 453
media_list_dmas, 454
media_list_optimizations, 454
media_list_optimizations_runs, 455
media_list_optimizations_runs_logs, 456
media_list_optimizations_shares, 457
media_list_ratecards, 458
media_list_ratecards_shares, 459
media_list_spot_orders, 460
media_list_spot_orders_shares, 460
media_list_targets, 461
media_patch_optimizations, 462
media_patch_ratecards, 464
media_post_optimizations, 465
media_post_optimizations_clone, 467
media_post_optimizations_runs, 469
media_post_ratecards, 470
media_post_spot_orders, 470
media_put_optimizations_archive, 471
media_put_optimizations_shares_groups, 473
media_put_optimizations_shares_users, 474
media_put_ratecards, 475
media_put_ratecards_archive, 476
media_put_ratecards_shares_groups, 476
media_put_ratecards_shares_users, 477
media_put_spot_orders, 478
media_put_spot_orders_archive, 479
media_put_spot_orders_shares_groups, 480
media_put_spot_orders_shares_users, 481
models_delete_builds, 482
models_delete_projects, 482
models_delete_shares_groups, 483
models_delete_shares_users, 483
models_get, 484
models_get_builds, 487
models_list, 488
models_list_builds, 491
models_list_builds_logs, 492
models_list_dependencies, 493
models_list_projects, 494
models_list_schedules, 495
models_list_shares, 495
models_list_types, 496
models_put_archive, 497
models_put_projects, 500
models_put_shares_groups, 500
models_put_shares_users, 501
models_put_transfer, 502
multipart_upload, 503

notebooks_delete, 504
notebooks_delete_deployments, 504
notebooks_delete_projects, 505
notebooks_delete_shares_groups, 505
notebooks_delete_shares_users, 506
notebooks_get, 506
notebooks_get_deployments, 508
notebooks_get_git_commits, 509
notebooks_list, 510
notebooks_list_dependencies, 511

notebooks_list_deployments, 512
notebooks_list_deployments_logs, 514
notebooks_list_git, 515
notebooks_list_git_commits, 515
notebooks_list_projects, 516
notebooks_list_shares, 517
notebooks_list_update_links, 518
notebooks_patch, 518
notebooks_patch_git, 521
notebooks_post, 522
notebooks_post_clone, 525
notebooks_post_deployments, 527
notebooks_post_git_commits, 528
notebooks_put, 529
notebooks_put_archive, 532
notebooks_put_git, 534
notebooks_put_projects, 535
notebooks_put_shares_groups, 536
notebooks_put_shares_users, 537
notebooks_put_transfer, 538
notifications_list, 539

ontology_list, 539
organizations_delete_favorites, 540
organizations_list_favorites, 540
organizations_post_favorites, 542

permission_sets_delete_resources, 543
permission_sets_delete_resources_shares_groups, 543
permission_sets_delete_resources_shares_users, 544
permission_sets_delete_shares_groups, 544
permission_sets_delete_shares_users, 545
permission_sets_get, 545
permission_sets_get_resources, 546
permission_sets_list, 547
permission_sets_list_dependencies, 548
permission_sets_list_resources, 549
permission_sets_list_resources_shares, 550
permission_sets_list_shares, 551
permission_sets_list_users_permissions, 552
permission_sets_patch, 552
permission_sets_patch_resources, 553
permission_sets_post, 554
permission_sets_post_resources, 555
permission_sets_put, 555
permission_sets_put_archive, 556
permission_sets_put_resources_shares_groups, 557
permission_sets_put_resources_shares_users, 558
permission_sets_put_shares_groups, 559
permission_sets_put_shares_users, 560
permission_sets_put_transfer, 561
plan, *85*, *1207*
plot.civis_ml_classifier, 562
plot.civis_ml_regressor, 563
predict.civis_ml *(*civis_ml*)*, 41
predictions_get, 563
predictions_list, 565
predictions_list_schedules, 566
print.civis_api, 566
projects_delete, 567
projects_delete_parent_projects, 568
projects_delete_shares_groups, 568
projects_delete_shares_users, 569
projects_get, 569
projects_list, 573
projects_list_dependencies, 575
projects_list_parent_projects, 575
projects_list_shares, 576
projects_post, 577
projects_post_clone, 581
projects_put, 585
projects_put_archive, 589
projects_put_auto_share, 593
projects_put_parent_projects, 597
projects_put_shares_groups, 598
projects_put_shares_users, 599
projects_put_transfer, 600
publish_addin, 601
publish_html, 601, *603*
publish_rmd, *601, 602*, 602

queries_delete, 604
queries_delete_runs, 605
queries_get, 605
queries_get_runs, 606
queries_list, 607
queries_list_runs, 609
queries_list_runs_logs, 610
queries_post, 611
queries_post_runs, 613

queries_put_scripts, 613
query_civis, *146*, 614, *617*, *619*, *1207, 1208*
query_civis_file, *146*, *615*, 616, *619*, *1207, 1208*

read_civis, *87*, *146*, *615–617*, 617, *1207, 1208*
refresh_table, *325*, 619, *1133*, *1207*
remote_hosts_delete_shares_groups, 620
remote_hosts_delete_shares_users, 621
remote_hosts_get, 621
remote_hosts_list, 622
remote_hosts_list_data_sets, 623
remote_hosts_list_shares, 624
remote_hosts_patch, 625
remote_hosts_post, 626
remote_hosts_post_authenticate, 627
remote_hosts_put, 628
remote_hosts_put_shares_groups, 629
remote_hosts_put_shares_users, 630
reports_delete_grants, 631
reports_delete_projects, 631
reports_delete_services_projects, 632
reports_delete_services_shares_groups, 632
reports_delete_services_shares_users, 633
reports_delete_shares_groups, 633
reports_delete_shares_users, 634
reports_delete_sql_projects, 634
reports_delete_sql_shares_groups, 635
reports_delete_sql_shares_users, 635
reports_get, 636
reports_get_git_commits, 637
reports_get_services, 638
reports_get_sql, 639
reports_list, 640
reports_list_dependencies, 642
reports_list_git, 643
reports_list_git_commits, 644
reports_list_projects, 644
reports_list_services_dependencies, 645
reports_list_services_projects, 646
reports_list_services_shares, 647
reports_list_shares, 648
reports_list_sql_dependencies, 649
reports_list_sql_projects, 649
reports_list_sql_shares, 650
reports_patch, 651
reports_patch_git, 653
reports_patch_services, 655
reports_patch_sql, 656
reports_post, 657
reports_post_git_commits, 659
reports_post_grants, 660
reports_post_refresh, 662
reports_post_services, 662
reports_post_sql, 663
reports_post_sql_refresh, 665
reports_put_archive, 666
reports_put_git, 668
reports_put_projects, 669
reports_put_services_archive, 669
reports_put_services_projects, 670
reports_put_services_shares_groups, 671
reports_put_services_shares_users, 672
reports_put_services_transfer, 673
reports_put_shares_groups, 674
reports_put_shares_users, 675
reports_put_sql_archive, 676
reports_put_sql_projects, 677
reports_put_sql_shares_groups, 678
reports_put_sql_shares_users, 679
reports_put_sql_transfer, 680
reports_put_transfer, 681
resolved, 682
resolved.CivisFuture *(*CivisFuture*)*, 38
result.CivisFuture *(*CivisFuture*)*, 38
roles_list, 682
run.CivisFuture *(*CivisFuture*)*, 38
run_civis, *87*, *298*, 683, *684*
run_template, *87*, *298*, *683*, 684

saveRDS, *1207, 1208*
scripts_delete, 685
scripts_delete_containers, 685
scripts_delete_containers_projects, 686
scripts_delete_containers_runs, 686
scripts_delete_containers_shares_groups, 687
scripts_delete_containers_shares_users, 687
scripts_delete_custom, 688
scripts_delete_custom_projects, 688
scripts_delete_custom_runs, 689

scripts_delete_custom_shares_groups, 689
scripts_delete_custom_shares_users, 690
scripts_delete_dbt, 690
scripts_delete_dbt_projects, 691
scripts_delete_dbt_runs, 691
scripts_delete_dbt_shares_groups, 692
scripts_delete_dbt_shares_users, 692
scripts_delete_javascript, 693
scripts_delete_javascript_projects, 693
scripts_delete_javascript_runs, 694
scripts_delete_javascript_shares_groups, 694
scripts_delete_javascript_shares_users, 695
scripts_delete_python3, 695
scripts_delete_python3_projects, 696
scripts_delete_python3_runs, 696
scripts_delete_python3_shares_groups, 697
scripts_delete_python3_shares_users, 697
scripts_delete_r, 698
scripts_delete_r_projects, 698
scripts_delete_r_runs, 699
scripts_delete_r_shares_groups, 699
scripts_delete_r_shares_users, 700
scripts_delete_sql, 700
scripts_delete_sql_projects, 701
scripts_delete_sql_runs, 701
scripts_delete_sql_shares_groups, 702
scripts_delete_sql_shares_users, 702
scripts_get, 703
scripts_get_containers, 706
scripts_get_containers_runs, 709
scripts_get_custom, *44*, *51*, *55*, *59*, *64*, *68*, *73*, *76*, *80*, *84*, 710
scripts_get_custom_runs, *44*, *51*, *55*, *59*, *64*, *68*, *73*, *76*, *80*, *84*, 714
scripts_get_dbt, 715
scripts_get_dbt_runs, 719
scripts_get_javascript, 720
scripts_get_javascript_git_commits, 723
scripts_get_javascript_runs, 723
scripts_get_python3, 724
scripts_get_python3_git_commits, 727
scripts_get_python3_runs, 728
scripts_get_r, 729
scripts_get_r_git_commits, 732
scripts_get_r_runs, 733
scripts_get_sql, 734
scripts_get_sql_git_commits, 737
scripts_get_sql_runs, 738
scripts_list, 739
scripts_list_containers_dependencies, 741
scripts_list_containers_projects, 741
scripts_list_containers_runs, 742
scripts_list_containers_runs_logs, *324*, 743
scripts_list_containers_runs_outputs, 744
scripts_list_containers_shares, 745
scripts_list_custom, 746
scripts_list_custom_dependencies, 748
scripts_list_custom_projects, 748
scripts_list_custom_runs, 749
scripts_list_custom_runs_logs, 750
scripts_list_custom_runs_outputs, *44*, *51*, *55*, *59*, *64*, *68*, *73*, *76*, *80*, *84*, 751
scripts_list_custom_shares, 752
scripts_list_dbt_dependencies, 753
scripts_list_dbt_projects, 754
scripts_list_dbt_runs, 755
scripts_list_dbt_runs_logs, 756
scripts_list_dbt_runs_outputs, 757
scripts_list_dbt_shares, 758
scripts_list_history, 759
scripts_list_javascript_dependencies, 759
scripts_list_javascript_git, 760
scripts_list_javascript_git_commits, 761
scripts_list_javascript_projects, 762
scripts_list_javascript_runs, 763
scripts_list_javascript_runs_logs, 764
scripts_list_javascript_runs_outputs, 764
scripts_list_javascript_shares, 765
scripts_list_python3_dependencies, 766
scripts_list_python3_git, 767
scripts_list_python3_git_commits, 768
scripts_list_python3_projects, 768

scripts_list_python3_runs, 769
scripts_list_python3_runs_logs, 770
scripts_list_python3_runs_outputs, 771
scripts_list_python3_shares, 772
scripts_list_r_dependencies, 773
scripts_list_r_git, 774
scripts_list_r_git_commits, 774
scripts_list_r_projects, 775
scripts_list_r_runs, 776
scripts_list_r_runs_logs, 777
scripts_list_r_runs_outputs, 778
scripts_list_r_shares, 779
scripts_list_sql_dependencies, 780
scripts_list_sql_git, 780
scripts_list_sql_git_commits, 781
scripts_list_sql_projects, 782
scripts_list_sql_runs, 783
scripts_list_sql_runs_logs, 784
scripts_list_sql_runs_outputs, 785
scripts_list_sql_shares, 786
scripts_list_types, 787
scripts_patch_container_runs, 793
scripts_patch_containers, 787
scripts_patch_custom, 794
scripts_patch_dbt, 799
scripts_patch_dbt_runs, 805
scripts_patch_javascript, 806
scripts_patch_javascript_git, 811
scripts_patch_javascript_runs, 812
scripts_patch_python3, 812
scripts_patch_python3_git, 818
scripts_patch_python3_runs, 819
scripts_patch_r, 819
scripts_patch_r_git, 825
scripts_patch_r_runs, 826
scripts_patch_sql, 826
scripts_patch_sql_git, 832
scripts_patch_sql_runs, 833
scripts_post, 833
scripts_post_cancel, 837
scripts_post_containers, *40*, *85*, 838
scripts_post_containers_clone, 844
scripts_post_containers_runs, 848
scripts_post_containers_runs_logs, 849
scripts_post_containers_runs_outputs, 850
scripts_post_custom, *43*, *50*, *55*, *59*, *64*, *68*, *72*, *76*, *80*, *84*, 850
scripts_post_custom_clone, 855
scripts_post_custom_runs, 859
scripts_post_custom_runs_outputs, 860
scripts_post_dbt, 861
scripts_post_dbt_clone, 868
scripts_post_dbt_runs, 872
scripts_post_dbt_runs_outputs, 873
scripts_post_javascript, 873
scripts_post_javascript_clone, 878
scripts_post_javascript_git_checkout, 881
scripts_post_javascript_git_checkout_latest, 882
scripts_post_javascript_git_commits, 883
scripts_post_javascript_runs, 883
scripts_post_javascript_runs_outputs, 884
scripts_post_python3, 885
scripts_post_python3_clone, 890
scripts_post_python3_git_checkout, 894
scripts_post_python3_git_checkout_latest, 894
scripts_post_python3_git_commits, 895
scripts_post_python3_runs, 896
scripts_post_python3_runs_outputs, 897
scripts_post_r, 897
scripts_post_r_clone, 903
scripts_post_r_git_checkout, 907
scripts_post_r_git_checkout_latest, 908
scripts_post_r_git_commits, 908
scripts_post_r_runs, 909
scripts_post_r_runs_outputs, 910
scripts_post_run, 903
scripts_post_sql, *616*, *617*, 910
scripts_post_sql_clone, 916
scripts_post_sql_git_checkout, 919
scripts_post_sql_git_checkout_latest, 920
scripts_post_sql_git_commits, 921
scripts_post_sql_runs, 921
scripts_put_containers, 922
scripts_put_containers_archive, 928
scripts_put_containers_projects, 932
scripts_put_containers_shares_groups, 933
scripts_put_containers_shares_users,

934
scripts_put_containers_transfer, 935
scripts_put_custom, 936
scripts_put_custom_archive, 941
scripts_put_custom_projects, 944
scripts_put_custom_shares_groups, 945
scripts_put_custom_shares_users, 946
scripts_put_custom_transfer, 947
scripts_put_dbt, 948
scripts_put_dbt_archive, 955
scripts_put_dbt_projects, 959
scripts_put_dbt_shares_groups, 959
scripts_put_dbt_shares_users, 960
scripts_put_dbt_transfer, 961
scripts_put_javascript, 962
scripts_put_javascript_archive, 967
scripts_put_javascript_git, 970
scripts_put_javascript_projects, 971
scripts_put_javascript_shares_groups, 972
scripts_put_javascript_shares_users, 973
scripts_put_javascript_transfer, 974
scripts_put_python3, 975
scripts_put_python3_archive, 980
scripts_put_python3_git, 984
scripts_put_python3_projects, 985
scripts_put_python3_shares_groups, 985
scripts_put_python3_shares_users, 986
scripts_put_python3_transfer, 987
scripts_put_r, 988
scripts_put_r_archive, 994
scripts_put_r_git, 997
scripts_put_r_projects, 998
scripts_put_r_shares_groups, 999
scripts_put_r_shares_users, 1000
scripts_put_r_transfer, 1001
scripts_put_sql, 1002
scripts_put_sql_archive, 1007
scripts_put_sql_git, 1011
scripts_put_sql_projects, 1012
scripts_put_sql_shares_groups, 1012
scripts_put_sql_shares_users, 1013
scripts_put_sql_transfer, 1014
search_list, 1015
search_list_queries, 1017
search_list_types, 1018
services_delete, 1019
services_delete_deployments, 1019
services_delete_projects, 1020
services_delete_shares_groups, 1020
services_delete_shares_users, 1021
services_delete_tokens, 1021
services_get, 1022
services_get_deployments, 1024
services_list, 1025
services_list_dependencies, 1027
services_list_deployments, 1027
services_list_deployments_logs, 1029
services_list_projects, 1030
services_list_shares, 1031
services_list_tokens, 1032
services_patch, 1032
services_post, 1036
services_post_clone, 1040
services_post_deployments, 1042
services_post_redeploy, 1043
services_post_tokens, 1044
services_put, 1045
services_put_archive, 1048
services_put_projects, 1051
services_put_shares_groups, 1051
services_put_shares_users, 1052
services_put_transfer, 1053
sql, 1054
stash_local_dataframe, 1055
storage_hosts_delete, 1055
storage_hosts_delete_shares_groups, 1056
storage_hosts_delete_shares_users, 1056
storage_hosts_get, 1057
storage_hosts_list, 1057
storage_hosts_list_dependencies, 1058
storage_hosts_list_shares, 1059
storage_hosts_patch, 1060
storage_hosts_post, 1061
storage_hosts_put, 1062
storage_hosts_put_shares_groups, 1063
storage_hosts_put_shares_users, 1064
storage_hosts_put_transfer, 1065
studios_delete_deployments, 1066
studios_get, 1066
studios_get_deployments, 1068
studios_list_deployments, 1069
studios_list_deployments_logs, 1071

studios_patch, 1072
studios_post, 1074
studios_post_deployments, 1077
studios_put, 1078
studios_put_archive, 1081
substitute, *39*

table_tags_delete, 1100
table_tags_get, 1101
table_tags_list, 1101
table_tags_post, 1102
tables_delete_projects, 1083
tables_delete_tags, 1083
tables_get, *620*, 1084
tables_get_enhancements_cass_ncoa, 1086
tables_get_enhancements_geocodings, 1087
tables_list, 1088
tables_list_columns, 1090
tables_list_projects, 1091
tables_patch, 1092
tables_post_enhancements_cass_ncoa, 1094
tables_post_enhancements_geocodings, 1095
tables_post_refresh, *619*, 1096
tables_post_scan, 1099
tables_put_projects, 1099
tables_put_tags, 1100
templates_delete_reports, 1103
templates_delete_reports_shares_groups, 1104
templates_delete_reports_shares_users, 1104
templates_delete_scripts, 1105
templates_delete_scripts_projects, 1105
templates_delete_scripts_shares_groups, 1106
templates_delete_scripts_shares_users, 1106
templates_get_reports, 1107
templates_get_scripts, 1108
templates_list_reports, 1109
templates_list_reports_dependencies, 1111
templates_list_reports_shares, 1111
templates_list_scripts, 1112
templates_list_scripts_dependencies, 1114
templates_list_scripts_projects, 1114
templates_list_scripts_shares, 1115
templates_patch_reports, 1116
templates_patch_scripts, 1118
templates_post_reports, 1119
templates_post_scripts, 1121
templates_put_reports, 1123
templates_put_reports_shares_groups, 1124
templates_put_reports_shares_users, 1125
templates_put_reports_transfer, 1126
templates_put_scripts, 1127
templates_put_scripts_projects, 1129
templates_put_scripts_shares_groups, 1129
templates_put_scripts_shares_users, 1130
templates_put_scripts_transfer, 1131
transfer_table, *325*, *620*, 1132

usage_get_llm, 1133
usage_limits_get_llm, 1134
usage_limits_get_matching, 1135
usage_limits_list_llm, 1135
usage_limits_list_matching, 1136
usage_list_llm, 1137
usage_list_llm_organization_summary, 1138
usage_list_matching, 1138
users_delete_2fa, 1139
users_delete_api_keys, 1141
users_delete_me_favorites, 1142
users_delete_sessions, 1143
users_get, 1144
users_get_api_keys, 1146
users_list, 1147
users_list_api_keys, 1149
users_list_me, 1150
users_list_me_activity, 1151
users_list_me_favorites, 1152
users_list_me_organization_admins, 1153
users_patch, 1154
users_patch_me, 1157
users_patch_me_favorites_ranking_bottom, 1160

users_patch_me_favorites_ranking_higher, 1161
users_patch_me_favorites_ranking_lower, 1161
users_patch_me_favorites_ranking_top, 1162
users_post, 1162
users_post_access_email, 1165
users_post_api_keys, 1166
users_post_me_favorites, 1167
users_post_unsuspend, 1168

workflows_delete_projects, 1168
workflows_delete_shares_groups, 1169
workflows_delete_shares_users, 1169
workflows_get, 1170
workflows_get_executions, 1171
workflows_get_executions_tasks, 1173
workflows_get_git_commits, 1174
workflows_list, 1174
workflows_list_dependencies, 1176
workflows_list_executions, 1177
workflows_list_git, 1178
workflows_list_git_commits, 1179
workflows_list_projects, 1179
workflows_list_shares, 1180
workflows_patch, 1181
workflows_patch_git, 1184
workflows_post, 1185
workflows_post_clone, 1187
workflows_post_executions, 1189
workflows_post_executions_cancel, 1191
workflows_post_executions_resume, 1192
workflows_post_executions_retry, 1193
workflows_post_git_checkout_latest, 1194
workflows_post_git_commits, 1195
workflows_put, 1195
workflows_put_archive, 1198
workflows_put_git, 1199
workflows_put_projects, 1201
workflows_put_shares_groups, 1201
workflows_put_shares_users, 1202
workflows_put_transfer, 1203
write.csv, *1207, 1208*
write_chunks, 1204
write_civis, *146*, *615*, *617*, *619*, 1205, *1208*
write_civis_file, *146*, *615*, *617*, *619*, *1207*, 1207
write_job_output, 1209