

---

# **Civis Client Documentation**

***Release 1.0.0***

**Civis Analytics**

**Mar 29, 2017**



---

## Contents

---

<b>1</b>	<b>Installation</b>	<b>3</b>
<b>2</b>	<b>Authentication</b>	<b>5</b>
<b>3</b>	<b>User Guide</b>	<b>7</b>
<b>4</b>	<b>Client API Reference</b>	<b>9</b>
<b>5</b>	<b>Indices and tables</b>	<b>407</b>



The Civis API Python client is a Python package that helps analysts and developers interact with the Civis Platform. The package includes a set of tools around common workflows as well as a convenient interface to make requests directly to the Civis API.



# CHAPTER 1

---

## Installation

---

The recommended install method is pip:

```
pip install civis
```

Alternatively, you may clone the code from github and build from source:

```
git clone https://github.com/civisanalytics/civis-python.git
cd civis-python
python setup.py install
```

The client has a soft dependency on `pandas` to support features such as data type parsing. If you are using the `io` namespace to read or write data from Civis, it is highly recommended that you install `pandas` and set `use_pandas=True` in functions that accept that parameter. To install `pandas`:

```
pip install pandas
```





## CHAPTER 2

---

### Authentication

---

In order to make requests to the Civis API, you will need an API key that is unique to you. Instructions for creating a new key are found here: <https://civis.zendesk.com/hc/en-us/articles/216341583-Generating-an-API-Key>. By default, the Python client will look for your key in the environment variable `CIVIS_API_KEY`. To add the API key to your environment, copy the key you generated to your clipboard and follow the instructions below for your operating system.

#### Mac

Open `.bash_profile` in TextEdit:

```
cd ~/
touch .bash_profile
open -e .bash_profile
```

Then add the following line, replacing `api_key_here` with your key:

```
export CIVIS_API_KEY="api_key_here"
```

#### Linux

Open `.bash_profile` in your favorite editor (nano is used here):

```
cd ~/
nano .bash_profile
```

Then add the following line, replacing `api_key_here` with your key:

```
export CIVIS_API_KEY="api_key_here"
```



## CHAPTER 3

---

### User Guide

---

For a more detailed walkthrough, see the *User Guide*.



## User Guide

### Getting Started

After installing the Cavis API Python client and setting up your API key, you can now import the package `cavis`:

```
>>> import cavis
```

There are two entrypoints for working with the Cavis API. The first is the `cavis` namespace, which contains tools for typical workflows in a user friendly manner. For example, you may want to perform some transformation on your data in Python that might be tricky to code in SQL. This code downloads data from Cavis, calculates the correlation between all the columns and then uploads the data back into Cavis:

```
>>> df = cavis.io.read_cavis(table="my_schema.my_table",
...                           database="database",
...                           use_pandas=True)
>>> correlation_matrix = df.corr()
>>> correlation_matrix["corr_var"] = correlation_matrix.index
>>> poller = cavis.io.dataframe_to_cavis(df=correlation_matrix,
...                                     database="database",
...                                     table="my_schema.my_correlations")
...
>>> poller.result()
```

### Pollable Results

In the code above, `dataframe_to_cavis()` returns a special `PollableResult` object. Making a request to the Cavis API usually results in a long running job. To account for this, various functions in the `cavis` namespace return a `PollableResult` to allow you to process multiple long running jobs simultaneously. For instance, you may want to start many jobs in parallel and wait for them all to finish rather than wait for each job to finish before starting the next one.

The `PollableResult` follows the `concurrent.futures.Future` API fairly closely. For example, calling `result()` on `poller` above forces the program to wait for the job started with `dataframe_to_civis()` to finish and returns the result.

## Working Directly with the Client

Although many common workflows are included in the Civis API Python client, projects often require direct calls to the Civis API. For convenience, the Civis API Python client implements an `APIClient` object to make these API calls with Python syntax rather than a manually crafted HTTP request. To make a call, first instantiate an `APIClient` object:

```
>>> client = civis.APIClient()
```

**Note:** Creating an instance of `APIClient` makes an HTTP request to determine the functions to attach to the object. You must have an API key and internet connection to create an `APIClient` object. By default, the functions attached to the object come from a base set of Civis API endpoints. Based on your user profile, you may have access to a set of developmental endpoints. To access these, instantiate the client with `client = civis.APIClient(resources='all')`.

With the client object instantiated, you can now make API requests like listing your user information:

```
>>> client.users.list_me()
{'email': 'user@email.com',
 'feature_flags': {'left_nav_basic': True,
                  'results': True,
                  'scripts_notify': True,
                  'table_person_matching': True},
 'id': 1,
 'initials': 'UN',
 'name': 'User Name',
 'username': 'uname'}
```

Suppose we did not have the `civis.io` namespace. This is how we might export a CSV file from Civis. As you will see, this can be quite involved and the `civis` namespace entrypoint should be preferred whenever possible.

First, we get the ID for our database then we get the default credential for the current user.

```
>>> db_id = client.get_database_id('cluster-name')
>>> cred_id = client.default_credential
```

In order to export a table, we need to write some SQL that will generate the data to export. Then we create the export job and run it.

```
>>> generate_table = "select * from schema.tablename"
>>> export_job = client.scripts.post_sql(name="our export job",
                                       remote_host_id=db_id,
                                       credential_id=cred_id,
                                       sql=generate_table)
>>> export_run = client.scripts.post_sql_runs(export_job.id)
```

We can then poll and wait for the export to be completed.

```
>>> import time
>>> export_state = client.scripts.get_sql_runs(export_job.id,
...                                           export_run.id)
```

```
>>> while export_state.state in ['queued', 'running']:
...     time.sleep(60)
...     export_state = client.scripts.get_sql_runs(export_job.id,
...                                                export_run.id)
```

Now, we can get the URL of the exported csv. First, we grab the result of our export job.

```
>>> export_result = client.scripts.get_sql_runs(export_job.id,
...                                             export_run.id)
```

In the future, a script may export multiple jobs, so the output of this is a list.

The path returned will have a gzipped csv file, which we could load, for example, with pandas.

```
>>> url = export_result.output[0].path
```

## Data Import and Export

The `civis.io` namespace provides several functions for moving data in and out of Civis.

### Tables

Often, your data will be in structured format like a table in a relational database, a CSV or a dataframe. The following functions handle moving structured data to and from Civis. When using these functions, it is recommended to have *pandas* installed and to pass *use\_pandas=True* in the appropriate functions. If *pandas* is not installed, data returned from Civis will all be treated as strings.

<code>civis_to_csv(filename, sql, database[, ...])</code>	Export data from Civis to a local CSV file.
<code>csv_to_civis(filename, database, table[, ...])</code>	Upload the contents of a local CSV file to Civis.
<code>dataframe_to_civis(df, database, table[, ...])</code>	Upload a <i>pandas DataFrame</i> into a Civis table.
<code>read_civis(table, database[, columns, ...])</code>	Read data from a Civis table.
<code>read_civis_sql(sql, database[, use_pandas, ...])</code>	Read data from Civis using a custom SQL string.

### `civis.io.civis_to_csv`

`civis.io.civis_to_csv(filename, sql, database, job_name=None, api_key=None, credential_id=None, archive=False, hidden=True, polling_interval=15)`  
Export data from Civis to a local CSV file.

**Parameters** `filename` : str

Download exported data into this file.

`sql` : str, optional

The SQL select string to be executed.

`database` : str or int

Export data from this database. Can be the database name or ID.

`job_name` : str, optional

A name to give the job. If omitted, a random job name will be used.

**api\_key** : str, optional

Your Civis API key. If not given, the `CIVIS_API_KEY` environment variable will be used.

**credential\_id** : str or int, optional

The ID of the database credential. If `None`, the default credential will be used.

**polling\_interval** : int or float, optional

Number of seconds to wait between checks for query completion.

**archive** : bool, optional (deprecated)

If `True`, archive the import job as soon as it completes.

**hidden** : bool, optional

If `True` (the default), this job will not appear in the Civis UI.

**Returns results** : *PollableResult*

A *PollableResult* object.

**See also:**

*civis.io.read\_civis* Read table contents into memory.

*civis.io.read\_civis\_sql* Read results of a SQL query into memory.

## Examples

```
>>> sql = "SELECT * FROM schema.table"
>>> poll = civis_to_csv("file.csv", sql, "my_database")
>>> poll.result() # Wait for job to complete
```

## civis.io.csv\_to\_civis

*civis.io.csv\_to\_civis* (*filename*, *database*, *table*, *api\_key=None*, *max\_errors=None*, *existing\_table\_rows='fail'*, *distkey=None*, *sortkey1=None*, *sortkey2=None*, *delimiter=''*, *headers=None*, *credential\_id=None*, *polling\_interval=15*, *archive=False*, *hidden=True*)

Upload the contents of a local CSV file to Civis.

**Parameters filename** : str

Upload the contents of this file.

**database** : str or int

Upload data into this database. Can be the database name or ID.

**table** : str

The schema and table you want to upload to. E.g., `'scratch.table'`.

**api\_key** : str, optional

Your Civis API key. If not given, the `CIVIS_API_KEY` environment variable will be used.

**max\_errors** : int, optional



The maximum number of rows with errors to remove from the import before failing.

**existing\_table\_rows** : str, optional

The behaviour if a table with the requested name already exists. One of 'fail', 'truncate', 'append' or 'drop'. Defaults to 'fail'.

**distkey** : str, optional

The column to use as the distkey for the table.

**sortkey1** : str, optional

The column to use as the sortkey for the table.

**sortkey2** : str, optional

The second column in a compound sortkey for the table.

**delimiter** : string, optional

The column delimiter. One of ',', '\t' or '| '.

**headers** : bool, optional

Whether or not the first row of the file should be treated as headers. The default, `None`, attempts to autodetect whether or not the first row contains headers.

**credential\_id** : str or int, optional

The ID of the database credential. If `None`, the default credential will be used.

**polling\_interval** : int or float, optional

Number of seconds to wait between checks for job completion.

**archive** : bool, optional (deprecated)

If `True`, archive the import job as soon as it completes.

**hidden** : bool, optional

If `True` (the default), this job will not appear in the Civis UI.

**Returns** **results** : *PollableResult*

A *PollableResult* object.

## Notes

This reads the contents of *filename* into memory.

## Examples

```
>>> with open('input_file.csv', 'w') as _input:
...     _input.write('a,b,c\n1,2,3')
>>> poller = civis.io.csv_to_civis('input_file.csv',
...                               'my-database',
...                               'scratch.my_data')
>>> poller.result()
```

**civis.io.dataframe\_to\_civis**

```
civis.io.dataframe_to_civis(df, database, table, api_key=None, max_errors=None, existing_table_rows='fail', distkey=None, sortkey1=None, sortkey2=None, headers=None, credential_id=None, polling_interval=15, archive=False, hidden=True, **kwargs)
```

Upload a *pandas DataFrame* into a Civis table.

**Parameters** **df**: `pandas.DataFrame`

The *DataFrame* to upload to Civis.

**database** : str or int

Upload data into this database. Can be the database name or ID.

**table** : str

The schema and table you want to upload to. E.g., 'scratch.table'.

**api\_key** : str, optional

Your Civis API key. If not given, the `CIVIS_API_KEY` environment variable will be used.

**max\_errors** : int, optional

The maximum number of rows with errors to remove from the import before failing.

**existing\_table\_rows** : str, optional

The behaviour if a table with the requested name already exists. One of 'fail', 'truncate', 'append' or 'drop'. Defaults to 'fail'.

**distkey** : str, optional

The column to use as the distkey for the table.

**sortkey1** : str, optional

The column to use as the sortkey for the table.

**sortkey2** : str, optional

The second column in a compound sortkey for the table.

**headers** : bool, optional

Whether or not the first row of the file should be treated as headers. The default, `None`, attempts to autodetect whether or not the first row contains headers.

**credential\_id** : str or int, optional

The ID of the database credential. If `None`, the default credential will be used.

**polling\_interval** : int or float, optional

Number of seconds to wait between checks for job completion.

**archive** : bool, optional (deprecated)

If `True`, archive the import job as soon as it completes.

**hidden** : bool, optional

If `True` (the default), this job will not appear in the Civis UI.

**\*\*kwargs** : kwargs

Extra keyword arguments will be passed to `pandas.DataFrame.to_csv()`.

**Returns** `poll` : *PollableResult*

A *PollableResult* object.

## Examples

```
>>> import pandas as pd
>>> df = pd.DataFrame({'a': [1, 2, 3], 'b': [4, 5, 6]})
>>> poller = civis.io.dataframe_to_civis(df, 'my-database',
...                                     'scratch.df_table')
>>> poller.result()
```

## civis.io.read\_civis

```
civis.io.read_civis(table, database, columns=None, use_pandas=False, job_name=None,
                    api_key=None, credential_id=None, polling_interval=15, archive=False,
                    hidden=True, **kwargs)
```

Read data from a Civis table.

**Parameters** `table` : str

Name of table, including schema, in the database. I.e. 'my\_schema.my\_table'.

**database** : str or int

Read data from this database. Can be the database name or ID.

**columns** : list, optional

A list of column names. Column SQL transformations are possible. If omitted, all columns are exported.

**use\_pandas** : bool, optional

If `True`, return a `pandas.DataFrame`. Otherwise, return a list of results from `csv.reader()`.

**job\_name** : str, optional

A name to give the job. If omitted, a random job name will be used.

**api\_key** : str, optional

Your Civis API key. If not given, the `CIVIS_API_KEY` environment variable will be used.

**credential\_id** : str or int, optional

The database credential ID. If `None`, the default credential will be used.

**polling\_interval** : int or float, optional

Number of seconds to wait between checks for query completion.

**archive** : bool, optional (deprecated)

If `True`, archive the import job as soon as it completes.

**hidden** : bool, optional

If `True` (the default), this job will not appear in the Civis UI.

**\*\*kwargs** : kwargs

Extra keyword arguments are passed into `pandas.read_csv()` if `use_pandas` is True or passed into `csv.reader()` if `use_pandas` is False.

**Returns data** : `pandas.DataFrame` or list

A list of rows (with header as first row) if `use_pandas` is False, otherwise a *pandas DataFrame*. Note that if `use_pandas` is False, no parsing of types is performed and each row will be a list of strings.

**Raises ImportError**

If `use_pandas` is True and *pandas* is not installed.

**See also:**

`civis.io.read_civis_sql` Read directly into memory using SQL.

`civis.io.civis_to_csv` Write directly to csv.

## Examples

```
>>> table = "schema.table"
>>> database = "my_data"
>>> columns = ["column_a", "ROW_NUMBER() OVER(ORDER BY date) AS order"]
>>> data = read_civis(table, database, columns=columns)
>>> columns = data.pop(0)
>>> col_a_index = columns.index("column_a")
>>> col_a = [row[col_a_index] for row in data]
```

```
>>> df = read_civis("schema.table", "my_data", use_pandas=True)
>>> col_a = df["column_a"]
```

## `civis.io.read_civis_sql`

`civis.io.read_civis_sql`(*sql*, *database*, *use\_pandas*=False, *job\_name*=None, *api\_key*=None, *credential\_id*=None, *polling\_interval*=15, *archive*=False, *hidden*=True, **\*\*kwargs**)

Read data from Civis using a custom SQL string.

**Parameters sql** : str, optional

The SQL select string to be executed.

**database** : str or int

Execute the query against this database. Can be the database name or ID.

**use\_pandas** : bool, optional

If True, return a `pandas.DataFrame`. Otherwise, return a list of results from `csv.reader()`.

**job\_name** : str, optional

A name to give the job. If omitted, a random job name will be used.

**api\_key** : str, optional

Your Civis API key. If not given, the `CIVIS_API_KEY` environment variable will be used.

**credential\_id** : str or int, optional

The database credential ID. If `None`, the default credential will be used.

**polling\_interval** : int or float, optional

Number of seconds to wait between checks for query completion.

**archive** : bool, optional (deprecated)

If `True`, archive the import job as soon as it completes.

**hidden** : bool, optional

If `True` (the default), this job will not appear in the Civis UI.

**\*\*kwargs** : kwargs

Extra keyword arguments are passed into `pandas.read_csv()` if `use_pandas` is `True` or passed into `csv.reader()` if `use_pandas` is `False`.

**Returns** **data** : `pandas.DataFrame` or list

A list of rows (with header as first row) if `use_pandas` is `False`, otherwise a `pandas.DataFrame`. Note that if `use_pandas` is `False`, no parsing of types is performed and each row will be a list of strings.

**Raises** **ImportError**

If `use_pandas` is `True` and `pandas` is not installed.

See also:

`civis.io.read_civis` Read directly into memory without SQL.

`civis.io.civis_to_csv` Write directly to a CSV file.

## Notes

This reads the data into memory.

## Examples

```
>>> sql = "SELECT * FROM schema.table"
>>> df = read_civis_sql(sql, "my_database", use_pandas=True)
>>> col_a = df["column_a"]
```

```
>>> data = read_civis_sql(sql, "my_database")
>>> columns = data.pop(0)
>>> col_a_index = columns.index("column_a")
>>> col_a = [row[col_a_index] for row in data]
```

## **Files**

These functions will pass flat files to and from Civis. This is useful if you have data stored in binary or JSON format. Any type of file can be stored in platform via the files endpoint.

<code>civis_to_file(file_id, buf[, api_key])</code>	Download a file from Civis.
<code>file_to_civis(buf, name[, api_key])</code>	Upload a file to Civis.

### civis.io.civis\_to\_file

`civis.io.civis_to_file` (*file\_id*, *buf*, *api\_key=None*)

Download a file from Civis.

**Parameters** *file\_id* : int

The Civis file ID.

*buf* : file-like object

The file or other buffer to write the contents of the Civis file into.

*api\_key* : str, optional

Your Civis API key. If not given, the `CIVIS_API_KEY` environment variable will be used.

**Returns** None

### Examples

```
>>> file_id = 100
>>> with open("my_file.txt", "w") as f:
...     civis_to_file(file_id, f)
```

### civis.io.file\_to\_civis

`civis.io.file_to_civis` (*buf*, *name*, *api\_key=None*, *\*\*kwargs*)

Upload a file to Civis.

**Parameters** *buf* : file-like object

The file or other buffer that you wish to upload.

*name* : str

The name you wish to give the file.

*api\_key* : str, optional

Your Civis API key. If not given, the `CIVIS_API_KEY` environment variable will be used.

**\*\*kwargs** : kwargs

Extra keyword arguments will be passed to the file creation endpoint. See `post()`.

**Returns** *file\_id* : int

The new Civis file ID.

## Notes

If you are opening a binary file (e.g., a compressed archive) to pass to this function, do so using the 'rb' (read binary) mode (e.g., `open('myfile.zip', 'rb')`).

## Examples

```
>>> # Upload file which expires in 30 days
>>> with open("my_data.csv", "r") as f:
...     file_id = file_to_civis(f, 'my_data')
>>> # Upload file which never expires
>>> with open("my_data.csv", "r") as f:
...     file_id = file_to_civis(f, 'my_data', expires_at=None)
```

## Databases

These functions move data from one database to another and expose an interface to run SQL in the database. Use `query_civis()` when you need to execute SQL that does not return data (for example, a GRANT or DROP TABLE statement).

<code>transfer_table(source_db, dest_db, ...[, ...])</code>	Transfer a table from one location to another.
<code>query_civis(sql, database[, api_key, ...])</code>	Execute a SQL statement as a Civis query.

### civis.io.transfer\_table

```
civis.io.transfer_table(source_db, dest_db, source_table, dest_table, job_name=None,
                        api_key=None, source_credential_id=None, dest_credential_id=None,
                        polling_interval=15, **advanced_options)
```

Transfer a table from one location to another.

**Parameters** `source_db` : str or int

The name of the database where the source table is located. Optionally, could be the database ID.

`dest_db` : str or int

The name of the database where the table will be transferred. Optionally, could be the database ID.

`source_table` : str

Full name of the table to transfer, e.g., 'schema.table'.

`dest_table` : str

Full name of the table in the destination database, e.g., 'schema.table'.

`job_name` : str, optional

A name to give the job. If omitted, a random job name will be used.

`api_key` : str, optional

Your Civis API key. If not given, the CIVIS\_API\_KEY environment variable will be used.



**source\_credential\_id** : str or int, optional

Optional credential ID for the source database. If `None`, the default credential will be used.

**dest\_credential\_id** : str or int, optional

Optional credential ID for the destination database. If `None`, the default credential will be used.

**polling\_interval** : int or float, optional

Number of seconds to wait between checks for job completion.

**\*\*advanced\_options** : kwargs

Extra keyword arguments will be passed to the import sync job. See `post_syncs()`.

**Returns results** : `PollableResult`

A `PollableResult` object.

## Examples

```
>>> transfer_table(source_db='Cluster A', dest_db='Cluster B',
...               source_table='schema.tbl', dest_table='schema.tbl')
```

## civis.io.query\_civis

`civis.io.query_civis(sql, database, api_key=None, credential_id=None, preview_rows=10, polling_interval=15, hidden=True)`

Execute a SQL statement as a Civis query.

Run a query that may return no results or where only a small preview is required. To execute a query that returns a large number of rows, see `read_civis_sql()`.

**Parameters sql** : str

The SQL statement to execute.

**database** : str or int

The name or ID of the database.

**api\_key** : str, optional

Your Civis API key. If not given, the `CIVIS_API_KEY` environment variable will be used.

**credential\_id** : str or int, optional

The ID of the database credential. If `None`, the default credential will be used.

**preview\_rows** : int, optional

The maximum number of rows to return. No more than 100 rows can be returned at once.

**polling\_interval** : int or float, optional

Number of seconds to wait between checks for query completion.

**hidden** : bool, optional

If `True` (the default), this job will not appear in the Civis UI.

**Returns** `results` : `PollableResult`

A `PollableResult` object.

### Examples

```
>>> run = query_civis(sql="DELETE schema.table", database='database')
>>> run.result() # Wait for query to complete
```

## API Client

`APIClient` is a class for handling requests to the Civis API. An instantiated `APIClient` contains a set of resources (listed below) where each resource is an object with methods. By convention, an instantiated `APIClient` object is named `client` and API requests are made with the following syntax:

```
client = civis.APIClient()
response = client.resource.method(params)
```

**class** `civis.APIClient` (*api\_key=None*, *return\_type='snake'*, *retry\_total=6*, *api\_version='1.0'*, *resources='base'*)

The Civis API client.

**Parameters** `api_key` : str, optional

Your API key obtained from the Civis Platform. If not given, the client will use the `CIVIS_API_KEY` environment variable.

**return\_type** : str, optional

The following types are implemented:

- `'raw'` Returns the raw `requests.Response` object.
- `'snake'` Returns a `civis.response.Response` object for the json-encoded content of a response. This maps the top-level json keys to snake\_case.
- `'pandas'` Returns a `pandas.DataFrame` for list-like responses and a `pandas.Series` for single a json response.

**retry\_total** : int, optional

A number indicating the maximum number of retries for 429, 502, 503, or 504 errors.

**api\_version** : string, optional

The version of endpoints to call. May instantiate multiple client objects with different versions. Currently only “1.0” is supported.

**resources** : string, optional

When set to “base”, only the default endpoints will be exposed in the client object. Set to “all” to include all endpoints available for a given user, including those that may be in development and subject to breaking changes at a later date.

## Attributes

credentials	An instance of the <i>Credentials</i> endpoint
databases	An instance of the <i>Databases</i> endpoint
files	An instance of the <i>Files</i> endpoint
imports	An instance of the <i>Imports</i> endpoint
jobs	An instance of the <i>Jobs</i> endpoint
models	An instance of the <i>Models</i> endpoint
predictions	An instance of the <i>Predictions</i> endpoint
projects	An instance of the <i>Projects</i> endpoint
queries	An instance of the <i>Queries</i> endpoint
reports	An instance of the <i>Reports</i> endpoint
scripts	An instance of the <i>Scripts</i> endpoint
tables	An instance of the <i>Tables</i> endpoint
users	An instance of the <i>Users</i> endpoint

### default\_credential

The current user's default credential.

### get\_aws\_credential\_id(cred\_name, owner=None)

Find an AWS credential ID.

**Parameters** `cred_name` : str or int

If an integer ID is given, this passes through directly. If a str is given, return the ID corresponding to the AWS credential with that name.

**owner** : str, optional

Return the credential with this owner. If not provided, search for credentials under your username to disambiguate multiple credentials with the same name. Note that this function cannot return credentials which are not associated with an owner.

**Returns** `aws_credential_id` : int

The ID number of the AWS credentials.

### Raises ValueError

If the AWS credential can't be found.

## Examples

```
>>> import civis
>>> client = civis.APIClient()
>>> client.get_aws_credential_id('jsmith')
1234
```

```
>>> client.get_aws_credential_id(1111)
1111
```

```
>>> client.get_aws_credential_id('shared-cred',
...                               owner='research-group')
99
```

### get\_database\_credential\_id(username, database\_name)

Return the credential ID for a given username in a given database.

**Parameters** `username` : str or int

If an integer ID is given, this passes through directly. If a str is given, return the ID corresponding to the database credential with that username.

**database\_name** : str or int

Return the ID of the database credential with username *username* for this database name or ID.

**Returns** `database_credential_id` : int

The ID of the database credentials.

**Raises** `ValueError`

If the credential can't be found.

## Examples

```
>>> import civis
>>> client = civis.APIClient()
>>> client.get_database_credential_id('jsmith', 'redshift-general')
1234
```

```
>>> client.get_database_credential_id(1111, 'redshift-general')
1111
```

**get\_database\_id** (*database*)

Return the database ID for a given database name.

**Parameters** `database` : str or int

If an integer ID is given, passes through. If a str is given the database ID corresponding to that database name is returned.

**Returns** `database_id` : int

The ID of the database.

**Raises** `ValueError`

If the database can't be found.

**get\_table\_id** (*table*, *database*)

Return the table ID for a given database and table name.

**Parameters** `table` : str

The name of the table in format schema.table.

**database** : str or int

The name or ID of the database.

**Returns** `table_id` : int

The ID of the table. Only returns exact match to specified table.

**Raises** `ValueError`

If an exact table match can't be found.

**username**

The current user's username.

## API Response Types

**class** `civis.response.Response` (*json\_data, snake\_case=True, headers=None*)  
Custom Civis response object.

### Notes

The main features of this class are that it maps camelCase to snake\_case at the top level of the json object and attaches keys as attributes. Nested object keys are not changed.

### Attributes

<code>json_data</code>	(dict   None) This is <i>json_data</i> as it is originally returned to the user without the key names being changed. See Notes. None is used if the original response returned a 204 No Content response.
<code>headers</code>	(dict) This is the header for the API call without changing the key names.
<code>calls_remaining</code>	(int) Number of API calls remaining before rate limit is reached.
<code>rate_limit</code>	(int) Total number of calls per API rate limit period.

**class** `civis.response.PaginatedResponse` (*path, initial\_params, endpoint*)  
A response object that supports iteration.

**Parameters** `path` : str

Make GET requests to this path.

**initial\_params** : dict

Query params that should be passed along with each request. Note that if *initial\_params* contains the keys *page\_num* or *limit*, they will be ignored. The given dict is not modified.

**endpoint** : *civis.base.Endpoint*

An endpoint used to make API requests.

### Notes

This response is returned automatically by endpoints which support pagination when the *iterator* kwarg is specified.

### Examples

```
>>> client = civis.APIClient()
>>> queries = client.queries.list(iterator=True)
>>> for query in queries:
...     print(query['id'])
```

**class** `civis.polling.PollableResult` (*poller, poller\_args, polling\_interval=15, api\_key=None*)  
Bases: `civis.base.CivisAsyncResultBase`

A class for tracking pollable results.

This class will begin polling immediately upon creation, and poll for job completion once every *polling\_interval* seconds until the job completes in Civis.

**Parameters** **poller** : func

A function which returns an object that has a `state` attribute.

**poller\_args** : tuple

The arguments with which to call the poller function.

**polling\_interval** : int or float

The number of seconds between API requests to check whether a result is ready.

**api\_key** : str, optional

This is not used by `PollableResult`, but is required to match the interface from `CivisAsyncResultBase`.

## Examples

```
>>> client = civis.APIClient()
>>> database_id = client.get_database_id("my_database")
>>> cred_id = client.default_credential
>>> sql = "SELECT 1"
>>> preview_rows = 10
>>> response = client.queries.post(database_id, sql, preview_rows,
>>>                                credential=cred_id)
>>> job_id = response.id
>>>
>>> poller = client.queries.get
>>> poller_args = (job_id, ) # (job_id, run_id) if poller requires run_id
>>> polling_interval = 10
>>> poll = PollableResult(poller, poller_args, polling_interval)
```

## API Resources

### Credentials

**class** `Credentials` (*session, return\_type='civis'*)

#### Methods

<code>get(id)</code>	Get a credential
<code>list(**kwargs)</code>	List credentials
<code>post(type, password, username, **kwargs)</code>	Create or update a credential
<code>post_authenticate(url, password, ...)</code>	Authenticate against a remote host
<code>post_temporary(id, **kwargs)</code>	Generate a temporary credential for accessing S3
<code>put(id, type, password, username, **kwargs)</code>	Update an existing credential

**get** (*id*)

Get a credential

**Parameters** *id* : integer

The ID of the credential.

**Returns** *id* : integer

The ID of the credential.

**description** : string

A long description of the credential.

**type** : string

The credential's type.

**name** : string

The name identifying the credential

**remote\_host\_id** : integer

The ID of the remote host associated with this credential.

**updated\_at** : string/time

The last modification time for this credential.

**created\_at** : string/time

The creation time for this credential.

**remote\_host\_name** : string

The name of the remote host associated with this credential.

**username** : string

The username for the credential.

**owner** : string

The name of the user who this credential belongs to.

**list** (*\*\*kwargs*)

List credentials

**Parameters** *type* : string, optional

The type (or types) of credentials to return. One or more of: Amazon Web Services S3, BSD::API, CASS/NCOA PAF, Catalist::API, Catalist::SFTP, Certificate, Civis Platform, Custom, Database, Google, Github, JobTraits::Ftp, Salesforce User, Salesforce Client, Silverpop Application, Silverpop Refresh Token, Silverpop User, TableauUser, VAN::MyVoterFile, VAN::MyCampaign, and VAN::BothModes. Specify multiple values as a comma-separated list (e.g., "A,B").

**default** : boolean, optional

If true, will return a list with a single credential which is the current user's default credential.

**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`.

**iterator** : bool, optional

If `True`, return a generator to iterate over all responses. Use when more results than the maximum allowed by `limit` are needed. When `True`, `limit` and `page_num` are ignored. Defaults to `False`.

**Returns** **id** : integer

The ID of the credential.

**description** : string

A long description of the credential.

**type** : string

The credential's type.

**name** : string

The name identifying the credential

**remote\_host\_id** : integer

The ID of the remote host associated with this credential.

**updated\_at** : string/time

The last modification time for this credential.

**created\_at** : string/time

The creation time for this credential.

**remote\_host\_name** : string

The name of the remote host associated with this credential.

**username** : string

The username for the credential.

**owner** : string

The name of the user who this credential belongs to.

**post** (*type, password, username, \*\*kwargs*)

Create or update a credential

**Parameters** **type** : string

**password** : string

The password for the credential.

**username** : string

The username for 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.

**name** : string, optional

The name identifying the credential.

**remote\_host** : dict, optional:

```
- url : string
    The URL to your host.
- type : string
    The type of remote host. One of: RemoteHostTypes::BSD,
    RemoteHostTypes::Ftp, RemoteHostTypes::Github,
    RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC,
    RemoteHostTypes::Redshift, RemoteHostTypes::Salesforce, and
    RemoteHostTypes::Van
- name : string
    The human readable name for the remote host.
```

**Returns id** : integer

The ID of the credential.

**description** : string

A long description of the credential.

**type** : string

The credential's type.

**name** : string

The name identifying the credential

**remote\_host\_id** : integer

The ID of the remote host associated with this credential.

**updated\_at** : string/time

The last modification time for this credential.

**created\_at** : string/time

The creation time for this credential.

**remote\_host\_name** : string

The name of the remote host associated with this credential.

**username** : string

The username for the credential.

**owner** : string

The name of the user who this credential belongs to.

**post\_authenticate** (*url, password, remote\_host\_type, username*)

Authenticate against a remote host

**Parameters** `url` : string

The URL to your host.

**password** : string

The password for the credential.

**remote\_host\_type** : string

The type of remote host. One of: RemoteHostTypes::BSD, RemoteHostTypes::Ftp, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Redshift, RemoteHostTypes::Salesforce, and RemoteHostTypes::Van

**username** : string

The username for the credential.

**Returns** `id` : integer

The ID of the credential.

**description** : string

A long description of the credential.

**type** : string

The credential's type.

**name** : string

The name identifying the credential

**remote\_host\_id** : integer

The ID of the remote host associated with this credential.

**updated\_at** : string/time

The last modification time for this credential.

**created\_at** : string/time

The creation time for this credential.

**remote\_host\_name** : string

The name of the remote host associated with this credential.

**username** : string

The username for the credential.

**owner** : string

The name of the user who this credential belongs to.

**post\_temporary** (*id*, *\*\*kwargs*)

Generate a temporary credential for accessing S3

**Parameters** `id` : integer

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.

**Returns** `access_key` : string

The identifier of the credential.

**secret\_access\_key** : string

The secret part of the credential.

**session\_token** : string

The session token identifier.

**put** (*id, type, password, username, \*\*kwargs*)

Update an existing credential

**Parameters** `id` : integer

The ID of the credential.

**type** : string

**password** : string

The password for the credential.

**username** : string

The username for 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.

**name** : string, optional

The name identifying the credential.

**remote\_host** : dict, optional:

```
- url : string
    The URL to your host.
- type : string
    The type of remote host. One of: RemoteHostTypes::BSD,
    RemoteHostTypes::Ftp, RemoteHostTypes::Github,
    RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC,
    RemoteHostTypes::Redshift, RemoteHostTypes::Salesforce, and
    RemoteHostTypes::Van
- name : string
    The human readable name for the remote host.
```

**Returns** `id` : integer

The ID of the credential.

**description** : string

A long description of the credential.

**type** : string

The credential's type.

**name** : string

The name identifying the credential

**remote\_host\_id** : integer

The ID of the remote host associated with this credential.

**updated\_at** : string/time

The last modification time for this credential.

**created\_at** : string/time

The creation time for this credential.

**remote\_host\_name** : string

The name of the remote host associated with this credential.

**username** : string

The username for the credential.

**owner** : string

The name of the user who this credential belongs to.

## Databases

**class Databases** (*session*, *return\_type*='civis')

### Methods

<code>delete_whitelist_ips(id, whitelisted_ip_id)</code>	Remove a whitelisted IP address
<code>get_whitelist_ips(id, whitelisted_ip_id)</code>	View details about a whitelisted IP
<code>list()</code>	List databases
<code>list_schemas(id)</code>	List schemas in this database
<code>list_whitelist_ips(id)</code>	List whitelisted IPs for the specified database
<code>post_whitelist_ips(id, subnet_mask)</code>	Whitelist an IP address

**delete\_whitelist\_ips** (*id*, *whitelisted\_ip\_id*)

Remove a whitelisted IP address

**Parameters** **id** : integer

The ID of the database this rule is applied to.

**whitelisted\_ip\_id** : integer

The ID of this whitelisted IP address.

**Returns** None

Response code 204: success

**get\_whitelist\_ips** (*id*, *whitelisted\_ip\_id*)

View details about a whitelisted IP

**Parameters** **id** : integer

The ID of the database this rule is applied to.

**whitelisted\_ip\_id** : integer

The ID of this whitelisted IP address.

**Returns id** : integer

The ID of this whitelisted IP address.

**remote\_host\_id** : integer

The ID of the database this rule is applied to.

**is\_active** : boolean

True if the rule is applied, false if it has been revoked.

**security\_group\_id** : string

The ID of the security group this rule is applied to.

**created\_at** : string/time

The time this rule was created.

**authorized\_by** : string

The user who authorized this rule.

**subnet\_mask** : string

The subnet mask that is allowed by this rule.

**updated\_at** : string/time

The time this rule was last updated.

**list** ()

List databases

**Returns id** : integer

The ID for the database.

**name** : string

The name of the database.

**list\_schemas** (*id*)

List schemas in this database

**Parameters id** : integer

The ID of the database.

**Returns schema** : string

The name of a schema.

**list\_whitelist\_ips** (*id*)

List whitelisted IPs for the specified database

**Parameters id** : integer

The ID for the database.

**Returns id** : integer

The ID of this whitelisted IP address.

**remote\_host\_id** : integer

The ID of the database this rule is applied to.

**security\_group\_id** : string

The ID of the security group this rule is applied to.

**created\_at** : string/time

The time this rule was created.

**subnet\_mask** : string

The subnet mask that is allowed by this rule.

**updated\_at** : string/time

The time this rule was last updated.

**post\_whitelist\_ips** (*id*, *subnet\_mask*)

Whitelist an IP address

**Parameters id** : integer

The ID of the database this rule is applied to.

**subnet\_mask** : string

The subnet mask that is allowed by this rule.

**Returns id** : integer

The ID of this whitelisted IP address.

**remote\_host\_id** : integer

The ID of the database this rule is applied to.

**is\_active** : boolean

True if the rule is applied, false if it has been revoked.

**security\_group\_id** : string

The ID of the security group this rule is applied to.

**created\_at** : string/time

The time this rule was created.

**authorized\_by** : string

The user who authorized this rule.

**subnet\_mask** : string

The subnet mask that is allowed by this rule.

**updated\_at** : string/time

The time this rule was last updated.

## *Files*

**class Files** (*session*, *return\_type*=*'civis'*)

## **Methods**

<code>delete_projects(id, project_id)</code>	Remove a Data::S3File from a project
<code>delete_shares_groups(id, group_id)</code>	Revoke the permissions a group has on this object
<code>delete_shares_users(id, user_id)</code>	Revoke the permissions a user has on this object
<code>get(id)</code>	Get details about a file
<code>list_projects(id)</code>	List the projects a Data::S3File belongs to
<code>list_shares(id)</code>	List users and groups permissioned on this object
<code>post(name, **kwargs)</code>	Initiate an upload of a file into the platform
<code>put_projects(id, project_id)</code>	Add a Data::S3File to a project
<code>put_shares_groups(id, permission_level, ...)</code>	Set the permissions groups has on this object
<code>put_shares_users(id, permission_level, user_ids)</code>	Set the permissions users have on this object

**delete\_projects** (*id, project\_id*)

Remove a Data::S3File from a project

**Parameters** *id* : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**delete\_shares\_groups** (*id, group\_id*)

Revoke the permissions a group has on this object

**Parameters** *id* : integer

ID of the resource to be revoked

**group\_id** : integer

ID of the group

**Returns** None

Response code 204: success

**delete\_shares\_users** (*id, user\_id*)

Revoke the permissions a user has on this object

**Parameters** *id* : integer

ID of the resource to be revoked

**user\_id** : integer

ID of the user

**Returns** None

Response code 204: success

**get** (*id*)

Get details about a file

**Parameters** *id* : integer

The ID of the file object.

**Returns** *id* : integer

The ID of the file object.

**file\_url** : string

The URL that may be used to download the file.

**name** : string

The file name.

**expires\_at** : string/date-time

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.

**created\_at** : string/date-time

The date and time the file was created.

**download\_url** : string

A JSON string containing information about the URL of the file.

**file\_size** : integer

The file size.

**list\_projects** (*id*)

List the projects a Data::S3File belongs to

**Parameters** **id** : integer

The ID of the resource.

**Returns** **id** : integer

The ID for this project.

**description** : string

A description of the project

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**created\_at** : string/time

**users** : list:

```
Users who can see the project
- id : integer
  The ID of this user.
- online : boolean
  Whether this user is online.
- initials : string
  This user's initials.
- name : string
  This user's name.
- username : string
  This user's username.
```

**author** : dict:



```

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

```

**updated\_at** : string/time

**auto\_share** : boolean

**list\_shares** (*id*)

List users and groups permissioned on this object

**Parameters** *id* : integer

The ID of the object.

**Returns** **readers** : dict:

```

- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string

```

**owners** : dict:

```

- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string

```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```

- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string

```

**post** (*name*, **\*\*kwargs**)

Initiate an upload of a file into the platform

**Parameters** **name** : string

The file name.

**expires\_at** : string/date-time, 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.

**Returns** **id** : integer

The ID of the file object.

**upload\_url** : 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.

**name** : string

The file name.

**expires\_at** : string/date-time

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.

**created\_at** : string/date-time

The date and time the file was created.

**upload\_fields** : dict

A hash containing the form fields to be included with the POST request.

**file\_size** : integer

The file size.

**put\_projects** (*id*, *project\_id*)

Add a Data::S3File to a project

**Parameters** **id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**put\_shares\_groups** (*id*, *permission\_level*, *group\_ids*)

Set the permissions groups has on this object

**Parameters** **id** : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**group\_ids** : list

An array of one or more group IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_shares\_users** (*id*, *permission\_level*, *user\_ids*)

Set the permissions users have on this object

**Parameters** *id* : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**user\_ids** : list

An array of one or more user IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

## Imports

**class Imports** (*session*, *return\_type*='civis')

### Methods

<code>delete_files_runs(id, run_id)</code>	Cancel a run
<code>delete_projects(id, project_id)</code>	Remove a JobTypes::Import from a project
<code>delete_shares_groups(id, group_id)</code>	Revoke the permissions a group has on this object
<code>delete_shares_users(id, user_id)</code>	Revoke the permissions a user has on this object
<code>delete_syncs(id, sync_id)</code>	Delete a sync
<code>get(id)</code>	Get details about an import
<code>get_batches(id)</code>	Get details about a batch import
<code>get_files_runs(id, run_id)</code>	Check status of a run
<code>list(**kwargs)</code>	List imports
<code>list_batches(**kwargs)</code>	List batch imports
<code>list_files_runs(id, **kwargs)</code>	List runs for the given import
<code>list_projects(id)</code>	List the projects a JobTypes::Import belongs to
<code>list_runs(id)</code>	Get the run history of this import
<code>list_shares(id)</code>	List users and groups permissioned on this object
<code>post(is_outbound, sync_type, name, **kwargs)</code>	Create a new import configuration
<code>post_batches(remote_host_id, table, schema, ...)</code>	Upload multiple files to Redshift
<code>post_cancel(id)</code>	Cancel a run
<code>post_files(remote_host_id, schema, name, ...)</code>	Initate an import of a tabular file into the platform
<code>post_files_runs(id)</code>	Start a run
<code>post_runs(id)</code>	Run an import

Continued on next page

Table 4.7 – continued from previous page

<i>post_syncs</i> (id, source, destination, **kwargs)	Create a sync
<i>put</i> (id, is_outbound, sync_type, name, **kwargs)	Update an import
<i>put_archive</i> (id, status)	Update the archive status of this object
<i>put_projects</i> (id, project_id)	Add a JobTypes::Import to a project
<i>put_shares_groups</i> (id, permission_level, ...)	Set the permissions groups has on this object
<i>put_shares_users</i> (id, permission_level, user_ids)	Set the permissions users have on this object
<i>put_syncs</i> (id, sync_id, source, destination, ...)	Update a sync

**delete\_files\_runs** (*id*, *run\_id*)

Cancel a run

**Parameters** *id* : integer

The ID of the import.

**run\_id** : integer

The ID of the run.

**Returns** None

Response code 202: success

**delete\_projects** (*id*, *project\_id*)

Remove a JobTypes::Import from a project

**Parameters** *id* : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**delete\_shares\_groups** (*id*, *group\_id*)

Revoke the permissions a group has on this object

**Parameters** *id* : integer

ID of the resource to be revoked

**group\_id** : integer

ID of the group

**Returns** None

Response code 204: success

**delete\_shares\_users** (*id*, *user\_id*)

Revoke the permissions a user has on this object

**Parameters** *id* : integer

ID of the resource to be revoked

**user\_id** : integer

ID of the user

**Returns** None

Response code 204: success

**delete\_syncs** (*id*, *sync\_id*)

Delete a sync

**Parameters** *id* : integer

The ID of the import to fetch.

*sync\_id* : integer

The ID of the sync to fetch.

**Returns** None

Response code 204: success

**get** (*id*)

Get details about an import

**Parameters** *id* : integer

The ID for the import.

**Returns** *is\_outbound* : boolean

*syncs* : list:

```
List of syncs.
- id : integer
- advanced_options : dict::
  - wipe_destination_table : boolean
  - sortkey2 : string
  - soql_query : string
  - max_errors : integer
  - distkey : string
  - row_chunk_size : integer
  - invalid_char_replacement : string
  - last_modified_column : string
  - sortkey1 : string
  - truncate_long_lines : boolean
  - existing_table_rows : string
  - first_row_is_header : boolean
  - column_delimiter : string
  - partition_column_name : string
  - verify_table_row_counts : boolean
  - partition_table_name : string
  - partition_table_partition_column_min_name : string
  - partition_schema_name : string
  - mysql_catalog_matches_schema : boolean
  - contact_lists : string
  - sql_query : string
  - partition_table_partition_column_max_name : string
  - identity_column : string
  - export_action : string
- source : dict::
  - 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.
- destination : dict::
```

```
- path : string
    The schema.tablename to sync to.
```

**notifications : dict:**

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes,
↳ successfully.
```

**running\_as : dict:**

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

**user : dict:**

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

**destination : dict:**

```
- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific
↳ imports. For
    salesforce imports, the first and only element is the client,
↳ credential
```

```
    id.  
- name : string
```

**updated\_at** : string/date-time

**parent\_id** : integer

Parent id to trigger this import from

**name** : string

The name of the import.

**last\_run** : dict:

```
- id : integer  
- error : string  
    The error message for this run, if present.  
- started_at : string/time  
    The time that the run started.  
- created_at : string/time  
    The time that the run was queued.  
- state : string  
- finished_at : string/time  
    The time that the run completed.
```

**id** : integer

The ID for the import.

**next\_run\_at** : string/time

The time of the next scheduled run.

**archived** : string

The archival status of the requested object(s).

**sync\_type** : string

The type of sync to perform; one of Dbsync, AutoImport, SilverpopDataImport, SilverpopContactImport, GdocImport, GdocExport, and Salesforce.

**source** : dict:

```
- remote_host_id : integer  
- credential_id : integer  
- additional_credentials : list  
    Array that holds additional credentials used for specific_  
↳ imports. For  
    salesforce imports, the first and only element is the client_  
↳ credential  
    id.  
- name : string
```

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**created\_at** : string/date-time

**state** : string



**time\_zone** : string

The time zone of this import.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    hour
```

**get\_batches** (*id*)

Get details about a batch import

**Parameters** *id* : integer

The ID for the import.

**Returns** *id* : integer

The ID for the import.

**remote\_host\_id** : integer

The ID of the destination database host.

**table** : string

The destination table name, without the schema prefix. This table must already exist in Redshift.

**error** : string

The error returned by the run, if any.

**schema** : string

The destination schema name. This schema must already exist in Redshift.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run; one of “queued”, “running”, “succeeded”, “failed”, or “cancelled”.

**finished\_at** : string/time

The time the last run completed.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints. The object can still be queried directly by ID

**get\_files\_runs** (*id*, *run\_id*)

Check status of a run

**Parameters** *id* : integer

The ID of the import.

**run\_id** : integer

The ID of the run.

**Returns id** : integer

The ID of the run.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**import\_id** : integer

The ID of the import.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**list** (*\*\*kwargs*)

List imports

**Parameters type** : string, optional

If specified, return imports of these types. It accepts a comma-separated list, possible values are 'AutoImport', 'DbSync', 'Salesforce', 'GdocImport'.

**author** : string, optional

If specified, return imports from this author. It accepts a comma-separated list of author ids.

**destination** : string, optional

If specified, returns imports with one of these destinations. It accepts a comma-separated list of remote host ids.

**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'.

**archived** : string, optional

The archival status of the requested object(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`.

**iterator** : bool, optional

If `True`, return a generator to iterate over all responses. Use when more results than the maximum allowed by `limit` are needed. When `True`, `limit` and `page_num` are ignored. Defaults to `False`.

**Returns** **is\_outbound** : boolean

**updated\_at** : string/date-time

**destination** : dict:

```
- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific
    imports. For
    salesforce imports, the first and only element is the client
    credential
    id.
- name : string
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**user** : dict:

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

**id** : integer

The ID for the import.

**archived** : string

The archival status of the requested object(s).

**sync\_type** : string

The type of sync to perform; one of Dbsync, AutoImport, SilverpopDataImport, SilverpopContactImport, GdocImport, GdocExport, and Salesforce.

**source** : dict:

```
- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific_
↳ imports. For
    salesforce imports, the first and only element is the client_
↳ credential
    id.
- name : string
```

**created\_at** : string/date-time

**state** : string

**name** : string

The name of the import.

**time\_zone** : string

The time zone of this import.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↳ hour
```

**list\_batches** (\*\*kwargs)

List batch imports

**Parameters 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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns id** : integer

The ID for the import.

**remote\_host\_id** : integer

The ID of the destination database host.

**table** : string

The destination table name, without the schema prefix. This table must already exist in Redshift.

**error** : string

The error returned by the run, if any.

**schema** : string

The destination schema name. This schema must already exist in Redshift.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run; one of “queued”, “running”, “succeeded”, “failed”, or “cancelled”.

**finished\_at** : string/time

The time the last run completed.

**list\_files\_runs** (*id*, *\*\*kwargs*)

List runs for the given import

**Parameters id** : integer

The ID of the import.

**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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **id** : integer

The ID of the run.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**import\_id** : integer

The ID of the import.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**list\_projects** (*id*)

List the projects a JobTypes::Import belongs to

**Parameters** **id** : integer

The ID of the resource.

**Returns** **id** : integer

The ID for this project.

**description** : string

A description of the project

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**created\_at** : string/time

**users** : list:

```
Users who can see the project
- id : integer
  The ID of this user.
- online : boolean
  Whether this user is online.
- initials : string
  This user's initials.
- name : string
  This user's name.
- username : string
  This user's username.
```

**author** : dict:

```

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

```

**updated\_at** : string/time

**auto\_share** : boolean

**list\_runs** (*id*)

Get the run history of this import

**Parameters** *id* : integer

**Returns** *id* : integer

**error** : string

The error message for this run, if present.

**started\_at** : string/time

The time that the run started.

**created\_at** : string/time

The time that the run was queued.

**state** : string

**finished\_at** : string/time

The time that the run completed.

**list\_shares** (*id*)

List users and groups permissioned on this object

**Parameters** *id* : integer

The ID of the object.

**Returns** **readers** : dict:

```

- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string

```

**owners** : dict:

```

- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string

```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**post** (*is\_outbound*, *sync\_type*, *name*, *\*\*kwargs*)

Create a new import configuration

**Parameters** **is\_outbound** : boolean

**sync\_type** : string

The type of sync to perform; one of DbSync, AutoImport, SilverpopDataImport, SilverpopContactImport, GdocImport, GdocExport, and Salesforce.

**name** : string

The name of the import.

**notifications** : dict, optional:

```
- success_on : boolean
  If success email notifications are on
- stall_warning_minutes : integer
  Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
  Addresses to notify by e-mail when the job fails.
- success_email_subject : string
  Custom subject line for success e-mail.
- failure_on : boolean
  If failure email notifications are on
- urls : list
  URLs to receive a POST request at job completion
- success_email_body : string
  Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
  Addresses to notify by e-mail when the job completes,
↪successfully.
```

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**schedule** : dict, optional:

```
- scheduled : boolean
  If the object is scheduled
- scheduled_days : list
  Day based on numeric value starting at 0 for Sunday
```



```

- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↳hour

```

**source** : dict, optional:

```

- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific_
↳imports. For
    salesforce imports, the first and only element is the client_
↳credential
    id.

```

**destination** : dict, optional:

```

- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific_
↳imports. For
    salesforce imports, the first and only element is the client_
↳credential
    id.

```

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**parent\_id** : integer, optional

Parent id to trigger this import from

**time\_zone** : string, optional

The time zone of this import.

**Returns is\_outbound** : boolean

**syncs** : list:

```

List of syncs.
- id : integer
- advanced_options : dict::
    - wipe_destination_table : boolean
    - sortkey2 : string
    - soql_query : string
    - max_errors : integer
    - distkey : string
    - row_chunk_size : integer
    - invalid_char_replacement : string
    - last_modified_column : string
    - sortkey1 : string
    - truncate_long_lines : boolean

```

```
- existing_table_rows : string
- first_row_is_header : boolean
- column_delimiter : string
- partition_column_name : string
- verify_table_row_counts : boolean
- partition_table_name : string
- partition_table_partition_column_min_name : string
- partition_schema_name : string
- mysql_catalog_matches_schema : boolean
- contact_lists : string
- sql_query : string
- partition_table_partition_column_max_name : string
- identity_column : string
- export_action : string
- source : dict::
  - 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.
- destination : dict::
  - path : string
    The schema.tablename to sync to.
```

**notifications : dict:**

```
- success_on : boolean
  If success email notifications are on
- stall_warning_minutes : integer
  Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
  Addresses to notify by e-mail when the job fails.
- success_email_subject : string
  Custom subject line for success e-mail.
- failure_on : boolean
  If failure email notifications are on
- urls : list
  URLs to receive a POST request at job completion
- success_email_body : string
  Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
  Addresses to notify by e-mail when the job completes,
  ↪successfully.
```

**running\_as : dict:**

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

**user : dict:**

```

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

```

**destination** : dict:

```

- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific_
↳imports. For
    salesforce imports, the first and only element is the client_
↳credential
    id.
- name : string

```

**updated\_at** : string/date-time

**parent\_id** : integer

Parent id to trigger this import from

**name** : string

The name of the import.

**last\_run** : dict:

```

- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.

```

**id** : integer

The ID for the import.

**next\_run\_at** : string/time

The time of the next scheduled run.

**archived** : string

The archival status of the requested object(s).

**sync\_type** : string

The type of sync to perform; one of Dbsync, AutoImport, SilverpopDataImport, SilverpopContactImport, GdocImport, GdocExport, and Salesforce.

**source** : dict:

```
- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific_
↳ imports. For
    salesforce imports, the first and only element is the client_
↳ credential
    id.
- name : string
```

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**created\_at** : string/date-time

**state** : string

**time\_zone** : string

The time zone of this import.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↳ hour
```

**post\_batches** (*remote\_host\_id, table, schema, credential\_id, file\_ids, \*\*kwargs*)

Upload multiple files to Redshift

**Parameters** **remote\_host\_id** : integer

The ID of the destination database host.

**table** : string

The destination table name, without the schema prefix. This table must already exist in Redshift.

**schema** : string

The destination schema name. This schema must already exist in Redshift.

**credential\_id** : integer

The ID of the credentials to be used when performing the database import.

**file\_ids** : list

The file IDs for the import.

**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.

**column\_delimiter** : string, optional

The column delimiter for the file. Valid arguments are “comma”, “tab”, and “pipe”. If unspecified, defaults to “comma”.

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints. The object can still be queried directly by ID

**compression** : string, optional

The type of compression. Valid arguments are “gzip”, “zip”, and “none”. If unspecified, defaults to “gzip”.

**Returns id** : integer

The ID for the import.

**remote\_host\_id** : integer

The ID of the destination database host.

**table** : string

The destination table name, without the schema prefix. This table must already exist in Redshift.

**error** : string

The error returned by the run, if any.

**schema** : string

The destination schema name. This schema must already exist in Redshift.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run; one of “queued”, “running”, “succeeded”, “failed”, or “cancelled”.

**finished\_at** : string/time

The time the last run completed.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints. The object can still be queried directly by ID

**post\_cancel** (*id*)

Cancel a run

**Parameters id** : integer

The ID of the job.

**Returns id** : integer

The ID of the run.

**state** : string

The state of the run, one of ‘queued’, ‘running’ or ‘cancelled’.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**post\_files** (*remote\_host\_id*, *schema*, *name*, *credential\_id*, *\*\*kwargs*)

Initiate an import of a tabular file into the platform

**Parameters** **remote\_host\_id** : integer

The id of the destination database host.

**schema** : string

The schema of the destination table.

**name** : string

The name of the destination table.

**credential\_id** : integer

The id of the credentials to be used when performing the database import.

**sortkey2** : string, optional

The second column in a compound sortkey for the table.

**multipart** : boolean, optional

If true, the upload URI will require a *multipart/form-data* POST request. Defaults to false.

**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.

**max\_errors** : integer, optional

The maximum number of rows with errors to remove from the import before failing.

**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”.

**distkey** : string, optional

The column to use as the distkey for the table.

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints. The object can still be queried directly by ID

**sortkey1** : string, optional

The column to use as the sort key for the table.

**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”.

**Returns** **id** : integer

The id of the import.

**run\_uri** : 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.

**upload\_fields** : dict

If multipart was set to true, these fields should be included in the multipart upload.

**upload\_uri** : 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.

**post\_files\_runs** (*id*)

Start a run

**Parameters** **id** : integer

The ID of the import.

**Returns** **id** : integer

The ID of the run.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**import\_id** : integer

The ID of the import.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**post\_runs** (*id*)

Run an import

**Parameters** **id** : integer

The ID of the import to run.

**Returns** **run\_id** : integer

The ID of the new run triggered.

**post\_syncs** (*id, source, destination, \*\*kwargs*)

Create a sync

**Parameters** **id** : integer

**source** : dict:

```
- path : string
    The path of the dataset to sync from; for a database source,
    schema.tablename.
```

**destination** : dict:

```
- path : string
    The schema.tablename to sync to.
```

**advanced\_options** : dict, optional:

```
- wipe_destination_table : boolean
- sortkey2 : string
- soql_query : string
- max_errors : integer
- distkey : string
- row_chunk_size : integer
- invalid_char_replacement : string
- last_modified_column : string
- sortkey1 : string
- truncate_long_lines : boolean
- existing_table_rows : string
- first_row_is_header : boolean
- column_delimiter : string
- partition_column_name : string
- verify_table_row_counts : boolean
- partition_table_name : string
- partition_table_partition_column_min_name : string
- partition_schema_name : string
- mysql_catalog_matches_schema : boolean
- contact_lists : string
- sql_query : string
- partition_table_partition_column_max_name : string
- identity_column : string
- export_action : string
```

**Returns** id : integer

**advanced\_options** : dict:

```
- wipe_destination_table : boolean
- sortkey2 : string
- soql_query : string
- max_errors : integer
- distkey : string
- row_chunk_size : integer
- invalid_char_replacement : string
- last_modified_column : string
- sortkey1 : string
- truncate_long_lines : boolean
- existing_table_rows : string
- first_row_is_header : boolean
- column_delimiter : string
- partition_column_name : string
- verify_table_row_counts : boolean
- partition_table_name : string
- partition_table_partition_column_min_name : string
- partition_schema_name : string
```



```
- mysql_catalog_matches_schema : boolean
- contact_lists : string
- sql_query : string
- partition_table_partition_column_max_name : string
- identity_column : string
- export_action : string
```

**source** : dict:

```
- 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.
```

**destination** : dict:

```
- path : string
    The schema.tablename to sync to.
```

**put** (*id, is\_outbound, sync\_type, name, \*\*kwargs*)

Update an import

**Parameters** **id** : integer

The ID for the import.

**is\_outbound** : boolean

**sync\_type** : string

The type of sync to perform; one of Dbsync, AutoImport, SilverpopDataImport, SilverpopContactImport, GdocImport, GdocExport, and Salesforce.

**name** : string

The name of the import.

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↳hour
```

**source** : dict, optional:

```
- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific_
↳imports. For
    salesforce imports, the first and only element is the client_
↳credential
    id.
```

**destination** : dict, optional:

```
- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific_
↳imports. For
    salesforce imports, the first and only element is the client_
↳credential
    id.
```

**parent\_id** : integer, optional

Parent id to trigger this import from

**time\_zone** : string, optional

The time zone of this import.

**Returns is\_outbound** : boolean

**syncs** : list:

```
List of syncs.
- id : integer
- advanced_options : dict::
    - wipe_destination_table : boolean
    - sortkey2 : string
    - soql_query : string
    - max_errors : integer
    - distkey : string
    - row_chunk_size : integer
    - invalid_char_replacement : string
    - last_modified_column : string
    - sortkey1 : string
```

```

- truncate_long_lines : boolean
- existing_table_rows : string
- first_row_is_header : boolean
- column_delimiter : string
- partition_column_name : string
- verify_table_row_counts : boolean
- partition_table_name : string
- partition_table_partition_column_min_name : string
- partition_schema_name : string
- mysql_catalog_matches_schema : boolean
- contact_lists : string
- sql_query : string
- partition_table_partition_column_max_name : string
- identity_column : string
- export_action : string
- source : dict::
  - 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.
- destination : dict::
  - path : string
    The schema.tablename to sync to.

```

**notifications : dict:**

```

- success_on : boolean
  If success email notifications are on
- stall_warning_minutes : integer
  Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
  Addresses to notify by e-mail when the job fails.
- success_email_subject : string
  Custom subject line for success e-mail.
- failure_on : boolean
  If failure email notifications are on
- urls : list
  URLs to receive a POST request at job completion
- success_email_body : string
  Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
  Addresses to notify by e-mail when the job completes,
↪ successfully.

```

**running\_as : dict:**

```

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

```

**user** : dict:

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

**destination** : dict:

```
- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific_
↳ imports. For
    salesforce imports, the first and only element is the client_
↳ credential
    id.
- name : string
```

**updated\_at** : string/date-time

**parent\_id** : integer

Parent id to trigger this import from

**name** : string

The name of the import.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**id** : integer

The ID for the import.

**next\_run\_at** : string/time

The time of the next scheduled run.

**archived** : string

The archival status of the requested object(s).

**sync\_type** : string

The type of sync to perform; one of Dbsync, AutoImport, SilverpopDataImport, SilverpopContactImport, GdocImport, GdocExport, and Salesforce.

**source** : dict:

```
- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific_
↳ imports. For
    salesforce imports, the first and only element is the client_
↳ credential
    id.
- name : string
```

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**created\_at** : string/date-time

**state** : string

**time\_zone** : string

The time zone of this import.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↳ hour
```

**put\_archive** (*id, status*)

Update the archive status of this object

**Parameters** **id** : integer

The ID of the object.

**status** : boolean

The desired archived status of the object.

**Returns** **is\_outbound** : boolean

**syncs** : list:

```
List of syncs.
- id : integer
- advanced_options : dict::
    - wipe_destination_table : boolean
    - sortkey2 : string
    - soql_query : string
```

```
- max_errors : integer
- distkey : string
- row_chunk_size : integer
- invalid_char_replacement : string
- last_modified_column : string
- sortkey1 : string
- truncate_long_lines : boolean
- existing_table_rows : string
- first_row_is_header : boolean
- column_delimiter : string
- partition_column_name : string
- verify_table_row_counts : boolean
- partition_table_name : string
- partition_table_partition_column_min_name : string
- partition_schema_name : string
- mysql_catalog_matches_schema : boolean
- contact_lists : string
- sql_query : string
- partition_table_partition_column_max_name : string
- identity_column : string
- export_action : string
- source : dict::
  - 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.
- destination : dict::
  - path : string
    The schema.tablename to sync to.
```

**notifications : dict:**

```
- success_on : boolean
  If success email notifications are on
- stall_warning_minutes : integer
  Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
  Addresses to notify by e-mail when the job fails.
- success_email_subject : string
  Custom subject line for success e-mail.
- failure_on : boolean
  If failure email notifications are on
- urls : list
  URLs to receive a POST request at job completion
- success_email_body : string
  Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
  Addresses to notify by e-mail when the job completes,
  ↪ successfully.
```

**running\_as : dict:**

```
- id : integer
  The ID of this user.
- online : boolean
  Whether this user is online.
- initials : string
```

```

    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.

```

**user** : dict:

```

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

```

**destination** : dict:

```

- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific_
↳ imports. For
    salesforce imports, the first and only element is the client_
↳ credential
    id.
- name : string

```

**updated\_at** : string/date-time

**parent\_id** : integer

Parent id to trigger this import from

**name** : string

The name of the import.

**last\_run** : dict:

```

- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.

```

**id** : integer

The ID for the import.

**next\_run\_at** : string/time

The time of the next scheduled run.

**archived** : string

The archival status of the requested object(s).

**sync\_type** : string

The type of sync to perform; one of Dbsync, AutoImport, SilverpopDataImport, SilverpopContactImport, GdocImport, GdocExport, and Salesforce.

**source** : dict:

```
- remote_host_id : integer
- credential_id : integer
- additional_credentials : list
    Array that holds additional credentials used for specific
    ↪ imports. For
    ↪ salesforce imports, the first and only element is the client
    ↪ credential
    ↪ id.
- name : string
```

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints. The object can still be queried directly by ID

**created\_at** : string/date-time

**state** : string

**time\_zone** : string

The time zone of this import.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    ↪ hour
```

**put\_projects** (*id*, *project\_id*)

Add a JobTypes::Import to a project

**Parameters** *id* : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**put\_shares\_groups** (*id*, *permission\_level*, *group\_ids*)

Set the permissions groups has on this object



**Parameters** `id` : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**group\_ids** : list

An array of one or more group IDs

**Returns** `readers` : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_shares\_users** (*id*, *permission\_level*, *user\_ids*)

Set the permissions users have on this object

**Parameters** `id` : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**user\_ids** : list

An array of one or more user IDs

**Returns** `readers` : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_syncs** (*id*, *sync\_id*, *source*, *destination*, *\*\*kwargs*)

Update a sync

**Parameters** `id` : integer

The ID of the import to fetch.

**sync\_id** : integer

The ID of the sync to fetch.

**source** : dict:

```
- path : string
  The path of the dataset to sync from; for a database source,
  schema.tablename.
```

**destination** : dict:

```
- path : string
  The schema.tablename to sync to.
```

**advanced\_options** : dict, optional:

```

- wipe_destination_table : boolean
- sortkey2 : string
- soql_query : string
- max_errors : integer
- distkey : string
- row_chunk_size : integer
- invalid_char_replacement : string
- last_modified_column : string
- sortkey1 : string
- truncate_long_lines : boolean
- existing_table_rows : string
- first_row_is_header : boolean
- column_delimiter : string
- partition_column_name : string
- verify_table_row_counts : boolean
- partition_table_name : string
- partition_table_partition_column_min_name : string
- partition_schema_name : string
- mysql_catalog_matches_schema : boolean
- contact_lists : string
- sql_query : string
- partition_table_partition_column_max_name : string
- identity_column : string
- export_action : string

```

**Returns** `id` : integer

**advanced\_options** : dict:

```

- wipe_destination_table : boolean
- sortkey2 : string
- soql_query : string
- max_errors : integer
- distkey : string
- row_chunk_size : integer
- invalid_char_replacement : string
- last_modified_column : string
- sortkey1 : string
- truncate_long_lines : boolean
- existing_table_rows : string
- first_row_is_header : boolean
- column_delimiter : string
- partition_column_name : string
- verify_table_row_counts : boolean
- partition_table_name : string
- partition_table_partition_column_min_name : string
- partition_schema_name : string
- mysql_catalog_matches_schema : boolean
- contact_lists : string
- sql_query : string
- partition_table_partition_column_max_name : string
- identity_column : string
- export_action : string

```

**source** : dict:

```

- 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.
```

**destination** : dict:

```
- path : string
    The schema.tablename to sync to.
```

## Jobs

**class Jobs** (*session*, *return\_type*='civis')

### Methods

<i>delete_projects</i> (id, project_id)	Remove a Job from a project
<i>delete_shares_groups</i> (id, group_id)	Revoke the permissions a group has on this object
<i>delete_shares_users</i> (id, user_id)	Revoke the permissions a user has on this object
<i>get</i> (id)	Show basic job info
<i>get_runs</i> (id, run_id)	Check status of a job
<i>list</i> (**kwargs)	List jobs
<i>list_children</i> (id)	Show nested tree of children that this job triggers
<i>list_parents</i> (id)	Show chain of parents as a list that this job triggers from
<i>list_projects</i> (id)	List the projects a Job belongs to
<i>list_shares</i> (id)	List users and groups permissioned on this object
<i>post_runs</i> (id)	Run a job
<i>post_trigger_email</i> (id)	Generate and retrieve trigger email address
<i>put_projects</i> (id, project_id)	Add a Job to a project
<i>put_shares_groups</i> (id, permission_level, ...)	Set the permissions groups has on this object
<i>put_shares_users</i> (id, permission_level, user_ids)	Set the permissions users have on this object

**delete\_projects** (*id*, *project\_id*)

Remove a Job from a project

**Parameters** *id* : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**delete\_shares\_groups** (*id*, *group\_id*)

Revoke the permissions a group has on this object

**Parameters** *id* : integer

ID of the resource to be revoked

**group\_id** : integer

ID of the group

**Returns** None

Response code 204: success

**delete\_shares\_users** (*id*, *user\_id*)

Revoke the permissions a user has on this object

**Parameters** *id* : integer

ID of the resource to be revoked

**user\_id** : integer

ID of the user

**Returns** None

Response code 204: success

**get** (*id*)

Show basic job info

**Parameters** *id* : integer

The ID for this job.

**Returns** *id* : integer

**updated\_at** : string/date-time

**runs** : list:

Information about the most recent runs of the job.

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**type** : string

**name** : string

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**created\_at** : string/date-time

**state** : string

Whether the job is idle, queued, running, cancelled, or failed.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
```

```
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**archived** : string

The archival status of the requested object(s).

**get\_runs** (*id*, *run\_id*)

Check status of a job

**Parameters** **id** : integer

The ID of the Job.

**run\_id** : integer

The ID of the Run.

**Returns** **id** : integer

**error** : string

The error message for this run, if present.

**started\_at** : string/time

The time that the run started.

**created\_at** : string/time

The time that the run was queued.

**state** : string

**finished\_at** : string/time

The time that the run completed.

**list** (*\*\*kwargs*)

List jobs

**Parameters** **limit** : integer, optional

The maximum number of jobs to return.

**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.

**archived** : string, optional

The archival status of the requested object(s).

**Returns** **id** : integer

**updated\_at** : string/date-time

**type** : string

**name** : string

**created\_at** : string/date-time

**state** : string

Whether the job is idle, queued, running, cancelled, or failed.

**last\_run** : dict:

```
- id : integer
- error : string
  The error message for this run, if present.
- started_at : string/time
  The time that the run started.
- created_at : string/time
  The time that the run was queued.
- state : string
- finished_at : string/time
  The time that the run completed.
```

**archived** : string

The archival status of the requested object(s).

**list\_children** (*id*)

Show nested tree of children that this job triggers

**Parameters** **id** : integer

The ID for this job.

**Returns** **id** : integer

**updated\_at** : string/date-time

**children** : list

**runs** : list:

```
- id : integer
- error : string
  The error message for this run, if present.
- started_at : string/time
  The time that the run started.
- created_at : string/time
  The time that the run was queued.
- state : string
- finished_at : string/time
  The time that the run completed.
```

**type** : string

**name** : string

**created\_at** : string/date-time

**state** : string

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**list\_parents** (*id*)

Show chain of parents as a list that this job triggers from

**Parameters** **id** : integer

The ID for this job.

**Returns** **id** : integer

**updated\_at** : string/date-time

**runs** : list:

```
Information about the most recent runs of the job.
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**type** : string

**name** : string

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**created\_at** : string/date-time

**state** : string

Whether the job is idle, queued, running, cancelled, or failed.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
```



```

    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.

```

**archived** : string

The archival status of the requested object(s).

**list\_projects** (*id*)

List the projects a Job belongs to

**Parameters** **id** : integer

The ID of the resource.

**Returns** **id** : integer

The ID for this project.

**description** : string

A description of the project

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**created\_at** : string/time

**users** : list:

```

Users who can see the project
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.

```

**author** : dict:

```

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

```

**updated\_at** : string/time

**auto\_share** : boolean

**list\_shares** (*id*)

List users and groups permissioned on this object

**Parameters** **id** : integer

The ID of the object.

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**post\_runs** (*id*)

Run a job

**Parameters** **id** : integer

The ID for this job.

**Returns** **id** : integer

**error** : string

The error message for this run, if present.

**started\_at** : string/time

The time that the run started.

**created\_at** : string/time

The time that the run was queued.

**state** : string

**finished\_at** : string/time

The time that the run completed.

**post\_trigger\_email** (*id*)

Generate and retrieve trigger email address

**Parameters** **id** : integer

The ID for this job.

**Returns** **trigger\_email** : string

Email address which may be used to trigger this job to run.

**put\_projects** (*id*, *project\_id*)

Add a Job to a project

**Parameters** **id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**put\_shares\_groups** (*id*, *permission\_level*, *group\_ids*)

Set the permissions groups has on this object

**Parameters** **id** : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**group\_ids** : list

An array of one or more group IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
```

```
- id : integer
- name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_shares\_users** (*id*, *permission\_level*, *user\_ids*)

Set the permissions users have on this object

**Parameters** *id* : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**user\_ids** : list

An array of one or more user IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

## Models

**class Models** (*session, return\_type='civis'*)

### Methods

<code>delete_builds(id, build_id)</code>	Cancel a build
<code>delete_projects(id, project_id)</code>	Remove a models from a project
<code>delete_shares_groups(id, group_id)</code>	Revoke the permissions a group has on this object
<code>delete_shares_users(id, user_id)</code>	Revoke the permissions a user has on this object
<code>get(id)</code>	Retrieve model configuration
<code>get_builds(id, build_id)</code>	Check status of a build
<code>list(**kwargs)</code>	List models
<code>list_builds(id, **kwargs)</code>	List builds for the given model
<code>list_projects(id)</code>	List the projects a models belongs to
<code>list_schedules(id)</code>	Show the model build schedule
<code>list_shares(id)</code>	List users and groups permissioned on this object
<code>list_types()</code>	List all available model types
<code>patch(id, **kwargs)</code>	Update model configuration
<code>post(**kwargs)</code>	Create new configuration for a model
<code>post_builds(id)</code>	Start a build
<code>put_archive(id, status)</code>	Update the archive status of this object
<code>put_predictions(id, primary_key, table_name, ...)</code>	Add a table on which to apply the predictive model
<code>put_projects(id, project_id)</code>	Add a models to a project
<code>put_schedules(id, schedule)</code>	Schedule the model build
<code>put_shares_groups(id, permission_level, ...)</code>	Set the permissions groups has on this object
<code>put_shares_users(id, permission_level, user_ids)</code>	Set the permissions users have on this object

**delete\_builds** (*id, build\_id*)

Cancel a build

**Parameters** **id** : integer

The ID of the model.

**build\_id** : integer

The ID of the build.

**Returns** None

Response code 202: success

**delete\_projects** (*id*, *project\_id*)

Remove a models from a project

**Parameters** *id* : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**delete\_shares\_groups** (*id*, *group\_id*)

Revoke the permissions a group has on this object

**Parameters** *id* : integer

ID of the resource to be revoked

**group\_id** : integer

ID of the group

**Returns** None

Response code 204: success

**delete\_shares\_users** (*id*, *user\_id*)

Revoke the permissions a user has on this object

**Parameters** *id* : integer

ID of the resource to be revoked

**user\_id** : integer

ID of the user

**Returns** None

Response code 204: success

**get** (*id*)

Retrieve model configuration

**Parameters** *id* : integer

The ID of the model.

**Returns** **description** : string

A description of the model.

**number\_of\_folds** : integer

Number of folds for cross validation. Default value is 5.

**notifications** : dict:

- **success\_on** : boolean  
If success email notifications are on
- **stall\_warning\_minutes** : integer  
Stall warning emails will be sent after this amount of minutes.
- **failure\_email\_addresses** : list  
Addresses to notify by e-mail when the job fails.

```

- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes.
↳successfully.

```

**limiting\_sql** : string

A custom SQL WHERE clause used to filter the rows used to build the model. (e.g., “id > 105”).

**predictions** : list:

```

The tables upon which the model will be applied.
- id : integer
    The ID of the model to which to apply the prediction.
- state : string
    The status of the prediction. One of: "succeeded", "failed",
↳"queued",
    or "running", or "idle", if no build has been attempted.
- output_table : string
    The qualified name of the table to be created which will
↳contain the
    model's predictions.
- limiting_sql : string
    A SQL WHERE clause used to scope the rows to be predicted.
- primary_key : list
    The primary key or composite keys of the table being predicted.
- table_name : string
    The qualified name of the table on which to apply the
↳predictive model.
- schedule : dict::
    - scheduled : boolean
        If the object is scheduled
    - scheduled_days : list
        Day based on numeric value starting at 0 for Sunday
    - scheduled_minutes : list
        Minutes of the day it is scheduled on
    - scheduled_hours : list
        Hours of the day it is scheduled on
    - scheduled_runs_per_hour : integer
        Alternative to scheduled minutes, number of times to run
↳per hour

```

**parent\_id** : integer

The ID of the parent job that will trigger this model.

**database\_id** : integer

The ID of the database holding the training set table used to build the model.

**excluded\_columns** : list

A list of columns which will be considered ineligible to be independent variables.

**user** : dict:

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

**dependent\_variable** : string

The dependent variable of the training dataset.

**active\_build\_id** : integer

The ID of the current active build, the build used to score predictions.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**interaction\_terms** : boolean

Whether to search for interaction terms.

**builds** : list:

```
A list of trained models available for making predictions.
- id : integer
    The ID of the model build.
- root_mean_squared_error : number/float
    A key metric for continuous models. Nil for other model types.
- roc_auc : number/float
    A key metric for binary, multinomial, and ordinal models. Nil_
↪for other
    model types.
- r_squared_error : number/float
    A key metric for continuous models. Nil for other model types.
- name : string
    The name of the model build.
- description : string
    A description of the model build.
- created_at : string
    The time the model build was created.
```



**created\_at** : string/date-time

The time the model was created.

**primary\_key** : string

The unique ID (primary key) of the training dataset.

**time\_zone** : string

The time zone of this model.

**current\_build\_exception** : string

Exception message, if applicable, of the current model build.

**updated\_at** : string/date-time

The time the model was updated.

**box\_cox\_transformation** : boolean

Whether to transform data so that it assumes a normal distribution. Valid only with continuous models.

**model\_type\_id** : integer

The ID of the model's type.

**last\_output\_location** : string

The output JSON for the last build.

**dependent\_variable\_order** : list

The order of dependent variables, especially useful for Ordinal Modeling.

**last\_run** : dict:

```
- id : integer
- error : string
  The error message for this run, if present.
- started_at : string/time
  The time that the run started.
- created_at : string/time
  The time that the run was queued.
- state : string
- finished_at : string/time
  The time that the run completed.
```

**id** : integer

The ID of the model.

**cross\_validation\_parameters** : dict

Cross validation parameter grid for tree methods, e.g. {"n\_estimators": [100, 200, 500], "learning\_rate": [0.01, 0.1], "max\_depth": [2, 3]}.

**archived** : string

The archival status of the requested object(s).

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints. The object can still be queried directly by ID

**credential\_id** : integer

The ID of the credential used to read the target table. Defaults to the user's default credential.

**running\_as** : dict:

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

**current\_build\_state** : string

The status of the current model build. One of “succeeded”, “failed”, “queued”, or “running,” or “idle”, if no build has been attempted.

**table\_name** : string

The qualified name of the table containing the training set from which to build the model.

**model\_name** : string

The name of the model.

**get\_builds** (*id*, *build\_id*)

Check status of a build

**Parameters** **id** : integer

The ID of the model.

**build\_id** : integer

The ID of the build.

**Returns** **description** : string

A description of the model build.

**root\_mean\_squared\_error** : number/float

A key metric for continuous models. Nil for other model types.

**r\_squared\_error** : number/float

A key metric for continuous models. Nil for other model types.

**state** : string

The state of the model build. one of ‘queued’ ‘running’ ‘succeeded’ ‘failed’ or ‘cancelled’.

**id** : integer

The ID of the model build.

**roc\_auc** : number/float

A key metric for binary, multinomial, and ordinal models. Nil for other model types.

**transformation\_metadata** : string

A string representing the full JSON output of the metadata for transformation of column names

**name** : string

The name of the model build.

**error** : string

The error, if any, returned by the build.

**created\_at** : string

The time the model build was created.

**output** : string

A string representing the JSON output for the specified build. Only present when smaller than 10KB in size.

**output\_location** : string

A URL representing the location of the full JSON output for the specified build. The URL link will be valid for 5 minutes.

**list** (*\*\*kwargs*)

List models

**Parameters** **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”).

**author** : string, optional

If specified, return models from this author. It accepts a comma-separated list of author ids.

**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’.

**archived** : string, optional

The archival status of the requested object(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`, `model_name`, `created_at`, `name`, `last_run.updated_at`.

**order\_dir** : string, optional

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

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by `limit` are needed. When True, `limit` and `page_num` are ignored. Defaults to False.

**Returns** **description** : string

A description of the model.

**number\_of\_folds** : integer

Number of folds for cross validation. Default value is 5.

**current\_build\_exception** : string

Exception message, if applicable, of the current model build.

**limiting\_sql** : string

A custom SQL WHERE clause used to filter the rows used to build the model. (e.g., `"id > 105"`).

**predictions** : list:

```
The tables upon which the model will be applied.
- id : integer
    The ID of the model to which to apply the prediction.
- state : string
    The status of the prediction. One of: "succeeded", "failed",
    ↪ "queued",
    or "running", or "idle", if no build has been attempted.
- output_table : string
    The qualified name of the table to be created which will_
    ↪ contain the
    model's predictions.
- limiting_sql : string
    A SQL WHERE clause used to scope the rows to be predicted.
- primary_key : list
    The primary key or composite keys of the table being predicted.
- table_name : string
    The qualified name of the table on which to apply the_
    ↪ predictive model.
```

**parent\_id** : integer

The ID of the parent job that will trigger this model.

**user** : dict:

```
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
```

```
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.
```

**excluded\_columns** : list

A list of columns which will be considered ineligible to be independent variables.

**database\_id** : integer

The ID of the database holding the training set table used to build the model.

**dependent\_variable** : string

The dependent variable of the training dataset.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    hour
```

**interaction\_terms** : boolean

Whether to search for interaction terms.

**builds** : list:

```
A list of trained models available for making predictions.
- id : integer
    The ID of the model build.
- root_mean_squared_error : number/float
    A key metric for continuous models. Nil for other model types.
- roc_auc : number/float
    A key metric for binary, multinomial, and ordinal models. Nil
    for other
    model types.
- r_squared_error : number/float
    A key metric for continuous models. Nil for other model types.
- name : string
    The name of the model build.
- description : string
    A description of the model build.
- created_at : string
    The time the model build was created.
```

**created\_at** : string/date-time

The time the model was created.

**primary\_key** : string

The unique ID (primary key) of the training dataset.

**time\_zone** : string

The time zone of this model.

**updated\_at** : string/date-time

The time the model was updated.

**box\_cox\_transformation** : boolean

Whether to transform data so that it assumes a normal distribution. Valid only with continuous models.

**model\_type\_id** : integer

The ID of the model's type.

**last\_output\_location** : string

The output JSON for the last build.

**dependent\_variable\_order** : list

The order of dependent variables, especially useful for Ordinal Modeling.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**id** : integer

The ID of the model.

**cross\_validation\_parameters** : dict

Cross validation parameter grid for tree methods, e.g. {"n\_estimators": [100, 200, 500], "learning\_rate": [0.01, 0.1], "max\_depth": [2, 3]}.

**archived** : string

The archival status of the requested object(s).

**credential\_id** : integer

The ID of the credential used to read the target table. Defaults to the user's default credential.

**current\_build\_state** : string

The status of the current model build. One of "succeeded", "failed", "queued", or "running," or "idle", if no build has been attempted.

**table\_name** : string

The qualified name of the table containing the training set from which to build the model.

**model\_name** : string

The name of the model.

**list\_builds** (*id*, *\*\*kwargs*)

List builds for the given model

**Parameters** **id** : integer

The ID of the model.

**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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **description** : string

A description of the model build.

**root\_mean\_squared\_error** : number/float

A key metric for continuous models. Nil for other model types.

**r\_squared\_error** : number/float

A key metric for continuous models. Nil for other model types.

**state** : string

The state of the model build. one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**id** : integer

The ID of the model build.

**roc\_auc** : number/float

A key metric for binary, multinomial, and ordinal models. Nil for other model types.

**transformation\_metadata** : string

A string representing the full JSON output of the metadata for transformation of column names

**name** : string

The name of the model build.

**error** : string

The error, if any, returned by the build.

**created\_at** : string

The time the model build was created.

**output** : string

A string representing the JSON output for the specified build. Only present when smaller than 10KB in size.

**output\_location** : string

A URL representing the location of the full JSON output for the specified build. The URL link will be valid for 5 minutes.

**list\_projects** (*id*)

List the projects a models belongs to

**Parameters** **id** : integer

The ID of the resource.

**Returns** **id** : integer

The ID for this project.

**description** : string

A description of the project

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**created\_at** : string/time

**users** : list:

```
Users who can see the project
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.
```

**author** : dict:

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



```
- username : string
    This user's username.
```

**updated\_at** : string/time

**auto\_share** : boolean

**list\_schedules** (*id*)

Show the model build schedule

**Parameters** **id** : integer

The ID of the model associated with this schedule.

**Returns** **id** : integer

The ID of the model associated with this schedule.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
    hour
```

**list\_shares** (*id*)

List users and groups permissioned on this object

**Parameters** **id** : integer

The ID of the object.

**Returns** **readers** : dict:

```
- users : list::
    - id : integer
    - name : string
- groups : list::
    - id : integer
    - name : string
```

**owners** : dict:

```
- users : list::
    - id : integer
    - name : string
- groups : list::
    - id : integer
    - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**list\_types()**

List all available model types

**Returns** **id** : integer

The ID of the model type.

**int\_allowed** : boolean

Whether this model type supports searching for interaction terms.

**dv\_type** : string

The type of dependent variable predicted by the model.

**algorithm** : string

The name of the algorithm used to train the model.

**patch** (*id*, *\*\*kwargs*)

Update model configuration

**Parameters** **id** : integer

The ID of the model.

**description** : string, optional

A description of the model.

**number\_of\_folds** : integer, optional

Number of folds for cross validation. Default value is 5.

**notifications** : dict, optional:

```
- success_on : boolean
  If success email notifications are on
- stall_warning_minutes : integer
  Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
  Addresses to notify by e-mail when the job fails.
- success_email_subject : string
  Custom subject line for success e-mail.
- failure_on : boolean
  If failure email notifications are on
- urls : list
  URLs to receive a POST request at job completion
- success_email_body : string
  Custom body text for success e-mail, written in Markdown.
```

```
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**box\_cox\_transformation** : boolean, optional

Whether to transform data so that it assumes a normal distribution. Valid only with continuous models.

**limiting\_sql** : string, optional

A custom SQL WHERE clause used to filter the rows used to build the model. (e.g., “id > 105”).

**model\_type\_id** : integer, optional

The ID of the model’s type.

**dependent\_variable\_order** : list, optional

The order of dependent variables, especially useful for Ordinal Modeling.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this model.

**database\_id** : integer, optional

The ID of the database holding the training set table used to build the model.

**excluded\_columns** : list, optional

A list of columns which will be considered ineligible to be independent variables.

**time\_zone** : string, optional

The time zone of this model.

**dependent\_variable** : string, optional

The dependent variable of the training dataset.

**cross\_validation\_parameters** : dict, optional

Cross validation parameter grid for tree methods, e.g. {“n\_estimators”: [100, 200, 500], “learning\_rate”: [0.01, 0.1], “max\_depth”: [2, 3]}.

**active\_build\_id** : integer, optional

The ID of the current active build, the build used to score predictions.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**interaction\_terms** : boolean, optional

Whether to search for interaction terms.

**credential\_id** : integer, optional

The ID of the credential used to read the target table. Defaults to the user's default credential.

**primary\_key** : string, optional

The unique ID (primary key) of the training dataset.

**table\_name** : string, optional

The qualified name of the table containing the training set from which to build the model.

**model\_name** : string, optional

The name of the model.

**Returns** None

Response code 204: success

**post** (*\*\*kwargs*)

Create new configuration for a model

**Parameters** **description** : string, optional

A description of the model.

**number\_of\_folds** : integer, optional

Number of folds for cross validation. Default value is 5.

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪ successfully.
```

**box\_cox\_transformation** : boolean, optional

Whether to transform data so that it assumes a normal distribution. Valid only with continuous models.

**limiting\_sql** : string, optional

A custom SQL WHERE clause used to filter the rows used to build the model. (e.g., "id > 105").

**model\_type\_id** : integer, optional

The ID of the model's type.

**dependent\_variable\_order** : list, optional

The order of dependent variables, especially useful for Ordinal Modeling.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this model.

**database\_id** : integer, optional

The ID of the database holding the training set table used to build the model.

**excluded\_columns** : list, optional

A list of columns which will be considered ineligible to be independent variables.

**time\_zone** : string, optional

The time zone of this model.

**dependent\_variable** : string, optional

The dependent variable of the training dataset.

**cross\_validation\_parameters** : dict, optional

Cross validation parameter grid for tree methods, e.g. {"n\_estimators": [100, 200, 500], "learning\_rate": [0.01, 0.1], "max\_depth": [2, 3]}.

**active\_build\_id** : integer, optional

The ID of the current active build, the build used to score predictions.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**interaction\_terms** : boolean, optional

Whether to search for interaction terms.

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints. The object can still be queried directly by ID

**credential\_id** : integer, optional

The ID of the credential used to read the target table. Defaults to the user's default credential.

**primary\_key** : string, optional

The unique ID (primary key) of the training dataset.

**table\_name** : string, optional

The qualified name of the table containing the training set from which to build the model.

**model\_name** : string, optional

The name of the model.

**Returns description** : string

A description of the model.

**number\_of\_folds** : integer

Number of folds for cross validation. Default value is 5.

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**limiting\_sql** : string

A custom SQL WHERE clause used to filter the rows used to build the model. (e.g., “id > 105”).

**predictions** : list:

```
The tables upon which the model will be applied.
- id : integer
    The ID of the model to which to apply the prediction.
- state : string
    The status of the prediction. One of: "succeeded", "failed",
↪"queued",
    or "running,"or "idle", if no build has been attempted.
- output_table : string
    The qualified name of the table to be created which will_
↪contain the
    model's predictions.
- limiting_sql : string
    A SQL WHERE clause used to scope the rows to be predicted.
- primary_key : list
    The primary key or composite keys of the table being predicted.
- table_name : string
    The qualified name of the table on which to apply the_
↪predictive model.
```

```

- schedule : dict::
  - scheduled : boolean
    If the object is scheduled
  - scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
  - scheduled_minutes : list
    Minutes of the day it is scheduled on
  - scheduled_hours : list
    Hours of the day it is scheduled on
  - scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run_
↪per hour

```

**parent\_id** : integer

The ID of the parent job that will trigger this model.

**database\_id** : integer

The ID of the database holding the training set table used to build the model.

**excluded\_columns** : list

A list of columns which will be considered ineligible to be independent variables.

**user** : dict:

```

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

```

**dependent\_variable** : string

The dependent variable of the training dataset.

**active\_build\_id** : integer

The ID of the current active build, the build used to score predictions.

**schedule** : dict:

```

- scheduled : boolean
  If the object is scheduled
- scheduled_days : list
  Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
  Minutes of the day it is scheduled on
- scheduled_hours : list
  Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
  Alternative to scheduled minutes, number of times to run per_
↪hour

```

**interaction\_terms** : boolean

Whether to search for interaction terms.

**builds** : list:

```
A list of trained models available for making predictions.
- id : integer
  The ID of the model build.
- root_mean_squared_error : number/float
  A key metric for continuous models. Nil for other model types.
- roc_auc : number/float
  A key metric for binary, multinomial, and ordinal models. Nil
  for other
  model types.
- r_squared_error : number/float
  A key metric for continuous models. Nil for other model types.
- name : string
  The name of the model build.
- description : string
  A description of the model build.
- created_at : string
  The time the model build was created.
```

**created\_at** : string/date-time

The time the model was created.

**primary\_key** : string

The unique ID (primary key) of the training dataset.

**time\_zone** : string

The time zone of this model.

**current\_build\_exception** : string

Exception message, if applicable, of the current model build.

**updated\_at** : string/date-time

The time the model was updated.

**box\_cox\_transformation** : boolean

Whether to transform data so that it assumes a normal distribution. Valid only with continuous models.

**model\_type\_id** : integer

The ID of the model's type.

**last\_output\_location** : string

The output JSON for the last build.

**dependent\_variable\_order** : list

The order of dependent variables, especially useful for Ordinal Modeling.

**last\_run** : dict:

```
- id : integer
- error : string
  The error message for this run, if present.
- started_at : string/time
```



```

    The time that the run started.
-   created_at : string/time
    The time that the run was queued.
-   state : string
-   finished_at : string/time
    The time that the run completed.

```

**id** : integer

The ID of the model.

**cross\_validation\_parameters** : dict

Cross validation parameter grid for tree methods, e.g. {"n\_estimators": [100, 200, 500], "learning\_rate": [0.01, 0.1], "max\_depth": [2, 3]}.

**archived** : string

The archival status of the requested object(s).

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints. The object can still be queried directly by ID

**credential\_id** : integer

The ID of the credential used to read the target table. Defaults to the user's default credential.

**running\_as** : dict:

```

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

```

**current\_build\_state** : string

The status of the current model build. One of "succeeded", "failed", "queued", or "running," or "idle", if no build has been attempted.

**table\_name** : string

The qualified name of the table containing the training set from which to build the model.

**model\_name** : string

The name of the model.

**post\_builds** (*id*)

Start a build

**Parameters id** : integer

The ID of the model.

**Returns description** : string

A description of the model build.

**root\_mean\_squared\_error** : number/float

A key metric for continuous models. Nil for other model types.

**r\_squared\_error** : number/float

A key metric for continuous models. Nil for other model types.

**state** : string

The state of the model build. one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**id** : integer

The ID of the model build.

**roc\_auc** : number/float

A key metric for binary, multinomial, and ordinal models. Nil for other model types.

**transformation\_metadata** : string

A string representing the full JSON output of the metadata for transformation of column names

**name** : string

The name of the model build.

**error** : string

The error, if any, returned by the build.

**created\_at** : string

The time the model build was created.

**output** : string

A string representing the JSON output for the specified build. Only present when smaller than 10KB in size.

**output\_location** : string

A URL representing the location of the full JSON output for the specified build. The URL link will be valid for 5 minutes.

**put\_archive** (*id, status*)

Update the archive status of this object

**Parameters id** : integer

The ID of the object.

**status** : boolean

The desired archived status of the object.

**Returns description** : string

A description of the model.

**number\_of\_folds** : integer

Number of folds for cross validation. Default value is 5.

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes
    successfully.
```

**limiting\_sql** : string

A custom SQL WHERE clause used to filter the rows used to build the model. (e.g., “id > 105”).

**predictions** : list:

```
The tables upon which the model will be applied.
- id : integer
    The ID of the model to which to apply the prediction.
- state : string
    The status of the prediction. One of: "succeeded", "failed",
    "queued",
    or "running", or "idle", if no build has been attempted.
- output_table : string
    The qualified name of the table to be created which will
    contain the
    model's predictions.
- limiting_sql : string
    A SQL WHERE clause used to scope the rows to be predicted.
- primary_key : list
    The primary key or composite keys of the table being predicted.
- table_name : string
    The qualified name of the table on which to apply the
    predictive model.
- schedule : dict::
    - scheduled : boolean
        If the object is scheduled
    - scheduled_days : list
        Day based on numeric value starting at 0 for Sunday
    - scheduled_minutes : list
        Minutes of the day it is scheduled on
    - scheduled_hours : list
        Hours of the day it is scheduled on
    - scheduled_runs_per_hour : integer
        Alternative to scheduled minutes, number of times to run
    per hour
```

**parent\_id** : integer

The ID of the parent job that will trigger this model.

**database\_id** : integer

The ID of the database holding the training set table used to build the model.

**excluded\_columns** : list

A list of columns which will be considered ineligible to be independent variables.

**user** : dict:

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

**dependent\_variable** : string

The dependent variable of the training dataset.

**active\_build\_id** : integer

The ID of the current active build, the build used to score predictions.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**interaction\_terms** : boolean

Whether to search for interaction terms.

**builds** : list:

```
A list of trained models available for making predictions.
- id : integer
    The ID of the model build.
- root_mean_squared_error : number/float
    A key metric for continuous models. Nil for other model types.
- roc_auc : number/float
    A key metric for binary, multinomial, and ordinal models. Nil_
↪for other
    model types.
- r_squared_error : number/float
    A key metric for continuous models. Nil for other model types.
- name : string
```

```

    The name of the model build.
- description : string
    A description of the model build.
- created_at : string
    The time the model build was created.

```

**created\_at** : string/date-time

The time the model was created.

**primary\_key** : string

The unique ID (primary key) of the training dataset.

**time\_zone** : string

The time zone of this model.

**current\_build\_exception** : string

Exception message, if applicable, of the current model build.

**updated\_at** : string/date-time

The time the model was updated.

**box\_cox\_transformation** : boolean

Whether to transform data so that it assumes a normal distribution. Valid only with continuous models.

**model\_type\_id** : integer

The ID of the model's type.

**last\_output\_location** : string

The output JSON for the last build.

**dependent\_variable\_order** : list

The order of dependent variables, especially useful for Ordinal Modeling.

**last\_run** : dict:

```

- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.

```

**id** : integer

The ID of the model.

**cross\_validation\_parameters** : dict

Cross validation parameter grid for tree methods, e.g. {"n\_estimators": [100, 200, 500], "learning\_rate": [0.01, 0.1], "max\_depth": [2, 3]}.

**archived** : string

The archival status of the requested object(s).

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**credential\_id** : integer

The ID of the credential used to read the target table. Defaults to the user's default credential.

**running\_as** : dict:

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

**current\_build\_state** : string

The status of the current model build. One of “succeeded”, “failed”, “queued”, or “running,” or “idle”, if no build has been attempted.

**table\_name** : string

The qualified name of the table containing the training set from which to build the model.

**model\_name** : string

The name of the model.

**put\_predictions** (*id*, *primary\_key*, *table\_name*, *\*\*kwargs*)

Add a table on which to apply the predictive model

**Parameters** **id** : integer

The ID of the model to which to apply the prediction.

**primary\_key** : list

The primary key or composite keys of the table being predicted.

**table\_name** : string

The qualified name of the table on which to apply the predictive model.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
```

```
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**limiting\_sql** : string, optional

A SQL WHERE clause used to scope the rows to be predicted.

**output\_table** : string, optional

The qualified name of the table to be created which will contain the model's predictions.

**Returns id** : integer

The ID of the model to which to apply the prediction.

**state** : string

The status of the prediction. One of: "succeeded", "failed", "queued", or "running," or "idle", if no build has been attempted.

**output\_table** : string

The qualified name of the table to be created which will contain the model's predictions.

**limiting\_sql** : string

A SQL WHERE clause used to scope the rows to be predicted.

**primary\_key** : list

The primary key or composite keys of the table being predicted.

**table\_name** : string

The qualified name of the table on which to apply the predictive model.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**put\_projects** (*id*, *project\_id*)

Add a models to a project

**Parameters id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**put\_schedules** (*id, schedule*)

Schedule the model build

**Parameters** *id* : integer

The ID of the model associated with this schedule.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**Returns** *id* : integer

The ID of the model associated with this schedule.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**put\_shares\_groups** (*id, permission\_level, group\_ids*)

Set the permissions groups has on this object

**Parameters** *id* : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**group\_ids** : list

An array of one or more group IDs

**Returns** *readers* : dict:

```
- users : list::
    - id : integer
    - name : string
- groups : list::
    - id : integer
    - name : string
```



**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_shares\_users** (*id*, *permission\_level*, *user\_ids*)

Set the permissions users have on this object

**Parameters** **id** : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**user\_ids** : list

An array of one or more user IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

## Predictions

**class Predictions** (*session*, *return\_type*='civis')

### Methods

<code>delete_runs(id, run_id)</code>	Cancel a run
<code>get(id)</code>	Show the specified prediction
<code>get_runs(id, run_id)</code>	Check status of a run
<code>list(**kwargs)</code>	List predictions
<code>list_runs(id, **kwargs)</code>	List runs for the given prediction
<code>list_schedules(id)</code>	Show the prediction schedule
<code>patch(id, **kwargs)</code>	Update a prediction
<code>post_runs(id)</code>	Start a run
<code>put_schedules(id, **kwargs)</code>	Schedule the prediction

**delete\_runs** (*id*, *run\_id*)

Cancel a run

**Parameters** *id* : integer

The ID of the prediction.

**run\_id** : integer

The ID of the run.

**Returns** None

Response code 202: success

**get** (*id*)

Show the specified prediction

**Parameters** *id* : integer

The ID of the prediction.

**Returns** *scored\_tables* : list:

```

An array of created prediction tables.
- id : integer
    The ID of the table with created predictions.
- created_at : string/date-time
    The time when the table with created predictions was created.
- score_stats : list::
    An array of metrics on the created predictions.
    - score_name : string
        The name of the score.
    - max_score : number/float
        The maximum score.
    - avg_score : number/float
        The average score.
    - min_score : number/float
        The minimum score.
    - histogram : list
        The histogram of the distribution of scores.
- name : string
    The name of table with created predictions.
- schema : string
    The schema of table with created predictions.

```

**primary\_key** : list

The primary key or composite keys of the table being predicted.

**limiting\_sql** : string

A SQL WHERE clause used to scope the rows to be predicted.

**output\_table\_name** : string

The name of the output table for this prediction.

**last\_run** : dict:

```

- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.

```

**scored\_table\_id** : integer

The ID of the source table for this prediction.

**finished\_at** : string/date-time

The end time of the last run of this prediction.

**id** : integer

The ID of the prediction.

**scored\_table\_name** : string

The name of the source table for this prediction.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**error** : string

The error, if any, of the last run of this prediction.

**started\_at** : string/date-time

The start time of the last run of this prediction.

**state** : string

The state of the last run of this prediction.

**model\_id** : integer

The ID of the model used for this prediction.

**get\_runs** (*id*, *run\_id*)

Check status of a run

**Parameters** **id** : integer

The ID of the prediction.

**run\_id** : integer

The ID of the run.

**Returns** **id** : integer

The ID of the prediction run.

**name** : string

The name of table created by this predictions run.

**prediction\_id** : integer

The ID of the prediction.

**score\_stats** : list:

```
An array of metrics on the created predictions.
- score_name : string
    The name of the score.
- max_score : number/float
    The maximum score.
- avg_score : number/float
    The average score.
- min_score : number/float
    The minimum score.
```

```
- histogram : list
    The histogram of the distribution of scores.
```

**created\_at** : string/date-time

The time when the table with created predictions was created.

**state** : string

The state of the prediction run.

**exception** : string

The exception, if any, returned by the prediction run.

**list** (\*\*kwargs)

List predictions

**Parameters** **model\_id** : integer, optional

If specified, only return predictions associated with this model ID.

**Returns** **id** : integer

The ID of the prediction.

**scored\_table\_name** : string

The name of the source table for this prediction.

**error** : string

The error, if any, of the last run of this prediction.

**output\_table\_name** : string

The name of the output table for this prediction.

**started\_at** : string/date-time

The start time of the last run of this prediction.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**state** : string

The state of the last run of this prediction.

**model\_id** : integer

The ID of the model used for this prediction.

**scored\_table\_id** : integer

The ID of the source table for this prediction.

**finished\_at** : string/date-time

The end time of the last run of this prediction.

**list\_runs** (*id*, *\*\*kwargs*)

List runs for the given prediction

**Parameters** **id** : integer

The ID of the prediction.

**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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **id** : integer

The ID of the prediction run.

**name** : string

The name of table created by this predictions run.

**prediction\_id** : integer

The ID of the prediction.

**score\_stats** : list:

```
An array of metrics on the created predictions.
- score_name : string
    The name of the score.
- max_score : number/float
    The maximum score.
- avg_score : number/float
    The average score.
- min_score : number/float
    The minimum score.
- histogram : list
    The histogram of the distribution of scores.
```

**created\_at** : string/date-time

The time when the table with created predictions was created.

**state** : string

The state of the prediction run.

**exception** : string

The exception, if any, returned by the prediction run.

**list\_schedules** (*id*)

Show the prediction schedule

**Parameters** *id* : integer

ID of the prediction associated with this schedule.

**Returns** *id* : integer

ID of the prediction associated with this schedule.

**score\_on\_model\_build** : boolean

Whether the prediction will run after a rebuild of the associated model.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**patch** (*id*, **\*\*kwargs**)

Update a prediction

**Parameters** *id* : integer

The ID of the prediction.

**output\_table\_name** : string, optional

The name of the output table for this prediction.

**primary\_key** : list, optional

The primary key or composite keys of the table being predicted.

**limiting\_sql** : string, optional

A SQL WHERE clause used to scope the rows to be predicted.

**Returns** *scored\_tables* : list:

```
An array of created prediction tables.
- id : integer
    The ID of the table with created predictions.
- created_at : string/date-time
    The time when the table with created predictions was created.
- score_stats : list::
    An array of metrics on the created predictions.
    - score_name : string
        The name of the score.
    - max_score : number/float
        The maximum score.
```

```
- avg_score : number/float
    The average score.
- min_score : number/float
    The minimum score.
- histogram : list
    The histogram of the distribution of scores.
- name : string
    The name of table with created predictions.
- schema : string
    The schema of table with created predictions.
```

**primary\_key** : list

The primary key or composite keys of the table being predicted.

**limiting\_sql** : string

A SQL WHERE clause used to scope the rows to be predicted.

**output\_table\_name** : string

The name of the output table for this prediction.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**scored\_table\_id** : integer

The ID of the source table for this prediction.

**finished\_at** : string/date-time

The end time of the last run of this prediction.

**id** : integer

The ID of the prediction.

**scored\_table\_name** : string

The name of the source table for this prediction.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
```



```
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
    ↪hour
```

**error** : string

The error, if any, of the last run of this prediction.

**started\_at** : string/date-time

The start time of the last run of this prediction.

**state** : string

The state of the last run of this prediction.

**model\_id** : integer

The ID of the model used for this prediction.

**post\_runs** (*id*)

Start a run

**Parameters** **id** : integer

The ID of the prediction.

**Returns** **id** : integer

The ID of the prediction run.

**name** : string

The name of table created by this predictions run.

**prediction\_id** : integer

The ID of the prediction.

**score\_stats** : list:

```
An array of metrics on the created predictions.
- score_name : string
    The name of the score.
- max_score : number/float
    The maximum score.
- avg_score : number/float
    The average score.
- min_score : number/float
    The minimum score.
- histogram : list
    The histogram of the distribution of scores.
```

**created\_at** : string/date-time

The time when the table with created predictions was created.

**state** : string

The state of the prediction run.

**exception** : string

The exception, if any, returned by the prediction run.

**put\_schedules** (*id*, *\*\*kwargs*)

Schedule the prediction

**Parameters** *id* : integer

ID of the prediction associated with this schedule.

**score\_on\_model\_build** : boolean, optional

Whether the prediction will run after a rebuild of the associated model.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**Returns** *id* : integer

ID of the prediction associated with this schedule.

**score\_on\_model\_build** : boolean

Whether the prediction will run after a rebuild of the associated model.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

## Projects

**class Projects** (*session*, *return\_type*='civis')

### Methods

<a href="#"><code>delete_shares_groups</code></a> ( <i>id</i> , <i>group_id</i> )	Revoke the permissions a group has on this object
<a href="#"><code>delete_shares_users</code></a> ( <i>id</i> , <i>user_id</i> )	Revoke the permissions a user has on this object
<a href="#"><code>get</code></a> ( <i>project_id</i> )	Get a detailed view of a project and the objects in it
<a href="#"><code>list</code></a> ( <i>**kwargs</i> )	List projects
Continued on next page	

Table 4.11 – continued from previous page

<code>list_shares(id)</code>	List users and groups permissioned on this object
<code>post(description, name, **kwargs)</code>	Create a project
<code>put(project_id, **kwargs)</code>	Update a project
<code>put_archive(id, status)</code>	Update the archive status of this object
<code>put_shares_groups(id, permission_level, ...)</code>	Set the permissions groups has on this object
<code>put_shares_users(id, permission_level, user_ids)</code>	Set the permissions users have on this object

**delete\_shares\_groups** (*id, group\_id*)

Revoke the permissions a group has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**group\_id** : integer

ID of the group

**Returns** None

Response code 204: success

**delete\_shares\_users** (*id, user\_id*)

Revoke the permissions a user has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**user\_id** : integer

ID of the user

**Returns** None

Response code 204: success

**get** (*project\_id*)

Get a detailed view of a project and the objects in it

**Parameters** **project\_id** : integer

**Returns** **description** : string

A description of the project

**surveys** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- updated_at : string/time
```

**tables** : list:

```
- updated_at : string/time
- name : string
- schema : string
- column_count : integer
- created_at : string/time
- row_count : integer
```

**updated\_at** : string/time

**app\_instances** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- name : string
- slug : string
- updated_at : string/time
```

**script\_templates** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- name : string
- updated_at : string/time
```

**author** : dict:

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

**all\_objects** : list:

```
- fco_type : string
- object_id : integer
- icon : string
- archived : string
    The archival status of the requested object(s).
- author : string
- project_id : integer
- name : string
- object_type : string
- sub_type : string
```

**id** : integer

The ID for this project.

**models** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- state : string
- name : string
- updated_at : string/time
```

**scripts** : list:

```
- id : integer
    The object ID.
- updated_at : string/time
- type : string
- name : string
- created_at : string/time
- state : string
- finished_at : string/time
```

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**files** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- file_size : integer
- file_name : string
- updated_at : string/time
```

**reports** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- state : string
- name : string
- updated_at : string/time
```

**imports** : list:

```
- id : integer
    The object ID.
- updated_at : string/time
- type : string
- name : string
- created_at : string/time
- state : string
- finished_at : string/time
```

**created\_at** : string/time

**note** : string

**users** : list:

```
Users who can see the project
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
    This user's initials.
- name : string
```

```
This user's name.  
- username : string  
  This user's username.
```

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**auto\_share** : boolean

**list** (*\*\*kwargs*)

List projects

**Parameters** **author** : string, optional

If specified, return projects owned by this author. It accepts a comma- separated list of author ids.

**permission** : string, optional

A permissions string, one of “read”, “write”, or “manage”. Lists only projects for which the current user has that permission.

**archived** : string, optional

The archival status of the requested object(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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **id** : integer

The ID for this project.

**description** : string

A description of the project

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**created\_at** : string/time**users** : list:

```

Users who can see the project
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.

```

**author** : dict:

```

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

```

**updated\_at** : string/time**auto\_share** : boolean**list\_shares** (*id*)

List users and groups permissioned on this object

**Parameters** **id** : integer

The ID of the object.

**Returns** **readers** : dict:

```

- users : list::
    - id : integer
    - name : string
- groups : list::
    - id : integer
    - name : string

```

**owners** : dict:

```

- users : list::
    - id : integer
    - name : string
- groups : list::
    - id : integer
    - name : string

```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**post** (*description*, *name*, *\*\*kwargs*)

Create a project

**Parameters** **description** : string

A description of the project

**name** : string

The name of this project.

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints. The object can still be queried directly by ID

**note** : string, optional

Notes for the project

**Returns** **description** : string

A description of the project

**surveys** : list:

```
- id : integer
  The object ID.
- created_at : string/time
- updated_at : string/time
```

**tables** : list:

```
- updated_at : string/time
- name : string
- schema : string
- column_count : integer
- created_at : string/time
- row_count : integer
```

**updated\_at** : string/time

**app\_instances** : list:

```
- id : integer
  The object ID.
- created_at : string/time
```



```
- name : string
- slug : string
- updated_at : string/time
```

**script\_templates** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- name : string
- updated_at : string/time
```

**author** : dict:

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

**all\_objects** : list:

```
- fco_type : string
- object_id : integer
- icon : string
- archived : string
    The archival status of the requested object(s).
- author : string
- project_id : integer
- name : string
- object_type : string
- sub_type : string
```

**id** : integer

The ID for this project.

**models** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- state : string
- name : string
- updated_at : string/time
```

**scripts** : list:

```
- id : integer
    The object ID.
- updated_at : string/time
- type : string
- name : string
```

```
- created_at : string/time
- state : string
- finished_at : string/time
```

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**files** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- file_size : integer
- file_name : string
- updated_at : string/time
```

**reports** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- state : string
- name : string
- updated_at : string/time
```

**imports** : list:

```
- id : integer
    The object ID.
- updated_at : string/time
- type : string
- name : string
- created_at : string/time
- state : string
- finished_at : string/time
```

**created\_at** : string/time

**note** : string

**users** : list:

```
Users who can see the project
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.
```

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**auto\_share** : boolean

**put** (*project\_id*, *\*\*kwargs*)  
Update a project

**Parameters** **project\_id** : integer

**description** : string, optional

A description of the project

**name** : string, optional

The name of this project.

**note** : string, optional

Notes for the project

**Returns** **description** : string

A description of the project

**surveys** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- updated_at : string/time
```

**tables** : list:

```
- updated_at : string/time
- name : string
- schema : string
- column_count : integer
- created_at : string/time
- row_count : integer
```

**updated\_at** : string/time

**app\_instances** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- name : string
- slug : string
- updated_at : string/time
```

**script\_templates** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- name : string
- updated_at : string/time
```

**author** : dict:

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

**all\_objects** : list:

```
- fco_type : string
- object_id : integer
- icon : string
- archived : string
    The archival status of the requested object(s).
- author : string
- project_id : integer
- name : string
- object_type : string
- sub_type : string
```

**id** : integer

The ID for this project.

**models** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- state : string
- name : string
- updated_at : string/time
```

**scripts** : list:

```
- id : integer
    The object ID.
- updated_at : string/time
- type : string
- name : string
- created_at : string/time
- state : string
- finished_at : string/time
```

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**files** : list:

```
- id : integer
    The object ID.
```

```
- created_at : string/time
- file_size : integer
- file_name : string
- updated_at : string/time
```

**reports** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- state : string
- name : string
- updated_at : string/time
```

**imports** : list:

```
- id : integer
    The object ID.
- updated_at : string/time
- type : string
- name : string
- created_at : string/time
- state : string
- finished_at : string/time
```

**created\_at** : string/time

**note** : string

**users** : list:

```
Users who can see the project
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.
```

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**auto\_share** : boolean

**put\_archive** (*id*, *status*)

Update the archive status of this object

**Parameters** **id** : integer

The ID of the object.

**status** : boolean

The desired archived status of the object.

**Returns** **description** : string

A description of the project

**surveys** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- updated_at : string/time
```

**tables** : list:

```
- updated_at : string/time
- name : string
- schema : string
- column_count : integer
- created_at : string/time
- row_count : integer
```

**updated\_at** : string/time

**app\_instances** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- name : string
- slug : string
- updated_at : string/time
```

**script\_templates** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- name : string
- updated_at : string/time
```

**author** : dict:

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

**all\_objects** : list:

```
- fco_type : string
- object_id : integer
- icon : string
- archived : string
    The archival status of the requested object(s).
```

```
- author : string
- project_id : integer
- name : string
- object_type : string
- sub_type : string
```

**id** : integer

The ID for this project.

**models** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- state : string
- name : string
- updated_at : string/time
```

**scripts** : list:

```
- id : integer
    The object ID.
- updated_at : string/time
- type : string
- name : string
- created_at : string/time
- state : string
- finished_at : string/time
```

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**files** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- file_size : integer
- file_name : string
- updated_at : string/time
```

**reports** : list:

```
- id : integer
    The object ID.
- created_at : string/time
- state : string
- name : string
- updated_at : string/time
```

**imports** : list:

```
- id : integer
    The object ID.
```

```
- updated_at : string/time
- type : string
- name : string
- created_at : string/time
- state : string
- finished_at : string/time
```

**created\_at** : string/time

**note** : string

**users** : list:

```
Users who can see the project
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.
```

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**auto\_share** : boolean

**put\_shares\_groups** (*id*, *permission\_level*, *group\_ids*)

Set the permissions groups has on this object

**Parameters** **id** : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**group\_ids** : list

An array of one or more group IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
```



```
- id : integer
- name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_shares\_users** (*id*, *permission\_level*, *user\_ids*)

Set the permissions users have on this object

**Parameters** *id* : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**user\_ids** : list

An array of one or more user IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

## Queries

**class** **Queries** (*session*, *return\_type*='civis')

### Methods

<code>delete_runs(id, run_id)</code>	Cancel a run
<code>get(id)</code>	Get details about a query
<code>get_runs(id, run_id)</code>	Check status of a run
<code>list(**kwargs)</code>	List all queries
<code>list_runs(id, **kwargs)</code>	List runs for the given query
<code>post(preview_rows, sql, database, **kwargs)</code>	Execute a query
<code>post_runs(id)</code>	Start a run
<code>put_scripts(id, script_id)</code>	Update the query's associated script

**delete\_runs** (*id*, *run\_id*)

Cancel a run

**Parameters** **id** : integer

The ID of the query.

**run\_id** : integer

The ID of the run.

**Returns** None

Response code 202: success

**get** (*id*)

Get details about a query

**Parameters** **id** : integer

The query ID.

**Returns** **updated\_at** : string/time

**state** : string

The state of the last run.

**result\_rows** : list

A preview of rows returned by the query.

**last\_run\_id** : integer

The ID of the last run.

**script\_id** : integer

The ID of the script associated with this query.

**author** : dict:

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

**exception** : string

Exception returned from the query, null if the query was a success.

**result\_columns** : list

A preview of columns returned by the query.

**id** : integer

The query ID.

**sql** : string

The SQL to execute.

**name** : string

The name of the query.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**started\_at** : string/date-time

The start time of the last run.

**created\_at** : string/time

**report\_id** : integer

The ID of the report associated with this query.

**credential** : integer

The credential ID.

**finished\_at** : string/date-time

The end time of the last run.

**database** : integer

The database ID.

**get\_runs** (*id*, *run\_id*)

Check status of a run

**Parameters** *id* : integer

The ID of the query.

*run\_id* : integer

The ID of the run.

**Returns** *id* : integer

The ID of the run.

*query\_id* : integer

The ID of the query.

*is\_cancel\_requested* : boolean

True if run cancel requested, else false.

*error* : string

The error, if any, returned by the run.

*started\_at* : string/time

The time the last run started at.

*state* : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

*finished\_at* : string/time

The time the last run completed.

**list** (*\*\*kwargs*)

List all queries

**Parameters** *database\_id* : integer, optional

The database ID.

*author\_id* : integer, optional

The author of the query.

*created\_before* : string, optional

An upper bound for the creation date of the query.

*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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **updated\_at** : string/time

**state** : string

The state of the last run.

**result\_rows** : list

A preview of rows returned by the query.

**last\_run\_id** : integer

The ID of the last run.

**script\_id** : integer

The ID of the script associated with this query.

**exception** : string

Exception returned from the query, null if the query was a success.

**result\_columns** : list

A preview of columns returned by the query.

**id** : integer

The query ID.

**sql** : string

The SQL to execute.

**preview\_rows** : integer

The number of rows to save from the query's result (maximum: 100).

**started\_at** : string/date-time

The start time of the last run.

**created\_at** : string/time

**report\_id** : integer

The ID of the report associated with this query.

**credential** : integer

The credential ID.

**finished\_at** : string/date-time

The end time of the last run.

**database** : integer

The database ID.

**list\_runs** (*id*, *\*\*kwargs*)

List runs for the given query

**Parameters** *id* : integer

The ID of the query.

**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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** *id* : integer

The ID of the run.

**query\_id** : integer

The ID of the query.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**post** (*preview\_rows*, *sql*, *database*, *\*\*kwargs*)

Execute a query

**Parameters** *preview\_rows* : integer

The number of rows to save from the query's result (maximum: 100).

**sql** : string

The SQL to execute.

**database** : integer

The database ID.

**column\_delimiter** : string, optional

The delimiter to use. One of comma or tab, or pipe [default: comma].

**include\_header** : boolean, optional

Whether the CSV output should include a header row [default: true].

**filename\_prefix** : string, optional

The output filename prefix.

**unquoted** : boolean, optional

If true, will not quote fields.

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**compression** : string, optional

The type of compression. One of gzip or zip, or none [default: gzip].

**interactive** : boolean, optional

Deprecated and not used.

**credential** : integer, optional

The credential ID.

**Returns include\_header** : boolean

Whether the CSV output should include a header row [default: true].

**sql** : string

The SQL to execute.

**exception** : string

Exception returned from the query, null if the query was a success.

**result\_columns** : list

A preview of columns returned by the query.

**created\_at** : string/time

**state** : string

The state of the last run.

**compression** : string

The type of compression. One of gzip or zip, or none [default: gzip].

**last\_run\_id** : integer

The ID of the last run.

**database** : integer

The database ID.

**updated\_at** : string/time

**credential** : integer

The credential ID.

**result\_rows** : list

A preview of rows returned by the query.

**unquoted** : boolean

If true, will not quote fields.

**script\_id** : integer

The ID of the script associated with this query.

**interactive** : boolean

Deprecated and not used.

**id** : integer

The query ID.

**column\_delimiter** : string

The delimiter to use. One of comma or tab, or pipe [default: comma].

**filename\_prefix** : string

The output filename prefix.

**preview\_rows** : integer

The number of rows to save from the query's result (maximum: 100).

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**started\_at** : string/date-time

The start time of the last run.

**report\_id** : integer

The ID of the report associated with this query.

**finished\_at** : string/date-time

The end time of the last run.

**post\_runs** (*id*)

Start a run

**Parameters id** : integer

The ID of the query.

**Returns id** : integer

The ID of the run.

**query\_id** : integer

The ID of the query.

**is\_cancel\_requested** : boolean



True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**put\_scripts** (*id*, *script\_id*)

Update the query's associated script

**Parameters** **id** : integer

The query ID.

**script\_id** : integer

The ID of the script associated with this query.

**Returns** **updated\_at** : string/time

**state** : string

The state of the last run.

**result\_rows** : list

A preview of rows returned by the query.

**last\_run\_id** : integer

The ID of the last run.

**script\_id** : integer

The ID of the script associated with this query.

**author** : dict:

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

**exception** : string

Exception returned from the query, null if the query was a success.

**result\_columns** : list

A preview of columns returned by the query.

**id** : integer

The query ID.

**sql** : string

The SQL to execute.

**name** : string

The name of the query.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**started\_at** : string/date-time

The start time of the last run.

**created\_at** : string/time

**report\_id** : integer

The ID of the report associated with this query.

**credential** : integer

The credential ID.

**finished\_at** : string/date-time

The end time of the last run.

**database** : integer

The database ID.

## Reports

**class Reports** (*session*, *return\_type*='civis')

### Methods

<code>delete_grants(id)</code>	Revoke permission for this report to perform Civis platform API operations on
<code>delete_projects(id, project_id)</code>	Remove a Report from a project
<code>delete_shares_groups(id, group_id)</code>	Revoke the permissions a group has on this object
<code>delete_shares_users(id, user_id)</code>	Revoke the permissions a user has on this object
<code>get(id)</code>	Show a single report
<code>list(**kwargs)</code>	List the reports visible to the current user
<code>list_projects(id)</code>	List the projects a Report belongs to
<code>list_shares(id)</code>	List users and groups permissioned on this object
<code>list_snapshots(id)</code>	Get details about the report's snapshot automation settings
<code>patch(id, **kwargs)</code>	Update a report
<code>patch_snapshots(id, **kwargs)</code>	Update the report's snapshot automation settings
<code>post(**kwargs)</code>	Create a report

Continued on next page

Table 4.13 – continued from previous page

<i>post_grants</i> (id)	Grant this report the ability to perform Civis platform API operations on your
<i>post_snapshots</i> (id, <b>**kwargs</b> )	Generate and optionally email a snapshot of the specified report
<i>put_archive</i> (id, status)	Update the archive status of this object
<i>put_projects</i> (id, project_id)	Add a Report to a project
<i>put_shares_groups</i> (id, permission_level, ...)	Set the permissions groups has on this object
<i>put_shares_users</i> (id, permission_level, user_ids)	Set the permissions users have on this object

**delete\_grants** (*id*)

Revoke permission for this report to perform Civis platform API operations on your behalf

**Parameters** **id** : integer

The ID of this report.

**Returns** None

Response code 204: success

**delete\_projects** (*id*, *project\_id*)

Remove a Report from a project

**Parameters** **id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**delete\_shares\_groups** (*id*, *group\_id*)

Revoke the permissions a group has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**group\_id** : integer

ID of the group

**Returns** None

Response code 204: success

**delete\_shares\_users** (*id*, *user\_id*)

Revoke the permissions a user has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**user\_id** : integer

ID of the user

**Returns** None

Response code 204: success

**get** (*id*)

Show a single report

**Parameters** *id* : integer

The ID of this report.

**Returns** *projects* : list:

```
A list of projects containing the report.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**provide\_api\_key** : boolean

Whether the report requests an API Key from the report viewer.

**template\_id** : integer

The ID of the template used for this report.

**viz\_updated\_at** : string/time

The time that the report's visualization was last updated.

**valid\_output\_file** : boolean

Whether the job (a script or a query) that backs the report currently has a valid output file.

**user** : dict:

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

**config** : string

Any configuration metadata for this report.

**app\_state** : dict

Any application state blob for this report.

**api\_key\_id** : integer

The ID of the API key. Can be used for auditing API use by this report.

**created\_at** : string/time

**state** : string

The status of the report's last run.

**auth\_thumbnail\_url** : string

URL for a thumbnail of the report.

**job\_path** : string

The link to details of the job that backs this report.

**updated\_at** : string/time

**api\_key** : string

A Civis API key that can be used by this report.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**script** : dict:

```
- id : integer
    The ID for the script.
- sql : string
    The raw SQL query for the script.
- name : string
    The name of the script.
```

**tableau\_id** : integer

**id** : integer

The ID of this report.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the report.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**auth\_code\_url** : string

**auth\_data\_url** : string

**finished\_at** : string/time

The time that the report's last run finished.

**list** (*\*\*kwargs*)

List the reports visible to the current user

**Parameters type** : string, optional

If specified, return report of these types. It accepts a comma-separated list, possible values are 'tableau', 'other'.

**author** : string, optional

If specified, return reports from this author. It accepts a comma-separated list of author ids.

**template\_id** : integer, optional

If specified, return reports using the provided Template.

**archived** : string, optional

The archival status of the requested object(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`.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **job\_path** : string

The link to details of the job that backs this report.

**updated\_at** : string/time

**template\_id** : integer

The ID of the template used for this report.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**projects** : list:

```
A list of projects containing the report.
- id : integer
    The ID for the project.
```

```
- name : string
    The name of the project.
```

**viz\_updated\_at** : string/time

The time that the report's visualization was last updated.

**user** : dict:

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

**script** : dict:

```
- id : integer
    The ID for the script.
- sql : string
    The raw SQL query for the script.
- name : string
    The name of the script.
```

**tableau\_id** : integer

**id** : integer

The ID of this report.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the report.

**created\_at** : string/time

**state** : string

The status of the report's last run.

**auth\_thumbnail\_url** : string

URL for a thumbnail of the report.

**finished\_at** : string/time

The time that the report's last run finished.

**list\_projects** (*id*)

List the projects a Report belongs to

**Parameters id** : integer

The ID of the resource.

**Returns id** : integer

The ID for this project.

**description** : string

A description of the project

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**created\_at** : string/time

**users** : list:

```
Users who can see the project
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.
```

**author** : dict:

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

**updated\_at** : string/time

**auto\_share** : boolean

**list\_shares** (*id*)

List users and groups permissioned on this object

**Parameters** *id* : integer

The ID of the object.

**Returns** **readers** : dict:

```
- users : list::
    - id : integer
    - name : string
- groups : list::
    - id : integer
    - name : string
```



**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**list\_snapshots** (*id*)

Get details about the report's snapshot automation settings

**Parameters** **id** : integer

The ID of this report.

**Returns** **id** : integer

The ID of this report.

**width** : integer

The width of the cropped snapshot image in screen pixels. The default value is 1440 pixels. Minimum value is 600 pixels.

**height** : integer

The height of the cropped snapshot image in screen pixels. The default value is 900 pixels. Minimum value is 600 pixels.

**email\_template** : string

Custom email template.

**schedule** : dict:

```
- scheduled : boolean
  If the object is scheduled
- scheduled_days : list
  Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
  Minutes of the day it is scheduled on
- scheduled_hours : list
  Hours of the day it is scheduled on
```

```
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**send\_email\_on\_completion** : boolean

Whether the job will send emails on completion.

**email\_subject** : string

Subject for Email.

**state** : string

The status of the job's last run.

**parent\_id** : integer

The ID of the parent job that will trigger this snapshot.

**finished\_at** : string/time

The time that the job's last run finished.

**recipient\_email\_addresses** : string

Email addresses to send report to, comma separated.

**patch** (*id*, *\*\*kwargs*)

Update a report

**Parameters** **id** : integer

The ID of the report to modify.

**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.

**name** : string, optional

The name of the report.

**app\_state** : dict, optional

The application state blob for this 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

**Returns** **projects** : list:

```
A list of projects containing the report.
- id : integer
    The ID for the project.
```

```
- name : string
    The name of the project.
```

**provide\_api\_key** : boolean

Whether the report requests an API Key from the report viewer.

**template\_id** : integer

The ID of the template used for this report.

**viz\_updated\_at** : string/time

The time that the report's visualization was last updated.

**valid\_output\_file** : boolean

Whether the job (a script or a query) that backs the report currently has a valid output file.

**user** : dict:

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

**config** : string

Any configuration metadata for this report.

**app\_state** : dict

Any application state blob for this report.

**api\_key\_id** : integer

The ID of the API key. Can be used for auditing API use by this report.

**created\_at** : string/time

**state** : string

The status of the report's last run.

**auth\_thumbnail\_url** : string

URL for a thumbnail of the report.

**job\_path** : string

The link to details of the job that backs this report.

**updated\_at** : string/time

**api\_key** : string

A Civis API key that can be used by this report.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**script** : dict:

```
- id : integer
    The ID for the script.
- sql : string
    The raw SQL query for the script.
- name : string
    The name of the script.
```

**tableau\_id** : integer

**id** : integer

The ID of this report.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the report.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**auth\_code\_url** : string

**auth\_data\_url** : string

**finished\_at** : string/time

The time that the report's last run finished.

**patch\_snapshots** (*id*, *\*\*kwargs*)

Update the report's snapshot automation settings

**Parameters** **id** : integer

The ID of this report.

**email\_subject** : string, optional

Subject for Email.

**width** : integer, optional

The width of the cropped snapshot image in screen pixels. The default value is 1440 pixels. Minimum value is 600 pixels.

**height** : integer, optional

The height of the cropped snapshot image in screen pixels. The default value is 900 pixels. Minimum value is 600 pixels.

**email\_template** : string, optional

Custom email template.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    hour
```

**send\_email\_on\_completion** : boolean, optional

Whether the job will send emails on completion.

**state** : string, optional

The status of the job's last run.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this snapshot.

**recipient\_email\_addresses** : string, optional

Email addresses to send report to, comma separated.

**finished\_at** : string/time, optional

The time that the job's last run finished.

**Returns id** : integer

The ID of this report.

**width** : integer

The width of the cropped snapshot image in screen pixels. The default value is 1440 pixels. Minimum value is 600 pixels.

**height** : integer

The height of the cropped snapshot image in screen pixels. The default value is 900 pixels. Minimum value is 600 pixels.

**email\_template** : string

Custom email template.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
```

```
Minutes of the day it is scheduled on
- scheduled_hours : list
  Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
  Alternative to scheduled minutes, number of times to run per_
↪hour
```

**send\_email\_on\_completion** : boolean

Whether the job will send emails on completion.

**email\_subject** : string

Subject for Email.

**state** : string

The status of the job's last run.

**parent\_id** : integer

The ID of the parent job that will trigger this snapshot.

**finished\_at** : string/time

The time that the job's last run finished.

**recipient\_email\_addresses** : string

Email addresses to send report to, comma separated.

**post** (*\*\*kwargs*)

Create a report

**Parameters** **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.

**name** : string, optional

The name of the report.

**app\_state** : dict, optional

Any application state blob for this report.

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**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.

**Returns** **projects** : list:

```
A list of projects containing the report.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**provide\_api\_key** : boolean

Whether the report requests an API Key from the report viewer.

**template\_id** : integer

The ID of the template used for this report.

**viz\_updated\_at** : string/time

The time that the report's visualization was last updated.

**valid\_output\_file** : boolean

Whether the job (a script or a query) that backs the report currently has a valid output file.

**user** : dict:

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

**config** : string

Any configuration metadata for this report.

**app\_state** : dict

Any application state blob for this report.

**api\_key\_id** : integer

The ID of the API key. Can be used for auditing API use by this report.

**created\_at** : string/time

**state** : string

The status of the report's last run.

**auth\_thumbnail\_url** : string

URL for a thumbnail of the report.

**job\_path** : string

The link to details of the job that backs this report.

**updated\_at** : string/time

**api\_key** : string

A Civis API key that can be used by this report.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**script** : dict:

```
- id : integer
    The ID for the script.
- sql : string
    The raw SQL query for the script.
- name : string
    The name of the script.
```

**tableau\_id** : integer

**id** : integer

The ID of this report.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the report.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**auth\_code\_url** : string

**auth\_data\_url** : string

**finished\_at** : string/time

The time that the report's last run finished.

**post\_grants** (*id*)

Grant this report the ability to perform Civis platform API operations on your behalf

**Parameters id** : integer

The ID of this report.

**Returns projects** : list:

```
A list of projects containing the report.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```



**provide\_api\_key** : boolean

Whether the report requests an API Key from the report viewer.

**template\_id** : integer

The ID of the template used for this report.

**viz\_updated\_at** : string/time

The time that the report's visualization was last updated.

**valid\_output\_file** : boolean

Whether the job (a script or a query) that backs the report currently has a valid output file.

**user** : dict:

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

**config** : string

Any configuration metadata for this report.

**app\_state** : dict

Any application state blob for this report.

**api\_key\_id** : integer

The ID of the API key. Can be used for auditing API use by this report.

**created\_at** : string/time

**state** : string

The status of the report's last run.

**auth\_thumbnail\_url** : string

URL for a thumbnail of the report.

**job\_path** : string

The link to details of the job that backs this report.

**updated\_at** : string/time

**api\_key** : string

A Civis API key that can be used by this report.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**script** : dict:

```
- id : integer
    The ID for the script.
- sql : string
    The raw SQL query for the script.
- name : string
    The name of the script.
```

**tableau\_id** : integer

**id** : integer

The ID of this report.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the report.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**auth\_code\_url** : string

**auth\_data\_url** : string

**finished\_at** : string/time

The time that the report's last run finished.

**post\_snapshots** (*id*, *\*\*kwargs*)

Generate and optionally email a snapshot of the specified report

**Parameters** **id** : integer

The ID of this report.

**email\_subject** : string, optional

Subject for Email.

**width** : integer, optional

The width of the cropped snapshot image in screen pixels. The default value is 1440 pixels. Minimum value is 600 pixels.

**height** : integer, optional

The height of the cropped snapshot image in screen pixels. The default value is 900 pixels. Minimum value is 600 pixels.

**email\_template** : string, optional

Custom email template.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    hour
```

**send\_email\_on\_completion** : boolean, optional

Whether the job will send emails on completion.

**state** : string, optional

The status of the job's last run.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this snapshot.

**recipient\_email\_addresses** : string, optional

Email addresses to send report to, comma separated.

**finished\_at** : string/time, optional

The time that the job's last run finished.

**Returns id** : integer

The ID of this report.

**width** : integer

The width of the cropped snapshot image in screen pixels. The default value is 1440 pixels. Minimum value is 600 pixels.

**height** : integer

The height of the cropped snapshot image in screen pixels. The default value is 900 pixels. Minimum value is 600 pixels.

**email\_template** : string

Custom email template.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
```

```
Minutes of the day it is scheduled on
- scheduled_hours : list
  Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
  Alternative to scheduled minutes, number of times to run per_
↪hour
```

**send\_email\_on\_completion** : boolean

Whether the job will send emails on completion.

**email\_subject** : string

Subject for Email.

**state** : string

The status of the job's last run.

**parent\_id** : integer

The ID of the parent job that will trigger this snapshot.

**finished\_at** : string/time

The time that the job's last run finished.

**recipient\_email\_addresses** : string

Email addresses to send report to, comma separated.

**put\_archive** (*id*, *status*)

Update the archive status of this object

**Parameters id** : integer

The ID of the object.

**status** : boolean

The desired archived status of the object.

**Returns projects** : list:

```
A list of projects containing the report.
- id : integer
  The ID for the project.
- name : string
  The name of the project.
```

**provide\_api\_key** : boolean

Whether the report requests an API Key from the report viewer.

**template\_id** : integer

The ID of the template used for this report.

**viz\_updated\_at** : string/time

The time that the report's visualization was last updated.

**valid\_output\_file** : boolean

Whether the job (a script or a query) that backs the report currently has a valid output file.

**user** : dict:

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

**config** : string

Any configuration metadata for this report.

**app\_state** : dict

Any application state blob for this report.

**api\_key\_id** : integer

The ID of the API key. Can be used for auditing API use by this report.

**created\_at** : string/time

**state** : string

The status of the report's last run.

**auth\_thumbnail\_url** : string

URL for a thumbnail of the report.

**job\_path** : string

The link to details of the job that backs this report.

**updated\_at** : string/time

**api\_key** : string

A Civis API key that can be used by this report.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**script** : dict:

```
- id : integer
    The ID for the script.
- sql : string
    The raw SQL query for the script.
```

```
- name : string
    The name of the script.
```

**tableau\_id** : integer

**id** : integer

The ID of this report.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the report.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**auth\_code\_url** : string

**auth\_data\_url** : string

**finished\_at** : string/time

The time that the report's last run finished.

**put\_projects** (*id*, *project\_id*)

Add a Report to a project

**Parameters id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**put\_shares\_groups** (*id*, *permission\_level*, *group\_ids*)

Set the permissions groups has on this object

**Parameters id** : integer

ID of the resource to be shared

**permission\_level** : string

Options are: "read", "write", or "manage"

**group\_ids** : list

An array of one or more group IDs

**Returns readers** : dict:

```
- users : list::
    - id : integer
    - name : string
- groups : list::
```

```
- id : integer
- name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_shares\_users** (*id*, *permission\_level*, *user\_ids*)

Set the permissions users have on this object

**Parameters** **id** : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**user\_ids** : list

An array of one or more user IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
```

```
- id : integer
- name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

## Scripts

**class Scripts** (*session*, *return\_type*='civis')

### Methods

<code>delete_containers_projects(id, project_id)</code>	Remove a container docker from a project
<code>delete_containers_runs(id, run_id)</code>	Cancel a run
<code>delete_containers_shares_groups(id, group_id)</code>	Revoke the permissions a group has on this object
<code>delete_containers_shares_users(id, user_id)</code>	Revoke the permissions a user has on this object
<code>delete_custom_projects(id, project_id)</code>	Remove a Job from a project
<code>delete_custom_runs(id, run_id)</code>	Cancel a run
<code>delete_custom_shares_groups(id, group_id)</code>	Revoke the permissions a group has on this object
<code>delete_custom_shares_users(id, user_id)</code>	Revoke the permissions a user has on this object
<code>delete_javascript_projects(id, project_id)</code>	Remove a scripted sql from a project
<code>delete_javascript_runs(id, run_id)</code>	Cancel a run
<code>delete_javascript_shares_groups(id, group_id)</code>	Revoke the permissions a group has on this object
<code>delete_javascript_shares_users(id, user_id)</code>	Revoke the permissions a user has on this object
<code>delete_python3_projects(id, project_id)</code>	Remove a python docker from a project
<code>delete_python3_runs(id, run_id)</code>	Cancel a run
<code>delete_python3_shares_groups(id, group_id)</code>	Revoke the permissions a group has on this object
<code>delete_python3_shares_users(id, user_id)</code>	Revoke the permissions a user has on this object
<code>delete_r_projects(id, project_id)</code>	Remove a r docker from a project
<code>delete_r_runs(id, run_id)</code>	Cancel a run
<code>delete_r_shares_groups(id, group_id)</code>	Revoke the permissions a group has on this object

Continued on next page



Table 4.14 – continued from previous page

<i>delete_r_shares_users</i> (id, user_id)	Revoke the permissions a user has on this object
<i>delete_sql_projects</i> (id, project_id)	Remove a scripts from a project
<i>delete_sql_runs</i> (id, run_id)	Cancel a run
<i>delete_sql_shares_groups</i> (id, group_id)	Revoke the permissions a group has on this object
<i>delete_sql_shares_users</i> (id, user_id)	Revoke the permissions a user has on this object
<i>get</i> (id)	Get details about a script
<i>get_containers</i> (id)	View a container
<i>get_containers_runs</i> (id, run_id)	Check status of a run
<i>get_custom</i> (id)	Get a CustomScript
<i>get_custom_runs</i> (id, run_id)	Check status of a run
<i>get_javascript</i> (id)	Get a JavaScript Script
<i>get_javascript_runs</i> (id, run_id)	Check status of a run
<i>get_python3</i> (id)	Get a Python Script
<i>get_python3_runs</i> (id, run_id)	Check status of a run
<i>get_r</i> (id)	Get an R Script
<i>get_r_runs</i> (id, run_id)	Check status of a run
<i>get_sql</i> (id)	Get a SQL script
<i>get_sql_runs</i> (id, run_id)	Check status of a run
<i>list</i> (**kwargs)	List scripts
<i>list_containers_projects</i> (id)	List the projects a container docker belongs to
<i>list_containers_runs</i> (id, **kwargs)	List runs for the given container
<i>list_containers_runs_logs</i> (id, run_id, **kwargs)	Get the logs for a run
<i>list_containers_runs_outputs</i> (id, run_id, ...)	List the outputs for a run
<i>list_containers_shares</i> (id)	List users and groups permissioned on this object
<i>list_custom</i> (**kwargs)	List Custom Scripts
<i>list_custom_projects</i> (id)	List the projects a Job belongs to
<i>list_custom_runs</i> (id, **kwargs)	List runs for the given custom
<i>list_custom_runs_logs</i> (id, run_id, **kwargs)	Get the logs for a run
<i>list_custom_runs_outputs</i> (id, run_id, **kwargs)	List the outputs for a run
<i>list_custom_shares</i> (id)	List users and groups permissioned on this object
<i>list_history</i> (id)	Get the run history and outputs of this script
<i>list_javascript_projects</i> (id)	List the projects a scripted sql belongs to
<i>list_javascript_runs</i> (id, **kwargs)	List runs for the given javascript
<i>list_javascript_runs_logs</i> (id, run_id, **kwargs)	Get the logs for a run
<i>list_javascript_runs_outputs</i> (id, run_id, ...)	List the outputs for a run
<i>list_javascript_shares</i> (id)	List users and groups permissioned on this object
<i>list_python3_projects</i> (id)	List the projects a python docker belongs to
<i>list_python3_runs</i> (id, **kwargs)	List runs for the given python
<i>list_python3_runs_logs</i> (id, run_id, **kwargs)	Get the logs for a run
<i>list_python3_runs_outputs</i> (id, run_id, **kwargs)	List the outputs for a run
<i>list_python3_shares</i> (id)	List users and groups permissioned on this object
<i>list_r_projects</i> (id)	List the projects a r docker belongs to
<i>list_r_runs</i> (id, **kwargs)	List runs for the given r
<i>list_r_runs_logs</i> (id, run_id, **kwargs)	Get the logs for a run

Continued on next page

Table 4.14 – continued from previous page

<i>list_r_runs_outputs</i> (id, run_id, **kwargs)	List the outputs for a run
<i>list_r_shares</i> (id)	List users and groups permissioned on this object
<i>list_sql_projects</i> (id)	List the projects a script belongs to
<i>list_sql_runs</i> (id, **kwargs)	List runs for the given sql
<i>list_sql_runs_logs</i> (id, run_id, **kwargs)	Get the logs for a run
<i>list_sql_runs_outputs</i> (id, run_id, **kwargs)	List the outputs for a run
<i>list_sql_shares</i> (id)	List users and groups permissioned on this object
<i>list_types</i> ()	List available script types
<i>patch</i> (id, **kwargs)	Update a script
<i>patch_containers</i> (id, **kwargs)	Update a container
<i>patch_containers_runs</i> (id, run_id, **kwargs)	Update a run
<i>patch_custom</i> (id, **kwargs)	Update some attributes of this CustomScript
<i>patch_javascript</i> (id, **kwargs)	Update some attributes of this JavaScript Script
<i>patch_python3</i> (id, **kwargs)	Update some attributes of this Python Script
<i>patch_r</i> (id, **kwargs)	Update some attributes of this R Script
<i>patch_sql</i> (id, **kwargs)	Update some attributes of this SQL script
<i>post</i> (remote_host_id, name, sql, ...)	Create a script
<i>post_cancel</i> (id)	Cancel a run
<i>post_containers</i> (required_resources, ...)	Create a container
<i>post_containers_runs</i> (id)	Start a run
<i>post_containers_runs_heartbeats</i> (id, run_id)	Indicate that the given run is being handled
<i>post_containers_runs_logs</i> (id, run_id, **kwargs)	Add log messages
<i>post_containers_runs_outputs</i> (id, run_id, ...)	Add an output for a run
<i>post_custom</i> (from_template_id, **kwargs)	Create a CustomScript
<i>post_custom_runs</i> (id)	Start a run
<i>post_custom_runs_outputs</i> (id, run_id, ...)	Add an output for a run
<i>post_javascript</i> (remote_host_id, name, ...)	Create a JavaScript Script
<i>post_javascript_runs</i> (id)	Start a run
<i>post_javascript_runs_outputs</i> (id, run_id, ...)	Add an output for a run
<i>post_python3</i> (name, source, **kwargs)	Create a Python Script
<i>post_python3_runs</i> (id)	Start a run
<i>post_python3_runs_outputs</i> (id, run_id, ...)	Add an output for a run
<i>post_r</i> (name, source, **kwargs)	Create an R Script
<i>post_r_runs</i> (id)	Start a run
<i>post_r_runs_outputs</i> (id, run_id, object_id, ...)	Add an output for a run
<i>post_run</i> (id)	Run a script
<i>post_sql</i> (remote_host_id, sql, name, ...)	Create a SQL script
<i>post_sql_runs</i> (id)	Start a run
<i>put_containers</i> (id, required_resources, ...)	Edit a container
<i>put_containers_archive</i> (id, status)	Update the archive status of this object
<i>put_containers_projects</i> (id, project_id)	Add a container docker to a project
<i>put_containers_shares_groups</i> (id, ...)	Set the permissions groups has on this object
<i>put_containers_shares_users</i> (id, ...)	Set the permissions users have on this object
<i>put_custom</i> (id, **kwargs)	Replace all attributes of this CustomScript
<i>put_custom_archive</i> (id, status)	Update the archive status of this object
<i>put_custom_projects</i> (id, project_id)	Add a Job to a project

Continued on next page

Table 4.14 – continued from previous page

<code>put_custom_shares_groups(id, ...)</code>	Set the permissions groups has on this object
<code>put_custom_shares_users(id, ...)</code>	Set the permissions users have on this object
<code>put_javascript(id, remote_host_id, name, ...)</code>	Replace all attributes of this JavaScript Script
<code>put_javascript_archive(id, status)</code>	Update the archive status of this object
<code>put_javascript_projects(id, project_id)</code>	Add a scripted sql to a project
<code>put_javascript_shares_groups(id, ...)</code>	Set the permissions groups has on this object
<code>put_javascript_shares_users(id, ...)</code>	Set the permissions users have on this object
<code>put_python3(id, name, source, **kwargs)</code>	Replace all attributes of this Python Script
<code>put_python3_archive(id, status)</code>	Update the archive status of this object
<code>put_python3_projects(id, project_id)</code>	Add a python docker to a project
<code>put_python3_shares_groups(id, ...)</code>	Set the permissions groups has on this object
<code>put_python3_shares_users(id, ...)</code>	Set the permissions users have on this object
<code>put_r(id, name, source, **kwargs)</code>	Replace all attributes of this R Script
<code>put_r_archive(id, status)</code>	Update the archive status of this object
<code>put_r_projects(id, project_id)</code>	Add a r docker to a project
<code>put_r_shares_groups(id, permission_level, ...)</code>	Set the permissions groups has on this object
<code>put_r_shares_users(id, permission_level, ...)</code>	Set the permissions users have on this object
<code>put_sql(id, remote_host_id, sql, name, ...)</code>	Replace all attributes of this SQL script
<code>put_sql_archive(id, status)</code>	Update the archive status of this object
<code>put_sql_projects(id, project_id)</code>	Add a scripts to a project
<code>put_sql_shares_groups(id, permission_level, ...)</code>	Set the permissions groups has on this object
<code>put_sql_shares_users(id, permission_level, ...)</code>	Set the permissions users have on this object

**delete\_containers\_projects** (*id, project\_id*)

Remove a container docker from a project

**Parameters** *id* : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**delete\_containers\_runs** (*id, run\_id*)

Cancel a run

**Parameters** *id* : integer

The ID of the container.

**run\_id** : integer

The ID of the run.

**Returns** None

Response code 202: success

**delete\_containers\_shares\_groups** (*id, group\_id*)

Revoke the permissions a group has on this object

**Parameters** *id* : integer

ID of the resource to be revoked

**group\_id** : integer

ID of the group

**Returns** None

Response code 204: success

**delete\_containers\_shares\_users** (*id, user\_id*)

Revoke the permissions a user has on this object

**Parameters id** : integer

ID of the resource to be revoked

**user\_id** : integer

ID of the user

**Returns** None

Response code 204: success

**delete\_custom\_projects** (*id, project\_id*)

Remove a Job from a project

**Parameters id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**delete\_custom\_runs** (*id, run\_id*)

Cancel a run

**Parameters id** : integer

The ID of the custom.

**run\_id** : integer

The ID of the run.

**Returns** None

Response code 202: success

**delete\_custom\_shares\_groups** (*id, group\_id*)

Revoke the permissions a group has on this object

**Parameters id** : integer

ID of the resource to be revoked

**group\_id** : integer

ID of the group

**Returns** None

Response code 204: success

**delete\_custom\_shares\_users** (*id, user\_id*)

Revoke the permissions a user has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**user\_id** : integer

ID of the user

**Returns** None

Response code 204: success

**delete\_javascript\_projects** (*id, project\_id*)

Remove a scripted sql from a project

**Parameters** **id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**delete\_javascript\_runs** (*id, run\_id*)

Cancel a run

**Parameters** **id** : integer

The ID of the javascript.

**run\_id** : integer

The ID of the run.

**Returns** None

Response code 202: success

**delete\_javascript\_shares\_groups** (*id, group\_id*)

Revoke the permissions a group has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**group\_id** : integer

ID of the group

**Returns** None

Response code 204: success

**delete\_javascript\_shares\_users** (*id, user\_id*)

Revoke the permissions a user has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**user\_id** : integer

ID of the user

**Returns** None

Response code 204: success

**delete\_python3\_projects** (*id, project\_id*)

Remove a python docker from a project

**Parameters** **id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**delete\_python3\_runs** (*id, run\_id*)

Cancel a run

**Parameters** **id** : integer

The ID of the python.

**run\_id** : integer

The ID of the run.

**Returns** None

Response code 202: success

**delete\_python3\_shares\_groups** (*id, group\_id*)

Revoke the permissions a group has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**group\_id** : integer

ID of the group

**Returns** None

Response code 204: success

**delete\_python3\_shares\_users** (*id, user\_id*)

Revoke the permissions a user has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**user\_id** : integer

ID of the user

**Returns** None

Response code 204: success

**delete\_r\_projects** (*id, project\_id*)

Remove a r docker from a project

**Parameters** **id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**delete\_r\_runs** (*id, run\_id*)

Cancel a run

**Parameters** **id** : integer

The ID of the r.

**run\_id** : integer

The ID of the run.

**Returns** None

Response code 202: success

**delete\_r\_shares\_groups** (*id, group\_id*)

Revoke the permissions a group has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**group\_id** : integer

ID of the group

**Returns** None

Response code 204: success

**delete\_r\_shares\_users** (*id, user\_id*)

Revoke the permissions a user has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**user\_id** : integer

ID of the user

**Returns** None

Response code 204: success

**delete\_sql\_projects** (*id, project\_id*)

Remove a scripts from a project

**Parameters** **id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**delete\_sql\_runs** (*id, run\_id*)

Cancel a run

**Parameters** **id** : integer

The ID of the sql.

**run\_id** : integer

The ID of the run.

**Returns** None

Response code 202: success

**delete\_sql\_shares\_groups** (*id*, *group\_id*)

Revoke the permissions a group has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**group\_id** : integer

ID of the group

**Returns** None

Response code 204: success

**delete\_sql\_shares\_users** (*id*, *user\_id*)

Revoke the permissions a user has on this object

**Parameters** **id** : integer

ID of the resource to be revoked

**user\_id** : integer

ID of the user

**Returns** None

Response code 204: success

**get** (*id*)

Get details about a script

**Parameters** **id** : integer

The ID for the script.

**Returns** **projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments.
↳field.
- description : string
    A short sentence or fragment describing this parameter to the
↳end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
↳bool,
    file, database, credential_aws, credential_redshift, or
```



```

    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of script.

**sql** : string

The raw SQL query for the script.

**author** : dict:

```

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

```

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.

```

```
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time this script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**expanded\_arguments** : dict

Expanded arguments for use in injecting into different environments.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**template\_script\_id** : integer

The ID of the template script, if any.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**get\_containers** (*id*)

View a container

**Parameters** **id** : integer

The ID for the script.

**Returns** **projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float,_
↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this_
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g Container)

**docker\_image\_tag** : string

The tag of the docker image to pull from DockerHub (default: latest).

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↳successfully.
```

**remote\_host\_credential\_id** : integer

The id of the database credentials to pass into the environment of the container.

**repo\_ref** : string

The tag or branch of the github repo to clone into the container.

**template\_script\_name** : string

The name of the template script.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each_
↪core has
    1024 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB).
```

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**docker\_image\_name** : string

The name of the docker image to pull from DockerHub.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**from\_template\_id** : integer

The ID of the template script.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**finished\_at** : string/time

The time that the script's last run finished.

**id** : integer

The ID for the script.

**time\_zone** : string

The time zone of this script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the container.

**docker\_command** : string

The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]

**repo\_http\_uri** : string

The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**running\_as** : dict:

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

**git\_credential\_id** : 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.

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**get\_containers\_runs** (*id*, *run\_id*)

Check status of a run

**Parameters** *id* : integer

The ID of the container.

**run\_id** : integer

The ID of the run.

**Returns** *id* : integer

The ID of the run.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**container\_id** : integer

The ID of the container.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**get\_custom** (*id*)

Get a CustomScript

**Parameters** *id* : integer

**Returns** *projects* : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
field.
- description : string
    A short sentence or fragment describing this parameter to the_
end user.
- type : string
    The type of parameter. Valid options: string, integer, float, _
bool,
```



```

    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**type** : string

The type of the script (e.g Custom)

**author** : dict:

```

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

```

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list

```

Addresses to notify by e-mail when the job completes,  
↪ successfully.

**template\_script\_name** : string

The name of the template script.

**schedule** : dict:

- `scheduled` : boolean  
If the `object` `is` scheduled
- `scheduled_days` : `list`  
Day based on numeric value starting at 0 `for` Sunday
- `scheduled_minutes` : `list`  
Minutes of the day it `is` scheduled on
- `scheduled_hours` : `list`  
Hours of the day it `is` scheduled on
- `scheduled_runs_per_hour` : integer  
Alternative to scheduled minutes, number of times to run per,  
↪ hour

**created\_at** : string/time

The time this script was created.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**code\_preview** : string

The code that this script will run with arguments inserted.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**last\_run** : dict:

- `id` : integer
- `error` : string  
The error message `for` this run, `if` present.
- `started_at` : string/time  
The time that the run started.
- `created_at` : string/time  
The time that the run was queued.
- `state` : string
- `finished_at` : string/time  
The time that the run completed.

**finished\_at** : string/time

The time that the script's last run finished.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template script.

**get\_custom\_runs** (*id*, *run\_id*)

Check status of a run

**Parameters** **id** : integer

The ID of the custom.

**run\_id** : integer

The ID of the run.

**Returns** **id** : integer

The ID of the run.

**custom\_id** : integer

The ID of the custom.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**get\_javascript** (*id*)

Get a JavaScript Script

**Parameters** *id* : integer

**Returns** *projects* : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```
A definition of the parameters this script accepts in the arguments.
↳field.
- description : string
    A short sentence or fragment describing this parameter to the
↳end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
↳bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
↳value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**get\_javascript\_runs** (*id*, *run\_id*)

Check status of a run

**Parameters** **id** : integer

The ID of the javascript.

**run\_id** : integer

The ID of the run.

**Returns** **id** : integer

The ID of the run.

**javascript\_id** : integer

The ID of the javascript.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**get\_python3** (*id*)

Get a Python Script

**Parameters** *id* : integer

**Returns** *projects* : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments.
↳field.
- description : string
    A short sentence or fragment describing this parameter to the
↳end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
↳bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
↳value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

```
- id : integer
    The ID of this user.
- online : boolean
```



```

    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.

```

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.

```

**last\_run** : dict:

```

- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.

```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```

- scheduled : boolean
    If the object is scheduled

```

```
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    ↪hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
    ↪core has
    1024 shares. Must be at least 2 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB). Must
    ↪be at
    least 4 MiB.
```

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```

- 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.

```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**running\_as** : dict:

```

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

```

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**get\_python3\_runs** (*id*, *run\_id*)

Check status of a run

**Parameters** **id** : integer

The ID of the python.

**run\_id** : integer

The ID of the run.

**Returns** **id** : integer

The ID of the run.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**python\_id** : integer

The ID of the python.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**get\_r** (*id*)

Get an R Script

**Parameters** **id** : integer

**Returns** **projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↳field.
- description : string
    A short sentence or fragment describing this parameter to the_
↳end user.
- type : string
    The type of parameter. Valid options: string, integer, float, _
↳bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this_
↳value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes,
↪ successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each_
↪core has
    1024 shares. Must be at least 2 shares.
- memory : integer
```

```

    The amount of RAM to allocate for the container (in MiB). Must
    ↪be at
    least 4 MiB.

```

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```

- 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.

```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**running\_as** : dict:

```

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

```

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**get\_r\_runs** (*id*, *run\_id*)

Check status of a run

**Parameters** *id* : integer

The ID of the r.

**run\_id** : integer

The ID of the run.

**Returns id** : integer

The ID of the run.

**r\_id** : integer

The ID of the r.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**get\_sql** (*id*)

Get a SQL script

**Parameters id** : integer

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```
A definition of the parameters this script accepts in the arguments.
↳field.
- description : string
    A short sentence or fragment describing this parameter to the
↳end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
↳bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
    ↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**sql** : string

The raw SQL query for the script.

**author** : dict:

```

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

```

**csv\_settings** : dict:

```

- include_header : boolean
    Whether or not to include headers in the output data. Default:
    ↪true
- filename_prefix : string
    A user specified filename prefix for the output file to have.
    ↪Default:
    null
- unquoted : boolean
    Whether or not to quote fields. Default: false
- column_delimiter : string
    Which delimiter to use, one of "comma", "tab", or "pipe".
    ↪Default:
    comma
- compression : string
    The type of compression to use, if any, one of "none", "zip", or
    "gzip". Default: gzip
- force_multifile : boolean

```

```
Whether or not the csv should be split into multiple files.↵
↵Default:
false
```

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes.↵
↵successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
```

```

- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour

```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**code\_preview** : string

The code that this script will run with arguments inserted.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```

- 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.

```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**expanded\_arguments** : dict

Expanded arguments for use in injecting into different environments.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**get\_sql\_runs** (*id*, *run\_id*)

Check status of a run

**Parameters** **id** : integer

The ID of the sql.

**run\_id** : integer

The ID of the run.

**Returns** **id** : integer

The ID of this run.

**started\_at** : string/time

The time the last run started.

**output** : list:

```
A list of the outputs of this script.
- output_name : string
    The name of the output file.
- file_id : integer
    The unique ID of the output file.
- path : string
    The temporary link to download this output file, valid for 36
    ↪hours.
```

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error message for this run, if present.

**sql\_id** : integer

The ID of this sql.

**state** : string

The state of this run.

**finished\_at** : string/time

The time that this run finished.

**list** (*\*\*kwargs*)

List scripts

**Parameters** **type** : string, optional

If specified, return objects of these types. The valid types are 'sql', 'python3', 'r', and 'javascript'.

**author** : string, optional

If specified, return objects from this author. Must use user IDs. A comma separated list of IDs is also accepted to return objects from multiple authors.

**status** : string, optional

If specified, returns objects 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 object(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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**updated\_at** : string/time

The time the script was last updated.

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**parent\_id** : integer

The ID of the parent job that will trigger this script

**author** : dict:

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

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**links** : dict:

```
- 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.
```

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**id** : integer

The ID for the script.

**time\_zone** : string

The time zone of this script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**template\_script\_id** : integer

The ID of the template script, if any.

**created\_at** : string/time

The time this script was created.

**state** : string

The status of the script's last run.

**finished\_at** : string/time

The time that the script's last run finished.

**list\_containers\_projects** (*id*)

List the projects a container docker belongs to

**Parameters id** : integer

The ID of the resource.

**Returns id** : integer

The ID for this project.

**description** : string

A description of the project

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**created\_at** : string/time

**users** : list:

```
Users who can see the project
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.
```

**author** : dict:

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

```
This user's name.
- username : string
  This user's username.
```

**updated\_at** : string/time

**auto\_share** : boolean

**list\_containers\_runs** (*id*, *\*\*kwargs*)

List runs for the given container

**Parameters** **id** : integer

The ID of the container.

**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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **id** : integer

The ID of the run.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**container\_id** : integer

The ID of the container.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**list\_containers\_runs\_logs** (*id*, *run\_id*, *\*\*kwargs*)

Get the logs for a run



**Parameters** **id** : integer

The ID of the container.

**run\_id** : integer

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.

**Returns** **id** : integer

The ID of the log.

**created\_at** : string/date-time

The time the log was created.

**level** : string

The level of the log. One of unknown,fatal,error,warn,info,debug.

**message** : string

The log message.

**list\_containers\_runs\_outputs** (*id*, *run\_id*, *\*\*kwargs*)

List the outputs for a run

**Parameters** **id** : integer

The ID of the output.

**run\_id** : integer

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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** `link` : string

The link to retrieve the output object.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**name** : string

The name of the output object.

**list\_containers\_shares** (*id*)

List users and groups permissioned on this object

**Parameters** `id` : integer

The ID of the object.

**Returns** `readers` : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**list\_custom** (*\*\*kwargs*)

List Custom Scripts

**Parameters** `from_template_id` : integer, optional

The template script that this app uses.

**author** : string, optional

If specified, return objects from this author. Must use user IDs. A comma separated list of IDs is also accepted to return objects from multiple authors.

**status** : string, optional

If specified, returns objects 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 object(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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**updated\_at** : string/time

The time the script was last updated.

**type** : string

The type of the script (e.g Custom)

**parent\_id** : integer

The ID of the parent job that will trigger this script

**author** : dict:

```
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
```

```
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**from\_template\_id** : integer

The ID of the template script.

**id** : integer

The ID for the script.

**time\_zone** : string

The time zone of this script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**created\_at** : string/time

The time this script was created.

**state** : string

The status of the script's last run.

**finished\_at** : string/time

The time that the script's last run finished.

**list\_custom\_projects** (*id*)

List the projects a Job belongs to

**Parameters id** : integer

The ID of the resource.

**Returns id** : integer

The ID for this project.

**description** : string

A description of the project

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**created\_at** : string/time

**users** : list:

```

Users who can see the project
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.

```

**author** : dict:

```

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

```

**updated\_at** : string/time

**auto\_share** : boolean

**list\_custom\_runs** (*id*, *\*\*kwargs*)

List runs for the given custom

**Parameters** *id* : integer

The ID of the custom.

**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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns id** : integer

The ID of the run.

**custom\_id** : integer

The ID of the custom.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**list\_custom\_runs\_logs** (*id, run\_id, \*\*kwargs*)  
Get the logs for a run

**Parameters id** : integer

The ID of the custom.

**run\_id** : integer

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.

**Returns id** : integer

The ID of the log.

**created\_at** : string/date-time

The time the log was created.

**level** : string

The level of the log. One of unknown,fatal,error,warn,info,debug.

**message** : string

The log message.

**list\_custom\_runs\_outputs** (*id*, *run\_id*, *\*\*kwargs*)

List the outputs for a run

**Parameters** *id* : integer

The ID of the output.

**run\_id** : integer

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`.

**iterator** : bool, optional

If `True`, return a generator to iterate over all responses. Use when more results than the maximum allowed by `limit` are needed. When `True`, `limit` and `page_num` are ignored. Defaults to `False`.

**Returns** *link* : string

The link to retrieve the output object.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are `File`, `Report`, `Table`, or `Project`

**name** : string

The name of the output object.

**list\_custom\_shares** (*id*)

List users and groups permissioned on this object

**Parameters** *id* : integer

The ID of the object.

**Returns** *readers* : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**list\_history** (*id*)

Get the run history and outputs of this script

**Parameters** *id* : integer

The ID for the script.

**Returns** *id* : integer

The ID of this run.

**output** : list:

```
A list of the outputs of this script.
- output_name : string
  The name of the output file.
- file_id : integer
  The unique ID of the output file.
- path : string
  The temporary link to download this output file, valid for 36
↳ hours.
```

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error message for this run, if present.

**sql\_id** : integer

The ID of this sql.

**state** : string



The state of this run.

**finished\_at** : string/time

The time that this run finished.

**list\_javascript\_projects** (*id*)

List the projects a scripted sql belongs to

**Parameters** **id** : integer

The ID of the resource.

**Returns** **id** : integer

The ID for this project.

**description** : string

A description of the project

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**created\_at** : string/time

**users** : list:

```
Users who can see the project
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.
```

**author** : dict:

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

**updated\_at** : string/time

**auto\_share** : boolean

**list\_javascript\_runs** (*id*, **\*\*kwargs**)

List runs for the given javascript

**Parameters** **id** : integer

The ID of the javascript.

**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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **id** : integer

The ID of the run.

**javascript\_id** : integer

The ID of the javascript.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**list\_javascript\_runs\_logs** (*id*, *run\_id*, *\*\*kwargs*)

Get the logs for a run

**Parameters** **id** : integer

The ID of the javascript.

**run\_id** : integer

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.

**Returns id** : integer

The ID of the log.

**created\_at** : string/date-time

The time the log was created.

**level** : string

The level of the log. One of unknown,fatal,error,warn,info,debug.

**message** : string

The log message.

**list\_javascript\_runs\_outputs** (*id*, *run\_id*, *\*\*kwargs*)

List the outputs for a run

**Parameters id** : integer

The ID of the output.

**run\_id** : integer

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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns link** : string

The link to retrieve the output object.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**name** : string

The name of the output object.

**list\_javascript\_shares** (*id*)

List users and groups permissioned on this object

**Parameters** **id** : integer

The ID of the object.

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**list\_python3\_projects** (*id*)

List the projects a python docker belongs to

**Parameters** **id** : integer

The ID of the resource.

**Returns** **id** : integer

The ID for this project.

**description** : string

A description of the project

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**created\_at** : string/time

**users** : list:

```

Users who can see the project
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.

```

**author** : dict:

```

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

```

**updated\_at** : string/time

**auto\_share** : boolean

**list\_python3\_runs** (*id*, *\*\*kwargs*)

List runs for the given python

**Parameters** *id* : integer

The ID of the python.

**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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns id** : integer

The ID of the run.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**python\_id** : integer

The ID of the python.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**list\_python3\_runs\_logs** (*id*, *run\_id*, *\*\*kwargs*)

Get the logs for a run

**Parameters id** : integer

The ID of the python.

**run\_id** : integer

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.

**Returns id** : integer

The ID of the log.

**created\_at** : string/date-time

The time the log was created.

**level** : string

The level of the log. One of unknown,fatal,error,warn,info,debug.

**message** : string

The log message.

**list\_python3\_runs\_outputs** (*id*, *run\_id*, *\*\*kwargs*)

List the outputs for a run

**Parameters** *id* : integer

The ID of the output.

**run\_id** : integer

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`.

**iterator** : bool, optional

If `True`, return a generator to iterate over all responses. Use when more results than the maximum allowed by `limit` are needed. When `True`, `limit` and `page_num` are ignored. Defaults to `False`.

**Returns** *link* : string

The link to retrieve the output object.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are `File`, `Report`, `Table`, or `Project`

**name** : string

The name of the output object.

**list\_python3\_shares** (*id*)

List users and groups permissioned on this object

**Parameters** *id* : integer

The ID of the object.

**Returns** *readers* : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**list\_r\_projects** (*id*)

List the projects a r docker belongs to

**Parameters** **id** : integer

The ID of the resource.

**Returns** **id** : integer

The ID for this project.

**description** : string

A description of the project

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**created\_at** : string/time

**users** : list:

```
Users who can see the project
- id : integer
  The ID of this user.
- online : boolean
  Whether this user is online.
- initials : string
  This user's initials.
- name : string
  This user's name.
```



```
- username : string
    This user's username.
```

**author** : dict:

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

**updated\_at** : string/time

**auto\_share** : boolean

**list\_r\_runs** (*id*, *\*\*kwargs*)

List runs for the given r

**Parameters** **id** : integer

The ID of the r.

**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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **id** : integer

The ID of the run.

**r\_id** : integer

The ID of the r.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**list\_r\_runs\_logs** (*id*, *run\_id*, *\*\*kwargs*)

Get the logs for a run

**Parameters** **id** : integer

The ID of the r.

**run\_id** : integer

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.

**Returns** **id** : integer

The ID of the log.

**created\_at** : string/date-time

The time the log was created.

**level** : string

The level of the log. One of unknown,fatal,error,warn,info,debug.

**message** : string

The log message.

**list\_r\_runs\_outputs** (*id*, *run\_id*, *\*\*kwargs*)

List the outputs for a run

**Parameters** **id** : integer

The ID of the output.

**run\_id** : integer

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`.

**iterator** : bool, optional

If `True`, return a generator to iterate over all responses. Use when more results than the maximum allowed by `limit` are needed. When `True`, `limit` and `page_num` are ignored. Defaults to `False`.

**Returns link** : string

The link to retrieve the output object.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are `File`, `Report`, `Table`, or `Project`

**name** : string

The name of the output object.

**list\_r\_shares** (*id*)

List users and groups permissioned on this object

**Parameters id** : integer

The ID of the object.

**Returns readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

### **list\_sql\_projects** (*id*)

List the projects a scripts belongs to

**Parameters** **id** : integer

The ID of the resource.

**Returns** **id** : integer

The ID for this project.

**description** : string

A description of the project

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of this project.

**created\_at** : string/time

**users** : list:

```
Users who can see the project
- id : integer
  The ID of this user.
- online : boolean
  Whether this user is online.
- initials : string
  This user's initials.
- name : string
  This user's name.
- username : string
  This user's username.
```

**author** : dict:

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

**updated\_at** : string/time

**auto\_share** : boolean

**list\_sql\_runs** (*id*, *\*\*kwargs*)

List runs for the given sql

**Parameters** **id** : integer

The ID of the sql.

**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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **id** : integer

The ID of this run.

**started\_at** : string/time

The time the last run started.

**output** : list:

```
A list of the outputs of this script.
- output_name : string
  The name of the output file.
- file_id : integer
  The unique ID of the output file.
- path : string
  The temporary link to download this output file, valid for 36
  ↪ hours.
```

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error message for this run, if present.

**sql\_id** : integer

The ID of this sql.

**state** : string

The state of this run.

**finished\_at** : string/time

The time that this run finished.

**list\_sql\_runs\_logs** (*id*, *run\_id*, *\*\*kwargs*)

Get the logs for a run

**Parameters** *id* : integer

The ID of the sql.

**run\_id** : integer

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.

**Returns** *id* : integer

The ID of the log.

**created\_at** : string/date-time

The time the log was created.

**level** : string

The level of the log. One of unknown,fatal,error,warn,info,debug.

**message** : string

The log message.

**list\_sql\_runs\_outputs** (*id*, *run\_id*, *\*\*kwargs*)

List the outputs for a run

**Parameters** *id* : integer

The ID of the output.

**run\_id** : integer

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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** link : string

The link to retrieve the output object.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**name** : string

The name of the output object.

**list\_sql\_shares** (*id*)

List users and groups permissioned on this object

**Parameters** id : integer

The ID of the object.

**Returns** readers : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**list\_types()**

List available script types

**Returns** **name** : string

The name of the type.

**patch**(*id*, *\*\*kwargs*)

Update a script

**Parameters** **id** : integer

The ID for the script.

**params** : list, optional:

```
A definition of the parameters this script accepts in the arguments.
↳field.
Cannot be set if this script uses a template script.
- description : string
    A short sentence or fragment describing this parameter to the
↳end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
↳bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
↳value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**name** : string, optional

The name of the script.

**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.

**sql** : string, optional

The raw SQL query for the script.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.



**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes,
    successfully.
```

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments
    field.
- description : string
    A short sentence or fragment describing this parameter to the
    end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
    bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
```

```
The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
    ↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of script.

**sql** : string

The raw SQL query for the script.

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
```

```

- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.

```

**last\_run** : dict:

```

- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.

```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```

- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour

```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time this script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**expanded\_arguments** : dict

Expanded arguments for use in injecting into different environments.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**template\_script\_id** : integer

The ID of the template script, if any.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**patch\_containers** (*id*, *\*\*kwargs*)

## Update a container

**Parameters** `id` : integer

The ID for the script.

**params** : list, optional:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float,_
↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this_
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**remote\_host\_credential\_id** : integer, optional

The id of the database credentials to pass into the environment of the container.

**docker\_image\_tag** : string, optional

The tag of the docker image to pull from DockerHub (default: latest).

**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`.

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes,
    successfully.
```

**time\_zone** : string, optional

The time zone of this script.

**repo\_ref** : string, optional

The tag or branch of the github repo to clone into the container.

**required\_resources** : dict, optional:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
    core has
    1024 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB).
```

**name** : string, optional

The name of the container.

**docker\_command** : string, optional

The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**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\_image\_name** : string, optional

The name of the docker image to pull from DockerHub.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float, _
↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this_
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g Container)

**docker\_image\_tag** : string

The tag of the docker image to pull from DockerHub (default: latest).

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**remote\_host\_credential\_id** : integer

The id of the database credentials to pass into the environment of the container.



**repo\_ref** : string

The tag or branch of the github repo to clone into the container.

**template\_script\_name** : string

The name of the template script.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
    ↳ core has 1024 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB).
```

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    ↳ hour
```

**created\_at** : string/time

The time this script was created.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**docker\_image\_name** : string

The name of the docker image to pull from DockerHub.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**from\_template\_id** : integer

The ID of the template script.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**finished\_at** : string/time

The time that the script's last run finished.

**id** : integer

The ID for the script.

**time\_zone** : string

The time zone of this script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the container.

**docker\_command** : string

The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]

**repo\_http\_uri** : string

The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**running\_as** : dict:

```
- id : integer
    The ID of this user.
- online : boolean
    Whether this user is online.
- initials : string
```

```

    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.

```

**git\_credential\_id** : 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.

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**patch\_containers\_runs** (*id*, *run\_id*, *\*\*kwargs*)

Update a run

**Parameters** *id* : integer

The ID for the script.

**run\_id** : integer

The ID of the script run.

**state** : string, optional

The state of the script.

**bocce\_started\_at** : string/date-time, optional

The time when a bocce worker began executing the script.

**bocce\_accepted\_at** : string/date-time, optional

The time when a bocce worker began processing the script.

**Returns** None

Response code 204: success

**patch\_custom** (*id*, *\*\*kwargs*)

Update some attributes of this CustomScript

**Parameters** *id* : integer

The ID for the script.

**remote\_host\_id** : integer, optional

The remote host ID that this script will connect to.

**notifications** : dict, optional:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on

```

```
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**name** : string, optional

The name of the script.

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**credential\_id** : integer, optional

The credential that this script will use.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```

A definition of the parameters this script accepts in the arguments_
↳field.
- description : string
    A short sentence or fragment describing this parameter to the_
↳end user.
- type : string
    The type of parameter. Valid options: string, integer, float,_
↳bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this_
↳value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**type** : string

The type of the script (e.g Custom)

**author** : dict:

```

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

```

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string

```

```
Custom subject line for success e-mail.
- failure_on : boolean
  If failure email notifications are on
- urls : list
  URLs to receive a POST request at job completion
- success_email_body : string
  Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
  Addresses to notify by e-mail when the job completes_
↳successfully.
```

**template\_script\_name** : string

The name of the template script.

**schedule** : dict:

```
- scheduled : boolean
  If the object is scheduled
- scheduled_days : list
  Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
  Minutes of the day it is scheduled on
- scheduled_hours : list
  Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
  Alternative to scheduled minutes, number of times to run per_
↳hour
```

**created\_at** : string/time

The time this script was created.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**code\_preview** : string

The code that this script will run with arguments inserted.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**last\_run** : dict:

```
- id : integer
- error : string
  The error message for this run, if present.
- started_at : string/time
```

```

    The time that the run started.
-   created_at : string/time
    The time that the run was queued.
-   state : string
-   finished_at : string/time
    The time that the run completed.

```

**finished\_at** : string/time

The time that the script's last run finished.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

```

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

```

**from\_template\_id** : integer

The ID of the template script.

**patch\_javascript** (*id*, *\*\*kwargs*)

Update some attributes of this JavaScript Script

**Parameters** **id** : integer

The ID for the script.

**remote\_host\_id** : integer, optional

The remote host ID that this script will connect to.

**params** : list, optional:

```

A definition of the parameters this script accepts in the arguments.
↪field.
-   description : string

```

```
    A short sentence or fragment describing this parameter to the
    ↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
    ↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
    ↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes
    ↪successfully.
```

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.



**name** : string, optional

The name of the script.

**source** : string, optional

The body/text of the script.

**credential\_id** : integer, optional

The credential that this script will use.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float, _
↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes.
↪successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**patch\_python3** (*id*, **\*\*kwargs**)

Update some attributes of this Python Script

**Parameters** **id** : integer

The ID for the script.

**params** : list, optional:

A definition of the parameters this script accepts **in** the arguments.

- ↪ **field**.
- **description** : string
  - A short sentence **or** fragment describing this parameter to the
  - ↪ end user.
- **type** : string
  - The **type** of parameter. Valid options: string, integer, **float**,
  - ↪ **bool**,
  - file, database, credential\_aws, credential\_redshift, **or**
  - credential\_custom
- **name** : string
  - The variable's name as used within your code.
- **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**.
- **value** : string
  - The value you would like to **set** this param to. Setting this
  - ↪ value makes
  - this parameter a fixed param.
- **label** : string
  - The label to present to users when asking them **for** the value.
- **required** : boolean
  - Whether this param **is** required.

**required\_resources** : dict, optional:

- **disk\_space** : number/float
  - 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.
- **cpu** : integer
  - The number of CPU shares to allocate **for** the container. Each
  - ↪ core has
  - 1024 shares. Must be at least 2 shares.
- **memory** : integer
  - The amount of RAM to allocate **for** the container (**in** MiB). Must
  - ↪ be at
  - least 4 MiB.

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes,
↳ successfully.
```

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**name** : string, optional

The name of the script.

**source** : string, optional

The body/text of the script.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per,
↳ hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
```

```
- name : string
    The name of the project.
```

**params** : list:

A definition of the parameters this script accepts **in** the arguments\_↵  
↵field.

```
- description : string
    A short sentence or fragment describing this parameter to the_↵
    ↵end user.
- type : string
    The type of parameter. Valid options: string, integer, float,_↵
    ↵bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this_↵
    ↵value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.



**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
↪core has
    1024 shares. Must be at least 2 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB). Must
↪be at
    least 4 MiB.
```

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**patch\_r** (*id*, **\*\*kwargs**)

Update some attributes of this R Script

**Parameters** **id** : integer

The ID for the script.

**params** : list, optional:

```
A definition of the parameters this script accepts in the arguments.
↪field.
- description : string
    A short sentence or fragment describing this parameter to the
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
↪value makes
```

```

    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**required\_resources** : dict, optional:

```

- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
↪core has
    1024 shares. Must be at least 2 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB). Must
↪be at
    least 4 MiB.

```

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes
↪successfully.

```

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**name** : string, optional

The name of the script.

**source** : string, optional

The body/text of the script.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float,_
↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
```

```

- value : string
    The value you would like to set this param to. Setting this_
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

```

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

```

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.

```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
  - cpu : integer
    The number of CPU shares to allocate for the container. Each
    ↪ core has
      1024 shares. Must be at least 2 shares.
  - memory : integer
    The amount of RAM to allocate for the container (in MiB). Must
    ↪ be at
      least 4 MiB.

```

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```

- 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.

```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**running\_as** : dict:

```

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

```

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**patch\_sql** (*id*, *\*\*kwargs*)

Update some attributes of this SQL script

**Parameters** *id* : integer

The ID for the script.

**remote\_host\_id** : integer, optional

The remote host ID that this script will connect to.

**params** : list, optional:

A definition of the parameters this script accepts **in** the arguments.

- ↪ **field**.
- **description** : string
  - A short sentence **or** fragment describing this parameter to the
  - ↪ **end user**.
- **type** : string
  - The **type** of parameter. Valid options: string, integer, float,
  - ↪ **bool**,
  - file, database, credential\_aws, credential\_redshift, **or**
  - credential\_custom
- **name** : string
  - The variable's **name** as used within your code.
- **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**.
- **value** : string
  - The value you would like to **set** this param to. Setting this
  - ↪ **value** makes
  - this parameter a fixed param.
- **label** : string
  - The label to present to users when asking them **for** the value.
- **required** : boolean
  - Whether this param **is** required.

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**sql** : string, optional

The raw SQL query for the script.

**csv\_settings** : dict, optional:

- **include\_header** : boolean
  - Whether **or not** to include headers **in** the output data. Default:
  - ↪ **true**
- **filename\_prefix** : string
  - A user specified filename prefix **for** the output file to have.
  - ↪ **Default:**



```

    null
- unquoted : boolean
    Whether or not to quote fields. Default: false
- column_delimiter : string
    Which delimiter to use, one of "comma", "tab", or "pipe".
↪Default:
    comma
- compression : string
    The type of compression to use, if any, one of "none", "zip", or
    "gzip". Default: gzip
- force_multifile : boolean
    Whether or not the csv should be split into multiple files.
↪Default:
    false

```

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes.
↪successfully.

```

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**name** : string, optional

The name of the script.

**credential\_id** : integer, optional

The credential that this script will use.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```
A definition of the parameters this script accepts in the arguments
    field.
- description : string
    A short sentence or fragment describing this parameter to the
    end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
    bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
    value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**sql** : string

The raw SQL query for the script.

**author** : dict:

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

**csv\_settings** : dict:

```
- include_header : boolean
    Whether or not to include headers in the output data. Default:
↳ true
- filename_prefix : string
    A user specified filename prefix for the output file to have.
↳ Default:
    null
- unquoted : boolean
    Whether or not to quote fields. Default: false
- column_delimiter : string
    Which delimiter to use, one of "comma", "tab", or "pipe".
↳ Default:
    comma
- compression : string
    The type of compression to use, if any, one of "none", "zip", or
    "gzip". Default: gzip
- force_multifile : boolean
    Whether or not the csv should be split into multiple files.
↳ Default:
    false
```

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
```

```
Custom subject line for success e-mail.
- failure_on : boolean
  If failure email notifications are on
- urls : list
  URLs to receive a POST request at job completion
- success_email_body : string
  Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
  Addresses to notify by e-mail when the job completes_
↳successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
  The error message for this run, if present.
- started_at : string/time
  The time that the run started.
- created_at : string/time
  The time that the run was queued.
- state : string
- finished_at : string/time
  The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
  If the object is scheduled
- scheduled_days : list
  Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
  Minutes of the day it is scheduled on
- scheduled_hours : list
  Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
  Alternative to scheduled minutes, number of times to run per_
↳hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**code\_preview** : string

The code that this script will run with arguments inserted.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**expanded\_arguments** : dict

Expanded arguments for use in injecting into different environments.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

```
- username : string
    This user's username.
```

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**post** (*remote\_host\_id*, *name*, *sql*, *credential\_id*, *\*\*kwargs*)  
Create a script

**Parameters** **remote\_host\_id** : integer

The database ID.

**name** : string

The name of the script.

**sql** : string

The raw SQL query for the script.

**credential\_id** : integer

The credential ID.

**params** : list, optional:

```
A definition of the parameters this script accepts in the arguments,
↳field.
Cannot be set if this script uses a template script.
- description : string
    A short sentence or fragment describing this parameter to the
↳end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
↳bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
↳value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**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.

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float, _
↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
```

```
- value : string
    The value you would like to set this param to. Setting this_
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**last\_run** : dict:



```

- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.

```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```

- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour

```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**template\_script\_id** : integer

The ID of the template script, if any.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**post\_cancel** (*id*)

Cancel a run

**Parameters id** : integer

The ID of the job.

**Returns id** : integer

The ID of the run.

**state** : string

The state of the run, one of 'queued', 'running' or 'cancelled'.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**post\_containers** (*required\_resources*, *docker\_command*, *docker\_image\_name*, *\*\*kwargs*)  
Create a container

**Parameters** *required\_resources* : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
↪core has
    1024 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB).
```

**docker\_command** : string

The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]

**docker\_image\_name** : string

The name of the docker image to pull from DockerHub.

**params** : list, optional:

```
A definition of the parameters this script accepts in the arguments.
↪field.
- description : string
    A short sentence or fragment describing this parameter to the
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**remote\_host\_credential\_id** : integer, optional

The id of the database credentials to pass into the environment of the container.

**docker\_image\_tag** : string, optional

The tag of the docker image to pull from DockerHub (default: latest).

**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.

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**time\_zone** : string, optional

The time zone of this script.

**repo\_ref** : string, optional

The tag or branch of the github repo to clone into the container.

**name** : string, optional

The name of the container.

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**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.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float, _
↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this_
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
```

```
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g Container)

**docker\_image\_tag** : string

The tag of the docker image to pull from DockerHub (default: latest).

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↵successfully.
```

**remote\_host\_credential\_id** : integer

The id of the database credentials to pass into the environment of the container.

**repo\_ref** : string

The tag or branch of the github repo to clone into the container.

**template\_script\_name** : string

The name of the template script.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
    ↳ core has 1024 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB).
```

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    ↳ hour
```

**created\_at** : string/time

The time this script was created.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**docker\_image\_name** : string

The name of the docker image to pull from DockerHub.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**from\_template\_id** : integer

The ID of the template script.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**finished\_at** : string/time

The time that the script's last run finished.

**id** : integer

The ID for the script.

**time\_zone** : string

The time zone of this script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the container.

**docker\_command** : string

The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]

**repo\_http\_uri** : string

The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**running\_as** : dict:

```
- id : integer
    The ID of this user.
- online : boolean
```



```

    Whether this user is online.
-   initials : string
        This user's initials.
-   name : string
        This user's name.
-   username : string
        This user's username.

```

**git\_credential\_id** : 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.

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**post\_containers\_runs** (*id*)

Start a run

**Parameters** **id** : integer

The ID of the container.

**Returns** **id** : integer

The ID of the run.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**container\_id** : integer

The ID of the container.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**post\_containers\_runs\_heartbeats** (*id*, *run\_id*)

Indicate that the given run is being handled

**Parameters** **id** : integer

The ID of the container.

**run\_id** : integer

The ID of the run.

**Returns** None

Response code 204: success

**post\_containers\_runs\_logs** (*id, run\_id, \*\*kwargs*)

Add log messages

**Parameters** **id** : integer

The ID of the script.

**run\_id** : integer

The ID of the script run.

**message** : string, optional

The log message to store.

**messages** : list, optional:

- **message** : string  
The log message to store.
- **level** : string  
The log level of this message [default: info]
- **created\_at** : string/date-time

**level** : string, optional

The log level of this message [default: info]

**Returns** None

Response code 204: success

**post\_containers\_runs\_outputs** (*id, run\_id, object\_id, object\_type*)

Add an output for a run

**Parameters** **id** : integer

The ID of the output.

**run\_id** : integer

The ID of the run.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**Returns** **link** : string

The link to retrieve the output object.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**name** : string

The name of the output object.

**post\_custom** (*from\_template\_id, \*\*kwargs*)

Create a CustomScript

**Parameters from\_template\_id** : integer

The ID of the template script.

**remote\_host\_id** : integer, optional

The remote host ID that this script will connect to.

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**time\_zone** : string, optional

The time zone of this script.

**name** : string, optional

The name of the script.

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**credential\_id** : integer, optional

The credential that this script will use.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
```

```
Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**Returns projects :** list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**remote\_host\_id :** integer

The remote host ID that this script will connect to.

**params :** list:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float_,
↪bool_,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this_
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**type :** string

The type of the script (e.g Custom)

**author :** dict:

```
- id : integer
    The ID of this user.
- online : boolean
```

```

    Whether this user is online.
- initials : string
    This user's initials.
- name : string
    This user's name.
- username : string
    This user's username.

```

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.

```

**template\_script\_name** : string

The name of the template script.

**schedule** : dict:

```

- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour

```

**created\_at** : string/time

The time this script was created.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**code\_preview** : string

The code that this script will run with arguments inserted.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**finished\_at** : string/time

The time that the script's last run finished.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

```
- username : string
    This user's username.
```

**from\_template\_id** : integer

The ID of the template script.

**post\_custom\_runs** (*id*)

Start a run

**Parameters id** : integer

The ID of the custom.

**Returns id** : integer

The ID of the run.

**custom\_id** : integer

The ID of the custom.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**post\_custom\_runs\_outputs** (*id, run\_id, object\_id, object\_type*)

Add an output for a run

**Parameters id** : integer

The ID of the output.

**run\_id** : integer

The ID of the run.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**Returns link** : string

The link to retrieve the output object.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**name** : string

The name of the output object.

**post\_javascript** (*remote\_host\_id*, *name*, *source*, *credential\_id*, *\*\*kwargs*)  
Create a JavaScript Script

**Parameters** **remote\_host\_id** : integer

The remote host ID that this script will connect to.

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**credential\_id** : integer

The credential that this script will use.

**params** : list, optional:

A definition of the parameters this script accepts **in** the arguments.

- ↪ **field**.
- **description** : string
  - A short sentence **or** fragment describing this parameter to the
  - ↪ end user.
- **type** : string
  - The **type** of parameter. Valid options: string, integer, float,
  - ↪ bool,
  - file, database, credential\_aws, credential\_redshift, **or**
  - credential\_custom
- **name** : string
  - The variable's name as used within your code.
- **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**.
- **value** : string
  - The value you would like to **set** this param to. Setting this
  - ↪ value makes
  - this parameter a fixed param.
- **label** : string
  - The label to present to users when asking them **for** the value.
- **required** : boolean
  - Whether this param **is** required.

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**user\_context** : string, optional



“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
```

```
The ID for the project.
- name : string
  The name of the project.
```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```
A definition of the parameters this script accepts in the arguments.
↳field.
- description : string
  A short sentence or fragment describing this parameter to the
↳end user.
- type : string
  The type of parameter. Valid options: string, integer, float,
↳bool,
  file, database, credential_aws, credential_redshift, or
  credential_custom
- name : string
  The variable's name as used within your code.
- 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.
- value : string
  The value you would like to set this param to. Setting this
↳value makes
  this parameter a fixed param.
- label : string
  The label to present to users when asking them for the value.
- required : boolean
  Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes,
    ↪ successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per,
    ↪ hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**post\_javascript\_runs** (*id*)

Start a run

**Parameters** **id** : integer

The ID of the javascript.

**Returns** **id** : integer

The ID of the run.

**javascript\_id** : integer

The ID of the javascript.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**post\_javascript\_runs\_outputs** (*id, run\_id, object\_id, object\_type*)

Add an output for a run

**Parameters** **id** : integer

The ID of the output.

**run\_id** : integer

The ID of the run.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**Returns** **link** : string

The link to retrieve the output object.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**name** : string

The name of the output object.

**post\_python3** (*name*, *source*, *\*\*kwargs*)

Create a Python Script

**Parameters** **name** : string

The name of the script.

**source** : string

The body/text of the script.

**params** : list, optional:

```
A definition of the parameters this script accepts in the arguments.
↳field.
- description : string
  A short sentence or fragment describing this parameter to the
↳end user.
- type : string
  The type of parameter. Valid options: string, integer, float,
↳bool,
  file, database, credential_aws, credential_redshift, or
  credential_custom
- name : string
  The variable's name as used within your code.
- 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.
- value : string
  The value you would like to set this param to. Setting this
↳value makes
  this parameter a fixed param.
- label : string
  The label to present to users when asking them for the value.
- required : boolean
  Whether this param is required.
```

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes
↳ successfully.
```

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**required\_resources** : dict, optional:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
↳ core has
    1024 shares. Must be at least 2 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB). Must
↳ be at
    least 4 MiB.
```

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float,_
↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this_
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.



**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each_
↪core has
    1024 shares. Must be at least 2 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB). Must_
↪be at
    least 4 MiB.
```

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**post\_python3\_runs** (*id*)

Start a run

**Parameters id** : integer

The ID of the python.

**Returns id** : integer

The ID of the run.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**python\_id** : integer

The ID of the python.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**post\_python3\_runs\_outputs** (*id, run\_id, object\_id, object\_type*)

Add an output for a run

**Parameters id** : integer

The ID of the output.

**run\_id** : integer

The ID of the run.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**Returns link** : string

The link to retrieve the output object.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**name** : string

The name of the output object.

**post\_r** (*name, source, \*\*kwargs*)

Create an R Script

**Parameters name** : string

The name of the script.

**source** : string

The body/text of the script.

**params** : list, optional:

A definition of the parameters this script accepts **in** the arguments.  
 ↳field.

- description : string  
 A short sentence **or** fragment describing this parameter to the  
 ↳end user.
- type : string  
 The **type** of parameter. Valid options: string, integer, **float**,  
 ↳bool,  
 file, database, credential\_aws, credential\_redshift, **or**  
 credential\_custom
- name : string  
 The variable's name as used within your code.
- 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**.
- value : string  
 The value you would like to **set** this param to. Setting this  
 ↳value makes  
 this parameter a fixed param.
- label : string  
 The label to present to users when asking them **for** the value.
- required : boolean  
 Whether this param **is** required.

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

- success\_on : boolean  
 If success email notifications are on
- stall\_warning\_minutes : integer  
 Stall warning emails will be sent after this amount of minutes.
- failure\_email\_addresses : list  
 Addresses to notify by e-mail when the job fails.
- success\_email\_subject : string  
 Custom subject line **for** success e-mail.
- failure\_on : boolean  
 If failure email notifications are on
- urls : list  
 URLs to receive a POST request at job completion
- success\_email\_body : string  
 Custom body text **for** success e-mail, written **in** Markdown.
- success\_email\_addresses : list  
 Addresses to notify by e-mail when the job completes  
 ↳successfully.

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**required\_resources** : dict, optional:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
↪core has
    1024 shares. Must be at least 2 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB). Must
↪be at
    least 4 MiB.
```

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
↪hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

A definition of the parameters this script accepts **in** the arguments.

- ↪field.
- description : string
  - A short sentence **or** fragment describing this parameter to the
  - ↪end user.
- type : string
  - The **type** of parameter. Valid options: string, integer, float,
  - ↪bool,
  - file, database, credential\_aws, credential\_redshift, **or**
  - credential\_custom
- name : string
  - The variable's **name** as used within your code.
- 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**.
- value : string
  - The value you would like to **set** this param to. Setting this
  - ↪value makes
  - this parameter a fixed param.
- label : string
  - The label to present to users when asking them **for** the value.
- required : boolean
  - Whether this param **is** required.

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes,
↳successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per,
↳hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.



**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
    ↳ core has 1024 shares. Must be at least 2 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB). Must
    ↳ be at least 4 MiB.
```

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**post\_r\_runs** (*id*)

Start a run

**Parameters id** : integer

The ID of the r.

**Returns id** : integer

The ID of the run.

**r\_id** : integer

The ID of the r.

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error, if any, returned by the run.

**started\_at** : string/time

The time the last run started at.

**state** : string

The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**finished\_at** : string/time

The time the last run completed.

**post\_r\_runs\_outputs** (*id, run\_id, object\_id, object\_type*)

Add an output for a run

**Parameters id** : integer

The ID of the output.

**run\_id** : integer

The ID of the run.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**Returns link** : string

The link to retrieve the output object.

**object\_id** : integer

The ID of the output object.

**object\_type** : string

The type of the output. Valid values are File, Report, Table, or Project

**name** : string

The name of the output object.

**post\_run** (*id*)

Run a script

**Parameters id** : integer

The ID for the script.

**Returns** None

Response code 204: success

**post\_sql** (*remote\_host\_id, sql, name, credential\_id, \*\*kwargs*)

Create a SQL script

**Parameters remote\_host\_id** : integer

The remote host ID that this script will connect to.

**sql** : string

The raw SQL query for the script.

**name** : string

The name of the script.

**credential\_id** : integer

The credential that this script will use.

**params** : list, optional:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
  A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
  The type of parameter. Valid options: string, integer, float, _
↪bool,
```

```
file, database, credential_aws, credential_redshift, or
credential_custom
- name : string
  The variable's name as used within your code.
- 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.
- value : string
  The value you would like to set this param to. Setting this
↳value makes
  this parameter a fixed param.
- label : string
  The label to present to users when asking them for the value.
- required : boolean
  Whether this param is required.
```

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**csv\_settings** : dict, optional:

```
- include_header : boolean
  Whether or not to include headers in the output data. Default:
↳true
- filename_prefix : string
  A user specified filename prefix for the output file to have.
↳Default:
  null
- unquoted : boolean
  Whether or not to quote fields. Default: false
- column_delimiter : string
  Which delimiter to use, one of "comma", "tab", or "pipe".
↳Default:
  comma
- compression : string
  The type of compression to use, if any, one of "none", "zip", or
  "gzip". Default: gzip
- force_multifile : boolean
  Whether or not the csv should be split into multiple files.
↳Default:
  false
```

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```
- success_on : boolean
  If success email notifications are on
```

```

- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes,
↳ successfully.

```

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**hidden** : boolean, optional

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```

- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per,
↳ hour

```

**Returns projects** : list:

```

A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.

```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```
A definition of the parameters this script accepts in the arguments.
↳field.
- description : string
    A short sentence or fragment describing this parameter to the
↳end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
↳bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
↳value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**sql** : string

The raw SQL query for the script.

**author** : dict:

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

**csv\_settings** : dict:

```
- include_header : boolean
    Whether or not to include headers in the output data. Default:
↳true
- filename_prefix : string
```

```

    A user specified filename prefix for the output file to have.
↪Default:
    null
- unquoted : boolean
    Whether or not to quote fields. Default: false
- column_delimiter : string
    Which delimiter to use, one of "comma", "tab", or "pipe".
↪Default:
    comma
- compression : string
    The type of compression to use, if any, one of "none", "zip", or
    "gzip". Default: gzip
- force_multifile : boolean
    Whether or not the csv should be split into multiple files.
↪Default:
    false

```

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes.
↪successfully.

```

**last\_run** : dict:

```

- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.

```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**code\_preview** : string

The code that this script will run with arguments inserted.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.



**id** : integer

The ID for the script.

**expanded\_arguments** : dict

Expanded arguments for use in injecting into different environments.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**post\_sql\_runs** (*id*)

Start a run

**Parameters** **id** : integer

The ID of the sql.

**Returns** **id** : integer

The ID of this run.

**started\_at** : string/time

The time the last run started.

**output** : list:

```
A list of the outputs of this script.
- output_name : string
    The name of the output file.
- file_id : integer
    The unique ID of the output file.
- path : string
    The temporary link to download this output file, valid for 36
hours.
```

**is\_cancel\_requested** : boolean

True if run cancel requested, else false.

**error** : string

The error message for this run, if present.

**sql\_id** : integer

The ID of this sql.

**state** : string

The state of this run.

**finished\_at** : string/time

The time that this run finished.

**put\_containers** (*id, required\_resources, docker\_command, docker\_image\_name, \*\*kwargs*)

Edit a container

**Parameters** **id** : integer

The ID for the script.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
↪ core has
    1024 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB).
```

**docker\_command** : string

The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]

**docker\_image\_name** : string

The name of the docker image to pull from DockerHub.

**params** : list, optional:

```
A definition of the parameters this script accepts in the arguments.
↪ field.
- description : string
    A short sentence or fragment describing this parameter to the
↪ end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
↪ bool,
```

```

    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
    ↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**remote\_host\_credential\_id** : integer, optional

The id of the database credentials to pass into the environment of the container.

**docker\_image\_tag** : string, optional

The tag of the docker image to pull from DockerHub (default: latest).

**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.

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.

```

```
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**time\_zone** : string, optional

The time zone of this script.

**repo\_ref** : string, optional

The tag or branch of the github repo to clone into the container.

**name** : string, optional

The name of the container.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**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.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float, _
↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
```

```

- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
    ↪value makes this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g Container)

**docker\_image\_tag** : string

The tag of the docker image to pull from DockerHub (default: latest).

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**author** : dict:

```

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

```

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer

```

```
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**remote\_host\_credential\_id** : integer

The id of the database credentials to pass into the environment of the container.

**repo\_ref** : string

The tag or branch of the github repo to clone into the container.

**template\_script\_name** : string

The name of the template script.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each_
↪core has
    1024 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB).
```

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**docker\_image\_name** : string

The name of the docker image to pull from DockerHub.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**from\_template\_id** : integer

The ID of the template script.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**finished\_at** : string/time

The time that the script's last run finished.

**id** : integer

The ID for the script.

**time\_zone** : string

The time zone of this script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the container.

**docker\_command** : string

The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]

**repo\_http\_uri** : string

The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**running\_as** : dict:

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

**git\_credential\_id** : 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.

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**put\_containers\_archive** (*id*, *status*)

Update the archive status of this object

**Parameters** *id* : integer

The ID of the object.

**status** : boolean

The desired archived status of the object.

**Returns** *projects* : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments_  
↪field.  
- description : string  
    A short sentence or fragment describing this parameter to the_  
↪end user.  
- type : string
```



```

    The type of parameter. Valid options: string, integer, float,
    ↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
    ↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g Container)

**docker\_image\_tag** : string

The tag of the docker image to pull from DockerHub (default: latest).

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**author** : dict:

```

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

```

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↳successfully.
```

**remote\_host\_credential\_id** : integer

The id of the database credentials to pass into the environment of the container.

**repo\_ref** : string

The tag or branch of the github repo to clone into the container.

**template\_script\_name** : string

The name of the template script.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each_
↳core has
    1024 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB).
```

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↳hour
```

**created\_at** : string/time

The time this script was created.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**docker\_image\_name** : string

The name of the docker image to pull from DockerHub.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**from\_template\_id** : integer

The ID of the template script.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**finished\_at** : string/time

The time that the script's last run finished.

**id** : integer

The ID for the script.

**time\_zone** : string

The time zone of this script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the container.

**docker\_command** : string

The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]

**repo\_http\_uri** : string

The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**running\_as** : dict:

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

**git\_credential\_id** : 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.

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**put\_containers\_projects** (*id, project\_id*)

Add a container docker to a project

**Parameters** **id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**put\_containers\_shares\_groups** (*id, permission\_level, group\_ids*)

Set the permissions groups has on this object

**Parameters** **id** : integer

ID of the resource to be shared

**permission\_level** : string

Options are: "read", "write", or "manage"

**group\_ids** : list

An array of one or more group IDs

**Returns** `readers` : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_containers\_shares\_users** (*id*, *permission\_level*, *user\_ids*)

Set the permissions users have on this object

**Parameters** `id` : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**user\_ids** : list

An array of one or more user IDs

**Returns** `readers` : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_custom** (*id*, *\*\*kwargs*)

Replace all attributes of this CustomScript

**Parameters** *id* : integer

The ID for the script.

**remote\_host\_id** : integer, optional

The remote host ID that this script will connect to.

**notifications** : dict, optional:

```
- success_on : boolean
  If success email notifications are on
- stall_warning_minutes : integer
  Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
  Addresses to notify by e-mail when the job fails.
- success_email_subject : string
  Custom subject line for success e-mail.
- failure_on : boolean
  If failure email notifications are on
- urls : list
  URLs to receive a POST request at job completion
- success_email_body : string
  Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
  Addresses to notify by e-mail when the job completes_
↪successfully.
```

**name** : string, optional

The name of the script.

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**credential\_id** : integer, optional

The credential that this script will use.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↳hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↳field.
- description : string
    A short sentence or fragment describing this parameter to the_
↳end user.
- type : string
    The type of parameter. Valid options: string, integer, float, _
↳bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
  The value you would like to set this param to. Setting this.
↪value makes
  this parameter a fixed param.
- label : string
  The label to present to users when asking them for the value.
- required : boolean
  Whether this param is required.
```

**type** : string

The type of the script (e.g Custom)

**author** : dict:

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

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
  If success email notifications are on
- stall_warning_minutes : integer
  Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
  Addresses to notify by e-mail when the job fails.
- success_email_subject : string
  Custom subject line for success e-mail.
- failure_on : boolean
  If failure email notifications are on
- urls : list
  URLs to receive a POST request at job completion
- success_email_body : string
  Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
  Addresses to notify by e-mail when the job completes.
↪successfully.
```

**template\_script\_name** : string

The name of the template script.

**schedule** : dict:



```

- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour

```

**created\_at** : string/time

The time this script was created.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**code\_preview** : string

The code that this script will run with arguments inserted.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**last\_run** : dict:

```

- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.

```

**finished\_at** : string/time

The time that the script's last run finished.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template script.

**put\_custom\_archive** (*id*, *status*)

Update the archive status of this object

**Parameters** **id** : integer

The ID of the object.

**status** : boolean

The desired archived status of the object.

**Returns** **projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float, _
↪bool,
    file, database, credential_aws, credential_redshift, or
```

```

credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
    ↪value makes this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**type : string**

The type of the script (e.g Custom)

**author : dict:**

```

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

```

**parent\_id : integer**

The ID of the parent job that will trigger this script

**notifications : dict:**

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes
    ↪successfully.

```

**template\_script\_name** : string

The name of the template script.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**code\_preview** : string

The code that this script will run with arguments inserted.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**finished\_at** : string/time

The time that the script's last run finished.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template script.

**put\_custom\_projects** (*id*, *project\_id*)

Add a Job to a project

**Parameters** **id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**put\_custom\_shares\_groups** (*id*, *permission\_level*, *group\_ids*)

Set the permissions groups has on this object

**Parameters** **id** : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**group\_ids** : list

An array of one or more group IDs

**Returns** `readers` : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_custom\_shares\_users** (*id*, *permission\_level*, *user\_ids*)

Set the permissions users have on this object

**Parameters** *id* : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**user\_ids** : list

An array of one or more user IDs

**Returns** `readers` : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_javascript** (*id*, *remote\_host\_id*, *name*, *source*, *credential\_id*, *\*\*kwargs*)

Replace all attributes of this JavaScript Script

**Parameters** **id** : integer

The ID for the script.

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**credential\_id** : integer

The credential that this script will use.

**params** : list, optional:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
  A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
  The type of parameter. Valid options: string, integer, float, _
↪bool,
  file, database, credential_aws, credential_redshift, or
  credential_custom
- name : string
  The variable's name as used within your code.
```

```
- 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.
- value : string
    The value you would like to set this param to. Setting this_
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string, optional

The time zone of this script.



**schedule** : dict, optional:

- `scheduled` : boolean  
If the `object` **is** scheduled
- `scheduled_days` : list  
Day based on numeric value starting at 0 **for** Sunday
- `scheduled_minutes` : list  
Minutes of the day it **is** scheduled on
- `scheduled_hours` : list  
Hours of the day it **is** scheduled on
- `scheduled_runs_per_hour` : integer  
Alternative to scheduled minutes, number of times to run per\_↵  
hour

**Returns projects** : list:

A `list` of projects containing the script.

- `id` : integer  
The ID **for** the project.
- `name` : string  
The name of the project.

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

A definition of the parameters this script accepts **in** the arguments\_↵  
field.

- `description` : string  
A short sentence **or** fragment describing this parameter to the\_↵  
end user.
- `type` : string  
The `type` of parameter. Valid options: string, integer, float,\_↵  
bool,  
file, database, credential\_aws, credential\_redshift, **or**  
credential\_custom
- `name` : string  
The variable's `name` as used within your code.
- `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`.
- `value` : string  
The value you would like to `set` this param to. Setting this\_↵  
value makes  
this parameter a fixed param.
- `label` : string  
The label to present to users when asking them **for** the value.
- `required` : boolean  
Whether this param **is** required.

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes,
↪ successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    ↪hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**put\_javascript\_archive** (*id*, *status*)

Update the archive status of this object

**Parameters** **id** : integer

The ID of the object.

**status** : boolean

The desired archived status of the object.

**Returns** **projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```
A definition of the parameters this script accepts in the arguments.
↳field.
- description : string
    A short sentence or fragment describing this parameter to the
↳end user.
```

```

- type : string
    The type of parameter. Valid options: string, integer, float,
    ↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
    ↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

```

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

```

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.

```

```
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**put\_javascript\_projects** (*id, project\_id*)

Add a scripted sql to a project

**Parameters** **id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**put\_javascript\_shares\_groups** (*id, permission\_level, group\_ids*)

Set the permissions groups has on this object

**Parameters** **id** : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**group\_ids** : list

An array of one or more group IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:



```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_javascript\_shares\_users** (*id, permission\_level, user\_ids*)

Set the permissions users have on this object

**Parameters** *id* : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**user\_ids** : list

An array of one or more user IDs

**Returns** *readers* : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_python3** (*id, name, source, \*\*kwargs*)

Replace all attributes of this Python Script

**Parameters** `id` : integer

The ID for the script.

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**params** : list, optional:

```
A definition of the parameters this script accepts in the arguments.  
↪field.  
- description : string  
  A short sentence or fragment describing this parameter to the  
↪end user.  
- type : string  
  The type of parameter. Valid options: string, integer, float,  
↪bool,  
  file, database, credential_aws, credential_redshift, or  
  credential_custom  
- name : string  
  The variable's name as used within your code.  
- 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.  
- value : string  
  The value you would like to set this param to. Setting this  
↪value makes  
  this parameter a fixed param.  
- label : string  
  The label to present to users when asking them for the value.  
- required : boolean  
  Whether this param is required.
```

**required\_resources** : dict, optional:

```
- disk_space : number/float  
  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.  
- cpu : integer  
  The number of CPU shares to allocate for the container. Each  
↪core has  
  1024 shares. Must be at least 2 shares.  
- memory : integer  
  The amount of RAM to allocate for the container (in MiB). Must  
↪be at  
  least 4 MiB.
```

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes,
    ↪ successfully.
```

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per,
    ↪ hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
```

```
The ID for the project.
- name : string
  The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments.
↳field.
- description : string
  A short sentence or fragment describing this parameter to the
↳end user.
- type : string
  The type of parameter. Valid options: string, integer, float,
↳bool,
  file, database, credential_aws, credential_redshift, or
  credential_custom
- name : string
  The variable's name as used within your code.
- 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.
- value : string
  The value you would like to set this param to. Setting this
↳value makes
  this parameter a fixed param.
- label : string
  The label to present to users when asking them for the value.
- required : boolean
  Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes,
    successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
↪core has
    1024 shares. Must be at least 2 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB). Must
↪be at
    least 4 MiB.
```

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**put\_python3\_archive** (*id*, *status*)

Update the archive status of this object

**Parameters** **id** : integer

The ID of the object.

**status** : boolean

The desired archived status of the object.

**Returns** **projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float, _
↪bool,
    file, database, credential_aws, credential_redshift, or
```

```
credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this_
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
```



```
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↳successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↳hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
    ↳ core has 1024 shares. Must be at least 2 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB). Must
    ↳ be at least 4 MiB.
```

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**running\_as** : dict:

```

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

```

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**put\_python3\_projects** (*id*, *project\_id*)

Add a python docker to a project

**Parameters** *id* : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**put\_python3\_shares\_groups** (*id*, *permission\_level*, *group\_ids*)

Set the permissions groups has on this object

**Parameters** *id* : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**group\_ids** : list

An array of one or more group IDs

**Returns** **readers** : dict:

```

- users : list::
    - id : integer
    - name : string
- groups : list::
    - id : integer
    - name : string

```

**owners** : dict:

```

- users : list::
    - id : integer
    - name : string
- groups : list::
    - id : integer
    - name : string

```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_python3\_shares\_users** (*id*, *permission\_level*, *user\_ids*)

Set the permissions users have on this object

**Parameters** **id** : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**user\_ids** : list

An array of one or more user IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```

- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string

```

**put\_r** (*id*, *name*, *source*, *\*\*kwargs*)

Replace all attributes of this R Script

**Parameters** *id* : integer

The ID for the script.

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**params** : list, optional:

A definition of the parameters this script accepts **in** the arguments\_↵  
↵field.

```

- description : string
  A short sentence or fragment describing this parameter to the_↵
  ↵end user.
- type : string
  The type of parameter. Valid options: string, integer, float,_↵
  ↵bool,
  file, database, credential_aws, credential_redshift, or
  credential_custom
- name : string
  The variable's name as used within your code.
- 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.
- value : string
  The value you would like to set this param to. Setting this_↵
  ↵value makes
  this parameter a fixed param.
- label : string
  The label to present to users when asking them for the value.
- required : boolean
  Whether this param is required.

```

**required\_resources** : dict, optional:

```

- disk_space : number/float
  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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each_
    ↪core has
      1024 shares. Must be at least 2 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB). Must_
    ↪be at
      least 4 MiB.
```

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
    ↪successfully.
```

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
```

```

    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per
    hour

```

**Returns projects :** list:

```

A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.

```

**params :** list:

```

A definition of the parameters this script accepts in the arguments
field.
- description : string
    A short sentence or fragment describing this parameter to the
    end user.
- type : string
    The type of parameter. Valid options: string, integer, float,
    bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
    value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**is\_template :** boolean

Whether others scripts use this one as a template.

**type :** string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author :** dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:



```

- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour

```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**required\_resources** : dict:

```

- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each_
↪core has
    1024 shares. Must be at least 2 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB). Must_
↪be at
    least 4 MiB.

```

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.  
The object can still be queried directly by ID

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**put\_r\_archive** (*id*, *status*)

Update the archive status of this object

**Parameters** **id** : integer

The ID of the object.

**status** : boolean

The desired archived status of the object.

**Returns** **projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
```

```
- name : string
    The name of the project.
```

**params** : list:

A definition of the parameters this script accepts **in** the arguments\_↵  
↵field.

```
- description : string
    A short sentence or fragment describing this parameter to the_↵
    ↵end user.
- type : string
    The type of parameter. Valid options: string, integer, float,_↵
    ↵bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this_↵
    ↵value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.
```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**author** : dict:

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

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**required\_resources** : dict:

```
- disk_space : number/float
    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.
- cpu : integer
    The number of CPU shares to allocate for the container. Each
    ↳ core has 1024 shares. Must be at least 2 shares.
- memory : integer
    The amount of RAM to allocate for the container (in MiB). Must
    ↳ be at least 4 MiB.
```

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**source** : string

The body/text of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**put\_r\_projects** (*id*, *project\_id*)

Add a r docker to a project

**Parameters** **id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**put\_r\_shares\_groups** (*id*, *permission\_level*, *group\_ids*)

Set the permissions groups has on this object

**Parameters** **id** : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**group\_ids** : list

An array of one or more group IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_r\_shares\_users** (*id*, *permission\_level*, *user\_ids*)

Set the permissions users have on this object

**Parameters** *id* : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**user\_ids** : list

An array of one or more user IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_sql** (*id*, *remote\_host\_id*, *sql*, *name*, *credential\_id*, *\*\*kwargs*)  
Replace all attributes of this SQL script

**Parameters** **id** : integer

The ID for the script.

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**sql** : string

The raw SQL query for the script.

**name** : string

The name of the script.

**credential\_id** : integer

The credential that this script will use.

**params** : list, optional:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
  A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
  The type of parameter. Valid options: string, integer, float, _
↪bool,
  file, database, credential_aws, credential_redshift, or
  credential_custom
- name : string
  The variable's name as used within your code.
```



```

- 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.
- value : string
    The value you would like to set this param to. Setting this
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**next\_run\_at** : string/time, optional

The time of the next scheduled run.

**csv\_settings** : dict, optional:

```

- include_header : boolean
    Whether or not to include headers in the output data. Default:
↪true
- filename_prefix : string
    A user specified filename prefix for the output file to have.
↪Default:
    null
- unquoted : boolean
    Whether or not to quote fields. Default: false
- column_delimiter : string
    Which delimiter to use, one of "comma", "tab", or "pipe".
↪Default:
    comma
- compression : string
    The type of compression to use, if any, one of "none", "zip", or
    "gzip". Default: gzip
- force_multifile : boolean
    Whether or not the csv should be split into multiple files.
↪Default:
    false

```

**user\_context** : string, optional

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer, optional

The ID of the parent job that will trigger this script

**notifications** : dict, optional:

```

- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.

```

```
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**target\_project\_id** : integer, optional

Target project to which script outputs will be added.

**arguments** : dict, optional

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**time\_zone** : string, optional

The time zone of this script.

**schedule** : dict, optional:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**Returns projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```
A definition of the parameters this script accepts in the arguments_
↪field.
- description : string
    A short sentence or fragment describing this parameter to the_
↪end user.
- type : string
    The type of parameter. Valid options: string, integer, float, _
↪bool,
```

```

    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**sql** : string

The raw SQL query for the script.

**author** : dict:

```

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

```

**csv\_settings** : dict:

```

- include_header : boolean
    Whether or not to include headers in the output data. Default:
↪true
- filename_prefix : string
    A user specified filename prefix for the output file to have.
↪Default:
    null
- unquoted : boolean
    Whether or not to quote fields. Default: false
- column_delimiter : string
    Which delimiter to use, one of "comma", "tab", or "pipe".
↪Default:

```

```
    comma
-   compression : string
    The type of compression to use, if any, one of "none", "zip", or
    "gzip". Default: gzip
-   force_multifile : boolean
    Whether or not the csv should be split into multiple files.
↪Default:
    false
```

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
-   success_on : boolean
    If success email notifications are on
-   stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
-   failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
-   success_email_subject : string
    Custom subject line for success e-mail.
-   failure_on : boolean
    If failure email notifications are on
-   urls : list
    URLs to receive a POST request at job completion
-   success_email_body : string
    Custom body text for success e-mail, written in Markdown.
-   success_email_addresses : list
    Addresses to notify by e-mail when the job completes_
↪successfully.
```

**last\_run** : dict:

```
-   id : integer
-   error : string
    The error message for this run, if present.
-   started_at : string/time
    The time that the run started.
-   created_at : string/time
    The time that the run was queued.
-   state : string
-   finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.

**schedule** : dict:

```

- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour

```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**code\_preview** : string

The code that this script will run with arguments inserted.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```

- 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.

```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**expanded\_arguments** : dict

Expanded arguments for use in injecting into different environments.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**put\_sql\_archive** (*id*, *status*)

Update the archive status of this object

**Parameters** **id** : integer

The ID of the object.

**status** : boolean

The desired archived status of the object.

**Returns** **projects** : list:

```
A list of projects containing the script.
- id : integer
    The ID for the project.
- name : string
    The name of the project.
```

**remote\_host\_id** : integer

The remote host ID that this script will connect to.

**params** : list:

```
A definition of the parameters this script accepts in the arguments.
↳field.
- description : string
    A short sentence or fragment describing this parameter to the
↳end user.
```

```

- type : string
    The type of parameter. Valid options: string, integer, float,
    ↪bool,
    file, database, credential_aws, credential_redshift, or
    credential_custom
- name : string
    The variable's name as used within your code.
- 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.
- value : string
    The value you would like to set this param to. Setting this
    ↪value makes
    this parameter a fixed param.
- label : string
    The label to present to users when asking them for the value.
- required : boolean
    Whether this param is required.

```

**is\_template** : boolean

Whether others scripts use this one as a template.

**type** : string

The type of the script (e.g SQL, Container, Python, R, JavaScript)

**sql** : string

The raw SQL query for the script.

**author** : dict:

```

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

```

**csv\_settings** : dict:

```

- include_header : boolean
    Whether or not to include headers in the output data. Default:
    ↪true
- filename_prefix : string
    A user specified filename prefix for the output file to have.
    ↪Default:
    null
- unquoted : boolean
    Whether or not to quote fields. Default: false

```

```
- column_delimiter : string
    Which delimiter to use, one of "comma", "tab", or "pipe".
↪Default:
    comma
- compression : string
    The type of compression to use, if any, one of "none", "zip", or
    "gzip". Default: gzip
- force_multifile : boolean
    Whether or not the csv should be split into multiple files.
↪Default:
    false
```

**user\_context** : string

“runner” or “author”, who to execute the script as when run as a template.

**parent\_id** : integer

The ID of the parent job that will trigger this script

**notifications** : dict:

```
- success_on : boolean
    If success email notifications are on
- stall_warning_minutes : integer
    Stall warning emails will be sent after this amount of minutes.
- failure_email_addresses : list
    Addresses to notify by e-mail when the job fails.
- success_email_subject : string
    Custom subject line for success e-mail.
- failure_on : boolean
    If failure email notifications are on
- urls : list
    URLs to receive a POST request at job completion
- success_email_body : string
    Custom body text for success e-mail, written in Markdown.
- success_email_addresses : list
    Addresses to notify by e-mail when the job completes
↪successfully.
```

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.
```

**template\_script\_name** : string

The name of the template script.

**next\_run\_at** : string/time

The time of the next scheduled run.



**schedule** : dict:

```
- scheduled : boolean
    If the object is scheduled
- scheduled_days : list
    Day based on numeric value starting at 0 for Sunday
- scheduled_minutes : list
    Minutes of the day it is scheduled on
- scheduled_hours : list
    Hours of the day it is scheduled on
- scheduled_runs_per_hour : integer
    Alternative to scheduled minutes, number of times to run per_
↪hour
```

**created\_at** : string/time

The time this script was created.

**finished\_at** : string/time

The time that the script's last run finished.

**arguments** : dict

Dictionary of name/value pairs to use to run this script. Only settable if this script has defined params.

**code\_preview** : string

The code that this script will run with arguments inserted.

**time\_zone** : string

The time zone of this script.

**updated\_at** : string/time

The time the script was last updated.

**state** : string

The status of the script's last run.

**target\_project\_id** : integer

Target project to which script outputs will be added.

**published\_as\_template\_id** : integer

The ID of the template that this script is backing.

**links** : dict:

```
- 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.
```

**template\_dependents\_count** : integer

How many other scripts use this one as a template.

**id** : integer

The ID for the script.

**expanded\_arguments** : dict

Expanded arguments for use in injecting into different environments.

**archived** : string

The archival status of the requested object(s).

**name** : string

The name of the script.

**hidden** : boolean

The hidden status of the object. Setting this to true hides it from most API endpoints.

The object can still be queried directly by ID

**credential\_id** : integer

The credential that this script will use.

**running\_as** : dict:

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

**from\_template\_id** : integer

The ID of the template this script uses, if any.

**put\_sql\_projects** (*id*, *project\_id*)

Add a scripts to a project

**Parameters** **id** : integer

ID of the resource

**project\_id** : integer

The ID of the project

**Returns** None

Response code 204: success

**put\_sql\_shares\_groups** (*id*, *permission\_level*, *group\_ids*)

Set the permissions groups has on this object

**Parameters** **id** : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**group\_ids** : list

An array of one or more group IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**put\_sql\_shares\_users** (*id*, *permission\_level*, *user\_ids*)

Set the permissions users have on this object

**Parameters** *id* : integer

ID of the resource to be shared

**permission\_level** : string

Options are: “read”, “write”, or “manage”

**user\_ids** : list

An array of one or more user IDs

**Returns** **readers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**owners** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

**total\_group\_shares** : integer

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

**total\_user\_shares** : integer

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

**writers** : dict:

```
- users : list::
  - id : integer
  - name : string
- groups : list::
  - id : integer
  - name : string
```

## Tables

**class Tables** (*session, return\_type='civis'*)

### Methods

<code>get(id)</code>	Show basic table info
<code>get_enhancements_cass_ncoa(id, source_table_id)</code>	View the status of a CASS / NCOA table enhancement
<code>get_enhancements_geocodings(id, source_table_id)</code>	View the status of a geocoding table enhancement
<code>get_enhancements_prepared_matchings(id, ...)</code>	View a prepared matching enhancement
<code>get_enhancements_table_matchings(id, ...)</code>	View a table matching enhancement
<code>list(**kwargs)</code>	List tables
<code>list_columns(id, **kwargs)</code>	List columns in the specified table
<code>patch(id, **kwargs)</code>	Update a table
<code>post(data, name, database_id, schema)</code>	Import a file into a table
<code>post_enhancements_cass_ncoa(source_table_id, ...)</code>	Standardize addresses in a table
<code>post_enhancements_geocodings(source_table_id)</code>	Geocode a table
<code>post_enhancements_prepared_matchings(...)</code>	Match person records against a dynamo table prepared by Civis
<code>post_enhancements_table_matchings(...)</code>	Match person records against an arbitrary Redshift table
<code>post_refresh(id)</code>	Request a refresh for column and table statistics

**get** (*id*)

Show basic table info

**Parameters** *id* : integer**Returns** *description* : string

The description of the table, as specified by the table owner

**last\_refresh** : string/date-time

The time of the last statistics refresh.

**distkey** : string

The column used as the Amazon Redshift distkey.

**database\_id** : integer

The ID of the database.

**refresh\_status** : 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.

**sortkeys** : string

The column used as the Amazon Redshift sortkey.

**multipart\_key** : list**column\_count** : integer

The number of columns in the table.

**ontology\_mapping** : dict

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

**view\_def** : string**enhancements** : list:

```
- created_at : string/time
- join_id : integer
- type : string
- updated_at : string/time
```

**id** : integer

The ID of the table.

**columns** : list:

```
- description : string
    The description of the column, as specified by the table owner
- encoding : string
    The compression encoding for this columnSee: http://docs.aws.amazon.com/redshift/latest/dg/c\_Compression\_encodings.html
- useable_as_primary_key : boolean
    Whether the column may be used as an primary key to identify
    table rows.
```

```
- order : integer
    Relative position of the column in the table.
- min_value : string
    Smallest value in the column.
- max_value : string
    Largest value in the column.
- possible_dependent_variable_types : list
    Possible dependent variable types the column may be used to
    ↪model.
    Null if it may not be used as a dependent variable.
- null_count : integer
    Number of null values in the column.
- sql_type : string
    SQL type of the column.
- distinct_count : integer
    Number of distinct values in the column.
- sample_values : list
    A sample of values from the column.
- value_distribution : dict
    An object mapping distinct values in the column to the number
    ↪of times
    they appear in the column
- name : string
    Name of the column.
- coverage_count : integer
    Number of non-null values in the column.
- useable_as_independent_variable : boolean
    Whether the column may be used as an independent variable to
    ↪train a
    model.
- stddev : number/float
    Stddev of the column, where applicable.
- value_distribution_percent : dict
    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.
- avg_value : number/float
    Average value of the column, where applicable.
```

**schema** : string

The name of the schema containing the table.

**row\_count** : integer

The number of rows in the table.

**last\_run** : dict:

```
- id : integer
- error : string
    The error message for this run, if present.
- started_at : string/time
    The time that the run started.
- created_at : string/time
    The time that the run was queued.
```

```
- state : string
- finished_at : string/time
    The time that the run completed.
```

**is\_view** : boolean

True if this table represents a view. False if it represents a regular table.

**refresh\_id** : string

The ID of the most recent statistics refresh.

**outgoing\_table\_matches** : list:

```
- target : dict::
    - name : string
- job : dict::
    - id : integer
    - updated_at : string/date-time
    - match_options : dict::
        - max_matches : integer
        - threshold : string
    - runs : list::
        Information about the most recent runs of the job.
        - id : integer
        - error : string
            The error message for this run, if present.
        - started_at : string/time
            The time that the run started.
        - created_at : string/time
            The time that the run was queued.
        - state : string
        - finished_at : string/time
            The time that the run completed.
    - type : string
    - name : string
    - hidden : boolean
        The hidden status of the object. Setting this to true hides_
↳it from
    most API endpoints. The object can still be queried_
↳directly by ID
    - created_at : string/date-time
    - state : string
        Whether the job is idle, queued, running, cancelled, or_
↳failed.
    - last_run : dict::
        - id : integer
        - error : string
            The error message for this run, if present.
        - started_at : string/time
            The time that the run started.
        - created_at : string/time
            The time that the run was queued.
        - state : string
        - finished_at : string/time
            The time that the run completed.
- source_table_id : integer
    Source table
- target_id : integer
```

```
Target ID
- target_type : string
  Target type
```

**joins** : list:

```
- id : integer
- updated_at : string/time
- on : string
- right_identfier : string
- left_join : boolean
- left_identfier : string
- created_at : string/time
- right_table_id : integer
- left_table_id : integer
```

**size\_mb** : number/float

The size of the table in megabytes.

**owner** : string

The database username of the table's owner.

**name** : string

Name of the table.

**get\_enhancements\_cass\_ncoa** (*id*, *source\_table\_id*)

View the status of a CASS / NCOA table enhancement

**Parameters** **id** : integer

The ID of the enhancement.

**source\_table\_id** : integer

The ID of the table that was enhanced.

**Returns** **id** : integer

The ID of the enhancement.

**ncoa\_credential\_id** : integer

Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

**perform\_ncoa** : boolean

Whether to update addresses for records matching the National Change of Address (NCOA) database.

**enhanced\_table\_name** : string

The name of the table created by the enhancement.

**output\_level** : 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.

**state** : string

The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.



**enhanced\_table\_schema** : string

The schema name of the table created by the enhancement.

**source\_table\_id** : integer

The ID of the table that was enhanced.

**get\_enhancements\_geocodings** (*id*, *source\_table\_id*)

View the status of a geocoding table enhancement

**Parameters id** : integer

The ID of the enhancement.

**source\_table\_id** : integer

The ID of the table that was enhanced.

**Returns id** : integer

The ID of the enhancement.

**enhanced\_table\_name** : string

The name of the table created by the enhancement.

**state** : string

The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**source\_table\_id** : integer

The ID of the table that was enhanced.

**enhanced\_table\_schema** : string

The schema name of the table created by the enhancement.

**get\_enhancements\_prepared\_matchings** (*id*, *source\_table\_id*)

View a prepared matching enhancement

**Parameters id** : integer

The ID of the enhancement.

**source\_table\_id** : integer

The ID of the table that was enhanced.

**Returns id** : integer

The ID of the enhancement.

**threshold** : number/float

The confidence threshold which must be met for two individuals to be declared a match. Must be less than or equal to 1 and greater than or equal to 0.

**match\_table\_id** : integer

The ID of the Dynamo table to match against.

**enhanced\_table\_name** : string

The name of the table created by the enhancement.

**max\_matches** : integer

The maximum number of individuals a person may be matched with. A value of 0 indicates that all matches should be returned.

**state** : string

The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**enhanced\_table\_schema** : string

The schema name of the table created by the enhancement.

**source\_table\_id** : integer

The ID of the table that was enhanced.

**get\_enhancements\_table\_matchings** (*id*, *source\_table\_id*)

View a table matching enhancement

**Parameters** **id** : integer

The ID of the enhancement.

**source\_table\_id** : integer

The ID of the table that was enhanced.

**Returns** **id** : integer

The ID of the enhancement.

**threshold** : number/float

The confidence threshold which must be met for two individuals to be declared a match. Must be less than or equal to 1 and greater than or equal to 0.

**match\_table\_id** : integer

The ID of the Redshift table to match against.

**enhanced\_table\_name** : string

The name of the table created by the enhancement.

**max\_matches** : integer

The maximum number of individuals a person may be matched with. A value of 0 indicates that all matches should be returned.

**state** : string

The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**enhanced\_table\_schema** : string

The schema name of the table created by the enhancement.

**source\_table\_id** : integer

The ID of the table that was enhanced.

**list** (*\*\*kwargs*)

List tables

**Parameters** **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.

**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.

**order\_dir** : string, optional

Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns description** : string

The description of the table, as specified by the table owner

**refresh\_id** : string

The ID of the most recent statistics refresh.

**last\_refresh** : string/date-time

The time of the last statistics refresh.

**schema** : string

The name of the schema containing the table.

**distkey** : string

The column used as the Amazon Redshift distkey.

**row\_count** : integer

The number of rows in the table.

**database\_id** : integer

The ID of the database.

**refresh\_status** : 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.

**is\_view** : boolean

True if this table represents a view. False if it represents a regular table.

**id** : integer

The ID of the table.

**size\_mb** : number/float

The size of the table in megabytes.

**sortkeys** : string

The column used as the Amazon Redshift sortkey.

**name** : string

Name of the table.

**column\_count** : integer

The number of columns in the table.

**last\_run** : dict:

```
- id : integer
- error : string
  The error message for this run, if present.
- started_at : string/time
  The time that the run started.
- created_at : string/time
  The time that the run was queued.
- state : string
- finished_at : string/time
  The time that the run completed.
```

**owner** : string

The database username of the table's owner.

**list\_columns** (*id*, *\*\*kwargs*)

List columns in the specified table

**Parameters** *id* : integer

**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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns description** : string

The description of the column, as specified by the table owner

**encoding** : string

The compression encoding for this columnSee: [http://docs.aws.amazon.com/redshift/latest/dg/c\\_Compression\\_encodings.html](http://docs.aws.amazon.com/redshift/latest/dg/c_Compression_encodings.html)

**useable\_as\_primary\_key** : boolean

Whether the column may be used as an primary key to identify table rows.

**order** : integer

Relative position of the column in the table.

**min\_value** : string

Smallest value in the column.

**max\_value** : string

Largest value in the column.

**possible\_dependent\_variable\_types** : list

Possible dependent variable types the column may be used to model. Null if it may not be used as a dependent variable.

**null\_count** : integer

Number of null values in the column.

**sql\_type** : string

SQL type of the column.

**distinct\_count** : integer

Number of distinct values in the column.

**sample\_values** : list

A sample of values from the column.

**value\_distribution** : dict

An object mapping distinct values in the column to the number of times they appear in the column

**name** : string

Name of the column.

**coverage\_count** : integer

Number of non-null values in the column.

**useable\_as\_independent\_variable** : boolean

Whether the column may be used as an independent variable to train a model.

**stddev** : number/float

Stddev of the column, where applicable.

**value\_distribution\_percent** : dict

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.

**avg\_value** : number/float

Average value of the column, where applicable.

**patch** (*id*, *\*\*kwargs*)

Update a table

**Parameters** **id** : integer

The ID of the table.

**description** : string, optional

The user-defined description of the table.

**ontology\_mapping** : dict, optional

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

**Returns** **description** : string

The description of the table, as specified by the table owner

**refresh\_id** : string

The ID of the most recent statistics refresh.

**last\_refresh** : string/date-time

The time of the last statistics refresh.

**schema** : string

The name of the schema containing the table.

**distkey** : string

The column used as the Amazon Redshift distkey.

**row\_count** : integer

The number of rows in the table.

**database\_id** : integer

The ID of the database.

**refresh\_status** : 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.

**is\_view** : boolean

True if this table represents a view. False if it represents a regular table.

**id** : integer

The ID of the table.

**size\_mb** : number/float

The size of the table in megabytes.

**sortkeys** : string

The column used as the Amazon Redshift sortkey.

**name** : string

Name of the table.

**column\_count** : integer

The number of columns in the table.

**last\_run** : dict:

```
- id : integer
- error : string
  The error message for this run, if present.
- started_at : string/time
  The time that the run started.
- created_at : string/time
  The time that the run was queued.
- state : string
- finished_at : string/time
  The time that the run completed.
```

**ontology\_mapping** : dict

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

**owner** : string

The database username of the table's owner.

**post** (*data*, *name*, *database\_id*, *schema*)

Import a file into a table

**Parameters** **data** : string

The file to import, uploaded using HTTP multipart.

**name** : string

The destination table name, without the schema prefix.

**database\_id** : integer

The ID of the destination database.

**schema** : string

The destination schema name.

**Returns** **name** : string

The destination table name, without the schema prefix.

**schema** : string

The destination schema name.

**started\_at** : string/date-time

The start time of the last run.

**state** : string

The state of the last run.

**database\_id** : integer

The ID of the destination database.

**finished\_at** : string/date-time

The end time of the last run.

**post\_enhancements\_cass\_ncoa** (*source\_table\_id*, *\*\*kwargs*)

Standardize addresses in a table

**Parameters** **source\_table\_id** : integer

The ID of the table to be enhanced.

**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.

**perform\_ncoa** : boolean, optional

Whether to update addresses for records matching the National Change of Address (NCOA) database.

**Returns** **id** : integer

The ID of the enhancement.

**ncoa\_credential\_id** : integer

Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

**perform\_ncoa** : boolean

Whether to update addresses for records matching the National Change of Address (NCOA) database.

**enhanced\_table\_name** : string

The name of the table created by the enhancement.

**output\_level** : 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.

**state** : string

The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**enhanced\_table\_schema** : string

The schema name of the table created by the enhancement.

**source\_table\_id** : integer



The ID of the table that was enhanced.

**post\_enhancements\_geocodings** (*source\_table\_id*)

Geocode a table

**Parameters** **source\_table\_id** : integer

The ID of the table to be enhanced.

**Returns** **id** : integer

The ID of the enhancement.

**enhanced\_table\_name** : string

The name of the table created by the enhancement.

**state** : string

The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**source\_table\_id** : integer

The ID of the table that was enhanced.

**enhanced\_table\_schema** : string

The schema name of the table created by the enhancement.

**post\_enhancements\_prepared\_matchings** (*source\_table\_id*, *match\_table\_id*, *threshold*,  
\*\**kwargs*)

Match person records against a dynamo table prepared by Civis

**Parameters** **source\_table\_id** : integer

The ID of the table to be enhanced.

**match\_table\_id** : integer

The ID of the Dynamo table to match against.

**threshold** : number/float

The confidence threshold which must be met for two individuals to be declared a match. Must be less than or equal to 1 and greater than or equal to 0.

**max\_matches** : integer, optional

The maximum number of individuals a person may be matched with. A value of 0 indicates that all matches should be returned.

**Returns** **id** : integer

The ID of the enhancement.

**threshold** : number/float

The confidence threshold which must be met for two individuals to be declared a match. Must be less than or equal to 1 and greater than or equal to 0.

**match\_table\_id** : integer

The ID of the Dynamo table to match against.

**enhanced\_table\_name** : string

The name of the table created by the enhancement.

**max\_matches** : integer

The maximum number of individuals a person may be matched with. A value of 0 indicates that all matches should be returned.

**state** : string

The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**enhanced\_table\_schema** : string

The schema name of the table created by the enhancement.

**source\_table\_id** : integer

The ID of the table that was enhanced.

**post\_enhancements\_table\_matchings** (*source\_table\_id*, *match\_table\_id*, *threshold*,  
\*\**kwargs*)

Match person records against an arbitrary Redshift table

**Parameters** **source\_table\_id** : integer

The ID of the table to be enhanced.

**match\_table\_id** : integer

The ID of the Redshift table to match against.

**threshold** : number/float

The confidence threshold which must be met for two individuals to be declared a match. Must be less than or equal to 1 and greater than or equal to 0.

**max\_matches** : integer, optional

The maximum number of individuals a person may be matched with. A value of 0 indicates that all matches should be returned.

**Returns** **id** : integer

The ID of the enhancement.

**threshold** : number/float

The confidence threshold which must be met for two individuals to be declared a match. Must be less than or equal to 1 and greater than or equal to 0.

**match\_table\_id** : integer

The ID of the Redshift table to match against.

**enhanced\_table\_name** : string

The name of the table created by the enhancement.

**max\_matches** : integer

The maximum number of individuals a person may be matched with. A value of 0 indicates that all matches should be returned.

**state** : string

The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**enhanced\_table\_schema** : string

The schema name of the table created by the enhancement.

**source\_table\_id** : integer

The ID of the table that was enhanced.

**post\_refresh** (*id*)

Request a refresh for column and table statistics

**Parameters** **id** : integer

**Returns** **description** : string

The description of the table, as specified by the table owner

**last\_refresh** : string/date-time

The time of the last statistics refresh.

**distkey** : string

The column used as the Amazon Redshift distkey.

**database\_id** : integer

The ID of the database.

**refresh\_status** : 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.

**sortkeys** : string

The column used as the Amazon Redshift sortkey.

**multipart\_key** : list

**column\_count** : integer

The number of columns in the table.

**ontology\_mapping** : dict

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

**view\_def** : string

**enhancements** : list:

```
- created_at : string/time
- join_id : integer
- type : string
- updated_at : string/time
```

**id** : integer

The ID of the table.

**columns** : list:

```
- description : string
    The description of the column, as specified by the table owner
- encoding : string
    The compression encoding for this columnSee: http://docs.aws.amazon.com/redshift/latest/dg/c\_Compression\_encodings.html
- useable_as_primary_key : boolean
```

```
Whether the column may be used as an primary key to identify
↳table
  rows.
- order : integer
  Relative position of the column in the table.
- min_value : string
  Smallest value in the column.
- max_value : string
  Largest value in the column.
- possible_dependent_variable_types : list
  Possible dependent variable types the column may be used to
↳model.
  Null if it may not be used as a dependent variable.
- null_count : integer
  Number of null values in the column.
- sql_type : string
  SQL type of the column.
- distinct_count : integer
  Number of distinct values in the column.
- sample_values : list
  A sample of values from the column.
- value_distribution : dict
  An object mapping distinct values in the column to the number
↳of times
  they appear in the column
- name : string
  Name of the column.
- coverage_count : integer
  Number of non-null values in the column.
- useable_as_independent_variable : boolean
  Whether the column may be used as an independent variable to
↳train a
  model.
- stddev : number/float
  Stddev of the column, where applicable.
- value_distribution_percent : dict
  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.
- avg_value : number/float
  Average value of the column, where applicable.
```

**schema** : string

The name of the schema containing the table.

**row\_count** : integer

The number of rows in the table.

**last\_run** : dict:

```
- id : integer
- error : string
  The error message for this run, if present.
- started_at : string/time
```

```

    The time that the run started.
- created_at : string/time
    The time that the run was queued.
- state : string
- finished_at : string/time
    The time that the run completed.

```

**is\_view** : boolean

True if this table represents a view. False if it represents a regular table.

**refresh\_id** : string

The ID of the most recent statistics refresh.

**outgoing\_table\_matches** : list:

```

- target : dict::
  - name : string
- job : dict::
  - id : integer
  - updated_at : string/date-time
  - match_options : dict::
    - max_matches : integer
    - threshold : string
  - runs : list::
    Information about the most recent runs of the job.
    - id : integer
    - error : string
      The error message for this run, if present.
    - started_at : string/time
      The time that the run started.
    - created_at : string/time
      The time that the run was queued.
    - state : string
    - finished_at : string/time
      The time that the run completed.
  - type : string
  - name : string
  - hidden : boolean
    The hidden status of the object. Setting this to true hides_
↳it from
    most API endpoints. The object can still be queried_
↳directly by ID
  - created_at : string/date-time
  - state : string
    Whether the job is idle, queued, running, cancelled, or_
↳failed.
  - last_run : dict::
    - id : integer
    - error : string
      The error message for this run, if present.
    - started_at : string/time
      The time that the run started.
    - created_at : string/time
      The time that the run was queued.
    - state : string
    - finished_at : string/time
      The time that the run completed.

```

```
- source_table_id : integer
    Source table
- target_id : integer
    Target ID
- target_type : string
    Target type
```

**joins** : list:

```
- id : integer
- updated_at : string/time
- on : string
- right_identifier : string
- left_join : boolean
- left_identifier : string
- created_at : string/time
- right_table_id : integer
- left_table_id : integer
```

**size\_mb** : number/float

The size of the table in megabytes.

**owner** : string

The database username of the table's owner.

**name** : string

Name of the table.

## Users

**class Users** (*session*, *return\_type*='civis')

### Methods

<code>delete_api_keys(id, key_id)</code>	Revoke the specified API key
<code>get(id)</code>	Show info about a user
<code>get_api_keys(id, key_id)</code>	Show the specified API key
<code>list(**kwargs)</code>	List users
<code>list_api_keys(id, **kwargs)</code>	Show API keys belonging to the specified user
<code>list_me()</code>	Show info about the logged-in user
<code>patch_me(**kwargs)</code>	Update info about the logged-in user
<code>post_api_keys(id, expires_in, name, **kwargs)</code>	Create a new API key belonging to the logged-in user

**delete\_api\_keys** (*id*, *key\_id*)

Revoke the specified API key

**Parameters** **id** : string

The ID of the user or 'me'.

**key\_id** : integer

The ID of the API key.

**Returns** `id` : integer

The ID of the API key.

**expired** : boolean

True if the key has expired.

**use\_count** : integer

The number of times the key has been used.

**last\_used\_at** : string/date-time

The date and time when the key was last used.

**name** : string

The name of the API key.

**revoked\_at** : string/date-time

The date and time when the key was revoked.

**constraints** : list:

```
Constraints on the abilities of the created key
- put_allowed : boolean
    Whether the constraint allows PUT requests.
- head_allowed : boolean
    Whether the constraint allows HEAD requests.
- constraint : string
    The path matcher of the constraint.
- post_allowed : boolean
    Whether the constraint allows POST requests.
- get_allowed : boolean
    Whether the constraint allows GET requests.
- patch_allowed : boolean
    Whether the constraint allows PATCH requests.
- delete_allowed : boolean
    Whether the constraint allows DELETE requests.
- constraint_type : string
    The type of constraint (exact/prefix/regex/verb).
```

**created\_at** : string/date-time

The date and time when the key was created.

**expires\_at** : string/date-time

The date and time when the key expired.

**active** : boolean

True if the key has neither expired nor been revoked.

**scopes** : list

The scopes which the key is permissioned on.

**get** (*id*)

Show info about a user

**Parameters** `id` : integer

The ID of this user.

**Returns** **state** : string

The state of this user.

**otp\_required\_for\_login** : string

The two factor authorization requirement for this user.

**email** : string

The email of this user.

**primary\_group\_id** : integer

The ID of the primary group of this user.

**user** : string

The username of this user.

**phone** : string

The phone number of this user.

**title** : string

The title of this user.

**id** : integer

The ID of this user.

**vpn\_enabled** : string

The availability of vpn for this user.

**department** : string

The deartment of this user.

**name** : string

The name of this user.

**prefers\_sms\_otp** : string

The preference for phone authorization of this user

**github\_username** : string

The GitHub username of this user.

**initials** : string

The initials of this user.

**city** : string

The city of this user.

**active** : string

The account status of this user.

**time\_zone** : string

The time zone of this user.

**groups** : list:



```

An array of all the groups this user is in.
- id : integer
    The ID of this group.
- organization_id : integer
    The organization associated with this group.
- name : string
    The name of this group.

```

**get\_api\_keys** (*id*, *key\_id*)

Show the specified API key

**Parameters** **id** : string

The ID of the user or 'me'.

**key\_id** : integer

The ID of the API key.

**Returns** **id** : integer

The ID of the API key.

**expired** : boolean

True if the key has expired.

**use\_count** : integer

The number of times the key has been used.

**last\_used\_at** : string/date-time

The date and time when the key was last used.

**name** : string

The name of the API key.

**revoked\_at** : string/date-time

The date and time when the key was revoked.

**constraints** : list:

```

Constraints on the abilities of the created key
- put_allowed : boolean
    Whether the constraint allows PUT requests.
- head_allowed : boolean
    Whether the constraint allows HEAD requests.
- constraint : string
    The path matcher of the constraint.
- post_allowed : boolean
    Whether the constraint allows POST requests.
- get_allowed : boolean
    Whether the constraint allows GET requests.
- patch_allowed : boolean
    Whether the constraint allows PATCH requests.
- delete_allowed : boolean
    Whether the constraint allows DELETE requests.
- constraint_type : string
    The type of constraint (exact/prefix/regex/verb).

```

**created\_at** : string/date-time

The date and time when the key was created.

**expires\_at** : string/date-time

The date and time when the key expired.

**active** : boolean

True if the key has neither expired nor been revoked.

**scopes** : list

The scopes which the key is permissioned on.

**list** (*\*\*kwargs*)

List users

**Parameters** **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”.

**query** : string, optional

Return users who match the given query, based on name, user, and email.

**group\_id** : integer, optional

The ID of the group by which to filter users. Cannot be present if organization\_id is.

**organization\_id** : integer, optional

The ID of the organization by which to filter users. Cannot be present if group\_id is.

**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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **id** : integer

The ID of this user.

**name** : string

The name of this user.

**current\_sign\_in\_at** : string/date-time

The date and time when the user’s current session began.

**created\_at** : string/date-time

The date and time when the user was created.

**primary\_group\_id** : integer

The ID of the primary group of this user.

**active** : string

The account status of this user.

**user** : string

The username of this user.

**groups** : list:

An array of **all** the groups this user **is in**.

- **id** : integer  
The ID of this group.
- **organization\_id** : integer  
The organization associated **with** this group.
- **name** : string  
The name of this group.

**email** : string

The email of this user.

**list\_api\_keys** (*id*, *\*\*kwargs*)

Show API keys belonging to the specified user

**Parameters** **id** : string

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.

**iterator** : bool, optional

If True, return a generator to iterate over all responses. Use when more results than the maximum allowed by limit are needed. When True, limit and page\_num are ignored. Defaults to False.

**Returns** **id** : integer

The ID of the API key.

**expired** : boolean

True if the key has expired.

**use\_count** : integer

The number of times the key has been used.

**last\_used\_at** : string/date-time

The date and time when the key was last used.

**name** : string

The name of the API key.

**revoked\_at** : string/date-time

The date and time when the key was revoked.

**created\_at** : string/date-time

The date and time when the key was created.

**expires\_at** : string/date-time

The date and time when the key expired.

**constraint\_count** : integer

The number of constraints on the created key

**active** : boolean

True if the key has neither expired nor been revoked.

**scopes** : list

The scopes which the key is permissioned on.

**list\_me()**

Show info about the logged-in user

**Returns last\_checked\_announcements** : string/date-time

The date and time at which the user last checked their announcements.

**feature\_flags** : dict

The feature flag settings for this user.

**roles** : list

The roles this user has, listed by slug.

**username** : string

This user's username.

**id** : integer

The ID of this user.

**name** : string

This user's name.

**preferences** : dict

This user's preferences.

**custom\_branding** : string

The branding of Platform for this user.

**initials** : string

This user's initials.

**email** : string

This user's email address.

**organization\_name** : string

The name of the organization the user belongs to.

**groups** : list:

An array of **all** the groups this user **is in**.

- **id** : integer  
The ID of this group.
- **organization\_id** : integer  
The organization associated **with** this group.
- **name** : string  
The name of this group.

**patch\_me** (\*\*kwargs)

Update info about the logged-in user

**Parameters last\_checked\_announcements** : string/date-time, optional

The date and time at which the user last checked their announcements.

**preferences** : dict, optional:

- **app\_index\_order\_field** : string  
Order field **for** the apps index pages.
- **import\_index\_type\_filter** : string  
Type **filter for** the imports index page.
- **import\_index\_dest\_filter** : string  
Destination **filter for** the imports index page.
- **model\_index\_archived\_filter** : string  
Archived **filter for** the models index page.
- **script\_index\_author\_filter** : string  
Author **filter for** the scripts index page.
- **script\_index\_order\_field** : string  
Order field **for** the scripts index page.
- **export\_index\_author\_filter** : string  
Author **filter for** the exports index page.
- **model\_index\_status\_filter** : string  
Status **filter for** the models index page.
- **script\_index\_archived\_filter** : string  
Archived **filter for** the scripts index page.
- **import\_index\_status\_filter** : string  
Status **filter for** the imports index page.
- **enhancement\_index\_author\_filter** : string  
Author **filter for** the enhancements index page.
- **import\_index\_order\_dir** : string  
Order direction **for** the imports index page.
- **model\_index\_order\_dir** : string  
Order direction **for** the models index page.
- **model\_index\_order\_field** : string  
Order field **for** the models index page.
- **import\_index\_author\_filter** : string  
Author **filter for** the imports index page.
- **result\_index\_type\_filter** : string

```
    Type filter for the results index page.
- script_index_status_filter : string
    Status filter for the scripts index page.
- app_index_order_dir : string
    Order direction for the apps index pages.
- project_index_archived_filter : string
    Archived filter for the projects index page.
- project_detail_order_dir : string
    Order direction for projects detail pages.
- project_index_order_field : string
    Order field for the projects index page.
- preferred_server_id : integer
    ID of preferred server.
- enhancement_index_order_dir : string
    Order direction for the enhancements index page.
- import_index_order_field : string
    Order field for the imports index page.
- import_index_archived_filter : string
    Archived filter for the imports index page.
- result_index_author_filter : string
    Author filter for the results index page.
- model_index_thumbnail_view : string
    Thumbnail view for the models index page.
- export_index_status_filter : string
    Status filter for the exports index page.
- script_index_type_filter : string
    Type filter for the scripts index page.
- result_index_archived_filter : string
    Archived filter for the results index page.
- project_index_author_filter : string
    Author filter for the projects index page.
- enhancement_index_archived_filter : string
    Archived filter for the enhancements index page.
- project_index_order_dir : string
    Order direction for the projects index page.
- result_index_order_dir : string
    Order direction for the results index page.
- script_index_order_dir : string
    Order direction for the scripts index page.
- project_detail_type_filter : string
    Type filter for projects detail pages.
- project_detail_archived_filter : string
    Archived filter for the projects detail pages.
- report_index_thumbnail_view : string
    Thumbnail view for the reports index page.
- export_index_type_filter : string
    Type filter for the exports index page.
- enhancement_index_order_field : string
    Order field for the enhancements index page.
- project_detail_order_field : string
    Order field for projects detail pages.
- model_index_author_filter : string
    Author filter for the models index page.
- civis_explore_skip_intro : boolean
    Whether the user is shown steps for each exploration.
- project_detail_author_filter : string
    Author filter for projects detail pages.
- export_index_order_dir : string
```

```

    Order direction for the exports index page.
- result_index_order_field : string
    Order field for the results index page.
- export_index_order_field : string
    Order field for the exports index page.

```

**Returns** `last_checked_announcements` : string/date-time

The date and time at which the user last checked their announcements.

**feature\_flags** : dict

The feature flag settings for this user.

**roles** : list

The roles this user has, listed by slug.

**username** : string

This user's username.

**id** : integer

The ID of this user.

**name** : string

This user's name.

**preferences** : dict

This user's preferences.

**custom\_branding** : string

The branding of Platform for this user.

**initials** : string

This user's initials.

**email** : string

This user's email address.

**organization\_name** : string

The name of the organization the user belongs to.

**groups** : list:

```

An array of all the groups this user is in.
- id : integer
    The ID of this group.
- organization_id : integer
    The organization associated with this group.
- name : string
    The name of this group.

```

**post\_api\_keys** (*id, expires\_in, name, \*\*kwargs*)

Create a new API key belonging to the logged-in user

**Parameters** `id` : string

The ID of the user or 'me'.

**expires\_in** : integer

The number of seconds the key should last for.

**name** : string

The name of the API key.

**constraints** : list, optional:

```
Constraints on the abilities of the created key.
- put_allowed : boolean
    Whether the constraint allows PUT requests.
- head_allowed : boolean
    Whether the constraint allows HEAD requests.
- constraint : string
    The path matcher of the constraint.
- post_allowed : boolean
    Whether the constraint allows POST requests.
- get_allowed : boolean
    Whether the constraint allows GET requests.
- patch_allowed : boolean
    Whether the constraint allows PATCH requests.
- delete_allowed : boolean
    Whether the constraint allows DELETE requests.
- constraint_type : string
    The type of constraint (exact/prefix/regex/verb).
```

**Returns** **expired** : boolean

True if the key has expired.

**use\_count** : integer

The number of times the key has been used.

**expires\_at** : string/date-time

The date and time when the key expired.

**constraints** : list:

```
Constraints on the abilities of the created key
- put_allowed : boolean
    Whether the constraint allows PUT requests.
- head_allowed : boolean
    Whether the constraint allows HEAD requests.
- constraint : string
    The path matcher of the constraint.
- post_allowed : boolean
    Whether the constraint allows POST requests.
- get_allowed : boolean
    Whether the constraint allows GET requests.
- patch_allowed : boolean
    Whether the constraint allows PATCH requests.
- delete_allowed : boolean
    Whether the constraint allows DELETE requests.
- constraint_type : string
    The type of constraint (exact/prefix/regex/verb).
```

**id** : integer

The ID of the API key.



**token** : string

The API key.

**last\_used\_at** : string/date-time

The date and time when the key was last used.

**name** : string

The name of the API key.

**revoked\_at** : string/date-time

The date and time when the key was revoked.

**created\_at** : string/date-time

The date and time when the key was created.

**active** : boolean

True if the key has neither expired nor been revoked.

**scopes** : list

The scopes which the key is permissioned on.

## Command Line Interface

A command line interface (CLI) to Civis is provided. This can be invoked by typing the command `civis` in the shell (sh, bash, zsh, etc.). It can also be used in Civis container scripts where the Docker image has this client installed. Here's a simple example of printing the types of scripts.

```
> civis scripts list-types
- name: sql
- name: python3
- name: javascript
- name: r
- name: containers
```

Not all API endpoints are available through the CLI since some take complex data types (e.g., arrays, objects/dictionaries) as input. However, functionality is available for getting information about scripts, logs, etc., as well as executing already created scripts.

There are a few extra, CLI-only commands that wrap the Files API endpoints to make uploading and downloading files easier: `civis files upload $PATH` and `civis files download $FILEID $PATH`.

The default output format is YAML, but the `--json-output` allows you to get output in JSON.



## CHAPTER 5

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



## A

APIClient (class in civis), 22

## C

CIVIS\_API\_KEY, 12, 14, 15, 17, 19–22

civis\_to\_csv() (in module civis.io), 11

civis\_to\_file() (in module civis.io), 19

Credentials (class in civis.resources.\_resources), 26

csv\_to\_civis() (in module civis.io), 12

## D

Databases (class in civis.resources.\_resources), 32

dataframe\_to\_civis() (in module civis.io), 14

default\_credential (civis.APIClient attribute), 23

delete\_api\_keys() (civis.resources.\_resources.Users method), 394

delete\_builds() (civis.resources.\_resources.Models method), 81

delete\_containers\_projects() (civis.resources.\_resources.Scripts method), 167

delete\_containers\_runs() (civis.resources.\_resources.Scripts method), 167

delete\_containers\_shares\_groups() (civis.resources.\_resources.Scripts method), 167

delete\_containers\_shares\_users() (civis.resources.\_resources.Scripts method), 168

delete\_custom\_projects() (civis.resources.\_resources.Scripts method), 168

delete\_custom\_runs() (civis.resources.\_resources.Scripts method), 168

delete\_custom\_shares\_groups() (civis.resources.\_resources.Scripts method), 168

delete\_custom\_shares\_users() (civis.resources.\_resources.Scripts method),

168

delete\_files\_runs() (civis.resources.\_resources.Imports method), 41

delete\_grants() (civis.resources.\_resources.Reports method), 143

delete\_javascript\_projects() (civis.resources.\_resources.Scripts method), 169

delete\_javascript\_runs() (civis.resources.\_resources.Scripts method), 169

delete\_javascript\_shares\_groups() (civis.resources.\_resources.Scripts method), 169

delete\_javascript\_shares\_users() (civis.resources.\_resources.Scripts method), 169

delete\_projects() (civis.resources.\_resources.Files method), 35

delete\_projects() (civis.resources.\_resources.Imports method), 41

delete\_projects() (civis.resources.\_resources.Jobs method), 72

delete\_projects() (civis.resources.\_resources.Models method), 81

delete\_projects() (civis.resources.\_resources.Reports method), 143

delete\_python3\_projects() (civis.resources.\_resources.Scripts method), 170

delete\_python3\_runs() (civis.resources.\_resources.Scripts method), 170

delete\_python3\_shares\_groups() (civis.resources.\_resources.Scripts method), 170

delete\_python3\_shares\_users() (civis.resources.\_resources.Scripts method), 170

delete\_r\_projects() (civis.resources.\_resources.Scripts method), 170

delete\_r\_runs() (civis.resources.\_resources.Scripts

method), 171  
delete\_r\_shares\_groups()  
    (civis.resources.\_resources.Scripts method),  
    171  
delete\_r\_shares\_users() (civis.resources.\_resources.Scripts  
    method), 171  
delete\_runs() (civis.resources.\_resources.Predictions  
    method), 110  
delete\_runs() (civis.resources.\_resources.Queries  
    method), 134  
delete\_shares\_groups() (civis.resources.\_resources.Files  
    method), 35  
delete\_shares\_groups() (civis.resources.\_resources.Imports  
    method), 41  
delete\_shares\_groups() (civis.resources.\_resources.Jobs  
    method), 72  
delete\_shares\_groups() (civis.resources.\_resources.Models  
    method), 82  
delete\_shares\_groups() (civis.resources.\_resources.Projects  
    method), 119  
delete\_shares\_groups() (civis.resources.\_resources.Reports  
    method), 143  
delete\_shares\_users() (civis.resources.\_resources.Files  
    method), 35  
delete\_shares\_users() (civis.resources.\_resources.Imports  
    method), 41  
delete\_shares\_users() (civis.resources.\_resources.Jobs  
    method), 73  
delete\_shares\_users() (civis.resources.\_resources.Models  
    method), 82  
delete\_shares\_users() (civis.resources.\_resources.Projects  
    method), 119  
delete\_shares\_users() (civis.resources.\_resources.Reports  
    method), 143  
delete\_sql\_projects() (civis.resources.\_resources.Scripts  
    method), 171  
delete\_sql\_runs() (civis.resources.\_resources.Scripts  
    method), 171  
delete\_sql\_shares\_groups()  
    (civis.resources.\_resources.Scripts method),  
    172  
delete\_sql\_shares\_users()  
    (civis.resources.\_resources.Scripts method),  
    172  
delete\_syncs() (civis.resources.\_resources.Imports  
    method), 42  
delete\_whitelist\_ips() (civis.resources.\_resources.Databases  
    method), 32

## E

environment variable  
    CIVIS\_API\_KEY, 12, 14, 15, 17, 19–22

## F

file\_to\_civis() (in module civis.io), 19  
Files (class in civis.resources.\_resources), 34

## G

get() (civis.resources.\_resources.Credentials method), 27  
get() (civis.resources.\_resources.Files method), 35  
get() (civis.resources.\_resources.Imports method), 42  
get() (civis.resources.\_resources.Jobs method), 73  
get() (civis.resources.\_resources.Models method), 82  
get() (civis.resources.\_resources.Predictions method),  
    110  
get() (civis.resources.\_resources.Projects method), 119  
get() (civis.resources.\_resources.Queries method), 134  
get() (civis.resources.\_resources.Reports method), 143  
get() (civis.resources.\_resources.Scripts method), 172  
get() (civis.resources.\_resources.Tables method), 377  
get() (civis.resources.\_resources.Users method), 395  
get\_api\_keys() (civis.resources.\_resources.Users  
    method), 397  
get\_aws\_credential\_id() (civis.APIClient method), 23  
get\_batches() (civis.resources.\_resources.Imports  
    method), 45  
get\_builds() (civis.resources.\_resources.Models method),  
    86  
get\_containers() (civis.resources.\_resources.Scripts  
    method), 176  
get\_containers\_runs() (civis.resources.\_resources.Scripts  
    method), 180  
get\_custom() (civis.resources.\_resources.Scripts  
    method), 180  
get\_custom\_runs() (civis.resources.\_resources.Scripts  
    method), 183  
get\_database\_credential\_id() (civis.APIClient method),  
    23  
get\_database\_id() (civis.APIClient method), 24  
get\_enhancements\_cass\_ncoa()  
    (civis.resources.\_resources.Tables method),  
    380  
get\_enhancements\_geocodings()  
    (civis.resources.\_resources.Tables method),  
    381  
get\_enhancements\_prepared\_matchings()  
    (civis.resources.\_resources.Tables method),  
    381  
get\_enhancements\_table\_matchings()  
    (civis.resources.\_resources.Tables method),  
    382  
get\_files\_runs() (civis.resources.\_resources.Imports  
    method), 45  
get\_javascript() (civis.resources.\_resources.Scripts  
    method), 184  
get\_javascript\_runs() (civis.resources.\_resources.Scripts  
    method), 187

- `get_python3()` (civis.resources.\_resources.Scripts method), 188
- `get_python3_runs()` (civis.resources.\_resources.Scripts method), 191
- `get_r()` (civis.resources.\_resources.Scripts method), 192
- `get_r_runs()` (civis.resources.\_resources.Scripts method), 195
- `get_runs()` (civis.resources.\_resources.Jobs method), 74
- `get_runs()` (civis.resources.\_resources.Predictions method), 112
- `get_runs()` (civis.resources.\_resources.Queries method), 135
- `get_sql()` (civis.resources.\_resources.Scripts method), 196
- `get_sql_runs()` (civis.resources.\_resources.Scripts method), 200
- `get_table_id()` (civis.APIClient method), 24
- `get_whitelist_ips()` (civis.resources.\_resources.Databases method), 32
- I**
- Imports (class in civis.resources.\_resources), 40
- J**
- Jobs (class in civis.resources.\_resources), 72
- L**
- `list()` (civis.resources.\_resources.Credentials method), 27
- `list()` (civis.resources.\_resources.Databases method), 33
- `list()` (civis.resources.\_resources.Imports method), 46
- `list()` (civis.resources.\_resources.Jobs method), 74
- `list()` (civis.resources.\_resources.Models method), 87
- `list()` (civis.resources.\_resources.Predictions method), 113
- `list()` (civis.resources.\_resources.Projects method), 122
- `list()` (civis.resources.\_resources.Queries method), 136
- `list()` (civis.resources.\_resources.Reports method), 145
- `list()` (civis.resources.\_resources.Scripts method), 201
- `list()` (civis.resources.\_resources.Tables method), 382
- `list()` (civis.resources.\_resources.Users method), 398
- `list_api_keys()` (civis.resources.\_resources.Users method), 399
- `list_batches()` (civis.resources.\_resources.Imports method), 48
- `list_builds()` (civis.resources.\_resources.Models method), 91
- `list_children()` (civis.resources.\_resources.Jobs method), 75
- `list_columns()` (civis.resources.\_resources.Tables method), 384
- `list_containers_projects()` (civis.resources.\_resources.Scripts method), 203
- `list_containers_runs()` (civis.resources.\_resources.Scripts method), 204
- `list_containers_runs_logs()` (civis.resources.\_resources.Scripts method), 204
- `list_containers_runs_outputs()` (civis.resources.\_resources.Scripts method), 205
- `list_containers_shares()` (civis.resources.\_resources.Scripts method), 206
- `list_custom()` (civis.resources.\_resources.Scripts method), 206
- `list_custom_projects()` (civis.resources.\_resources.Scripts method), 208
- `list_custom_runs()` (civis.resources.\_resources.Scripts method), 209
- `list_custom_runs_logs()` (civis.resources.\_resources.Scripts method), 210
- `list_custom_runs_outputs()` (civis.resources.\_resources.Scripts method), 210
- `list_custom_shares()` (civis.resources.\_resources.Scripts method), 211
- `list_files_runs()` (civis.resources.\_resources.Imports method), 49
- `list_history()` (civis.resources.\_resources.Scripts method), 212
- `list_javascript_projects()` (civis.resources.\_resources.Scripts method), 213
- `list_javascript_runs()` (civis.resources.\_resources.Scripts method), 213
- `list_javascript_runs_logs()` (civis.resources.\_resources.Scripts method), 214
- `list_javascript_runs_outputs()` (civis.resources.\_resources.Scripts method), 215
- `list_javascript_shares()` (civis.resources.\_resources.Scripts method), 216
- `list_me()` (civis.resources.\_resources.Users method), 400
- `list_parents()` (civis.resources.\_resources.Jobs method), 76
- `list_projects()` (civis.resources.\_resources.Files method), 36
- `list_projects()` (civis.resources.\_resources.Imports method), 50
- `list_projects()` (civis.resources.\_resources.Jobs method), 77
- `list_projects()` (civis.resources.\_resources.Models method), 92
- `list_projects()` (civis.resources.\_resources.Reports method), 147
- `list_python3_projects()` (civis.resources.\_resources.Scripts method), 216

`list_python3_runs()` (civis.resources.\_resources.Scripts method), 217

`list_python3_runs_logs()` (civis.resources.\_resources.Scripts method), 218

`list_python3_runs_outputs()` (civis.resources.\_resources.Scripts method), 218

`list_python3_shares()` (civis.resources.\_resources.Scripts method), 219

`list_r_projects()` (civis.resources.\_resources.Scripts method), 220

`list_r_runs()` (civis.resources.\_resources.Scripts method), 221

`list_r_runs_logs()` (civis.resources.\_resources.Scripts method), 222

`list_r_runs_outputs()` (civis.resources.\_resources.Scripts method), 222

`list_r_shares()` (civis.resources.\_resources.Scripts method), 223

`list_runs()` (civis.resources.\_resources.Imports method), 51

`list_runs()` (civis.resources.\_resources.Predictions method), 114

`list_runs()` (civis.resources.\_resources.Queries method), 137

`list_schedules()` (civis.resources.\_resources.Models method), 93

`list_schedules()` (civis.resources.\_resources.Predictions method), 115

`list_schemas()` (civis.resources.\_resources.Databases method), 33

`list_shares()` (civis.resources.\_resources.Files method), 37

`list_shares()` (civis.resources.\_resources.Imports method), 51

`list_shares()` (civis.resources.\_resources.Jobs method), 78

`list_shares()` (civis.resources.\_resources.Models method), 93

`list_shares()` (civis.resources.\_resources.Projects method), 123

`list_shares()` (civis.resources.\_resources.Reports method), 148

`list_snapshots()` (civis.resources.\_resources.Reports method), 149

`list_sql_projects()` (civis.resources.\_resources.Scripts method), 224

`list_sql_runs()` (civis.resources.\_resources.Scripts method), 225

`list_sql_runs_logs()` (civis.resources.\_resources.Scripts method), 226

`list_sql_runs_outputs()` (civis.resources.\_resources.Scripts method), 226

`list_sql_shares()` (civis.resources.\_resources.Scripts

method), 227

`list_types()` (civis.resources.\_resources.Models method), 94

`list_types()` (civis.resources.\_resources.Scripts method), 227

`list_whitelist_ips()` (civis.resources.\_resources.Databases method), 33

## M

Models (class in civis.resources.\_resources), 81

## P

PaginatedResponse (class in civis.response), 25

`patch()` (civis.resources.\_resources.Models method), 94

`patch()` (civis.resources.\_resources.Predictions method), 115

`patch()` (civis.resources.\_resources.Reports method), 150

`patch()` (civis.resources.\_resources.Scripts method), 228

`patch()` (civis.resources.\_resources.Tables method), 386

`patch_containers()` (civis.resources.\_resources.Scripts method), 233

`patch_containers_runs()` (civis.resources.\_resources.Scripts method), 239

`patch_custom()` (civis.resources.\_resources.Scripts method), 239

`patch_javascript()` (civis.resources.\_resources.Scripts method), 243

`patch_me()` (civis.resources.\_resources.Users method), 401

`patch_python3()` (civis.resources.\_resources.Scripts method), 248

`patch_r()` (civis.resources.\_resources.Scripts method), 254

`patch_snapshots()` (civis.resources.\_resources.Reports method), 152

`patch_sql()` (civis.resources.\_resources.Scripts method), 260

PollableResult (class in civis.polling), 25

`post()` (civis.resources.\_resources.Credentials method), 28

`post()` (civis.resources.\_resources.Files method), 37

`post()` (civis.resources.\_resources.Imports method), 52

`post()` (civis.resources.\_resources.Models method), 96

`post()` (civis.resources.\_resources.Projects method), 124

`post()` (civis.resources.\_resources.Queries method), 138

`post()` (civis.resources.\_resources.Reports method), 154

`post()` (civis.resources.\_resources.Scripts method), 266

`post()` (civis.resources.\_resources.Tables method), 387

`post_api_keys()` (civis.resources.\_resources.Users method), 403

`post_authenticate()` (civis.resources.\_resources.Credentials method), 29

`post_batches()` (civis.resources.\_resources.Imports method), 56



[post\\_builds\(\)](#) (civis.resources.\_resources.Models method), [101](#)  
[post\\_cancel\(\)](#) (civis.resources.\_resources.Imports method), [57](#)  
[post\\_cancel\(\)](#) (civis.resources.\_resources.Scripts method), [270](#)  
[post\\_containers\(\)](#) (civis.resources.\_resources.Scripts method), [271](#)  
[post\\_containers\\_runs\(\)](#) (civis.resources.\_resources.Scripts method), [277](#)  
[post\\_containers\\_runs\\_heartbeats\(\)](#) (civis.resources.\_resources.Scripts method), [277](#)  
[post\\_containers\\_runs\\_logs\(\)](#) (civis.resources.\_resources.Scripts method), [277](#)  
[post\\_containers\\_runs\\_outputs\(\)](#) (civis.resources.\_resources.Scripts method), [278](#)  
[post\\_custom\(\)](#) (civis.resources.\_resources.Scripts method), [278](#)  
[post\\_custom\\_runs\(\)](#) (civis.resources.\_resources.Scripts method), [283](#)  
[post\\_custom\\_runs\\_outputs\(\)](#) (civis.resources.\_resources.Scripts method), [283](#)  
[post\\_enhancements\\_cass\\_ncoa\(\)](#) (civis.resources.\_resources.Tables method), [388](#)  
[post\\_enhancements\\_geocodings\(\)](#) (civis.resources.\_resources.Tables method), [389](#)  
[post\\_enhancements\\_prepared\\_matchings\(\)](#) (civis.resources.\_resources.Tables method), [389](#)  
[post\\_enhancements\\_table\\_matchings\(\)](#) (civis.resources.\_resources.Tables method), [390](#)  
[post\\_files\(\)](#) (civis.resources.\_resources.Imports method), [58](#)  
[post\\_files\\_runs\(\)](#) (civis.resources.\_resources.Imports method), [59](#)  
[post\\_grants\(\)](#) (civis.resources.\_resources.Reports method), [156](#)  
[post\\_javascript\(\)](#) (civis.resources.\_resources.Scripts method), [284](#)  
[post\\_javascript\\_runs\(\)](#) (civis.resources.\_resources.Scripts method), [289](#)  
[post\\_javascript\\_runs\\_outputs\(\)](#) (civis.resources.\_resources.Scripts method), [289](#)  
[post\\_python3\(\)](#) (civis.resources.\_resources.Scripts method), [290](#)  
[post\\_python3\\_runs\(\)](#) (civis.resources.\_resources.Scripts method), [295](#)  
[post\\_python3\\_runs\\_outputs\(\)](#) (civis.resources.\_resources.Scripts method), [296](#)  
[post\\_r\(\)](#) (civis.resources.\_resources.Scripts method), [296](#)  
[post\\_r\\_runs\(\)](#) (civis.resources.\_resources.Scripts method), [302](#)  
[post\\_r\\_runs\\_outputs\(\)](#) (civis.resources.\_resources.Scripts method), [302](#)  
[post\\_refresh\(\)](#) (civis.resources.\_resources.Tables method), [391](#)  
[post\\_run\(\)](#) (civis.resources.\_resources.Scripts method), [303](#)  
[post\\_runs\(\)](#) (civis.resources.\_resources.Imports method), [59](#)  
[post\\_runs\(\)](#) (civis.resources.\_resources.Jobs method), [78](#)  
[post\\_runs\(\)](#) (civis.resources.\_resources.Predictions method), [117](#)  
[post\\_runs\(\)](#) (civis.resources.\_resources.Queries method), [140](#)  
[post\\_snapshots\(\)](#) (civis.resources.\_resources.Reports method), [158](#)  
[post\\_sql\(\)](#) (civis.resources.\_resources.Scripts method), [303](#)  
[post\\_sql\\_runs\(\)](#) (civis.resources.\_resources.Scripts method), [309](#)  
[post\\_syncs\(\)](#) (civis.resources.\_resources.Imports method), [59](#)  
[post\\_temporary\(\)](#) (civis.resources.\_resources.Credentials method), [30](#)  
[post\\_trigger\\_email\(\)](#) (civis.resources.\_resources.Jobs method), [79](#)  
[post\\_whitelist\\_ips\(\)](#) (civis.resources.\_resources.Databases method), [34](#)  
[Predictions](#) (class in civis.resources.\_resources), [110](#)  
[Projects](#) (class in civis.resources.\_resources), [118](#)  
[put\(\)](#) (civis.resources.\_resources.Credentials method), [31](#)  
[put\(\)](#) (civis.resources.\_resources.Imports method), [61](#)  
[put\(\)](#) (civis.resources.\_resources.Projects method), [127](#)  
[put\\_archive\(\)](#) (civis.resources.\_resources.Imports method), [65](#)  
[put\\_archive\(\)](#) (civis.resources.\_resources.Models method), [102](#)  
[put\\_archive\(\)](#) (civis.resources.\_resources.Projects method), [129](#)  
[put\\_archive\(\)](#) (civis.resources.\_resources.Reports method), [160](#)  
[put\\_containers\(\)](#) (civis.resources.\_resources.Scripts method), [310](#)  
[put\\_containers\\_archive\(\)](#) (civis.resources.\_resources.Scripts method), [316](#)  
[put\\_containers\\_projects\(\)](#) (civis.resources.\_resources.Scripts method),

320  
put\_containers\_shares\_groups()  
    (civis.resources.\_resources.Scripts method),  
    320  
put\_containers\_shares\_users()  
    (civis.resources.\_resources.Scripts method),  
    321  
put\_custom() (civis.resources.\_resources.Scripts  
    method), 322  
put\_custom\_archive() (civis.resources.\_resources.Scripts  
    method), 326  
put\_custom\_projects() (civis.resources.\_resources.Scripts  
    method), 329  
put\_custom\_shares\_groups()  
    (civis.resources.\_resources.Scripts method),  
    329  
put\_custom\_shares\_users()  
    (civis.resources.\_resources.Scripts method),  
    330  
put\_javascript() (civis.resources.\_resources.Scripts  
    method), 331  
put\_javascript\_archive() (civis.resources.\_resources.Scripts  
    method), 336  
put\_javascript\_projects()  
    (civis.resources.\_resources.Scripts method),  
    340  
put\_javascript\_shares\_groups()  
    (civis.resources.\_resources.Scripts method),  
    340  
put\_javascript\_shares\_users()  
    (civis.resources.\_resources.Scripts method),  
    341  
put\_predictions() (civis.resources.\_resources.Models  
    method), 106  
put\_projects() (civis.resources.\_resources.Files method),  
    38  
put\_projects() (civis.resources.\_resources.Imports  
    method), 68  
put\_projects() (civis.resources.\_resources.Jobs method),  
    79  
put\_projects() (civis.resources.\_resources.Models  
    method), 107  
put\_projects() (civis.resources.\_resources.Reports  
    method), 162  
put\_python3() (civis.resources.\_resources.Scripts  
    method), 341  
put\_python3\_archive() (civis.resources.\_resources.Scripts  
    method), 347  
put\_python3\_projects() (civis.resources.\_resources.Scripts  
    method), 351  
put\_python3\_shares\_groups()  
    (civis.resources.\_resources.Scripts method),  
    351  
put\_python3\_shares\_users()  
    (civis.resources.\_resources.Scripts method),  
    352  
put\_r() (civis.resources.\_resources.Scripts method), 353  
put\_r\_archive() (civis.resources.\_resources.Scripts  
    method), 358  
put\_r\_projects() (civis.resources.\_resources.Scripts  
    method), 362  
put\_r\_shares\_groups() (civis.resources.\_resources.Scripts  
    method), 362  
put\_r\_shares\_users() (civis.resources.\_resources.Scripts  
    method), 363  
put\_schedules() (civis.resources.\_resources.Models  
    method), 107  
put\_schedules() (civis.resources.\_resources.Predictions  
    method), 117  
put\_scripts() (civis.resources.\_resources.Queries  
    method), 141  
put\_shares\_groups() (civis.resources.\_resources.Files  
    method), 38  
put\_shares\_groups() (civis.resources.\_resources.Imports  
    method), 68  
put\_shares\_groups() (civis.resources.\_resources.Jobs  
    method), 79  
put\_shares\_groups() (civis.resources.\_resources.Models  
    method), 108  
put\_shares\_groups() (civis.resources.\_resources.Projects  
    method), 132  
put\_shares\_groups() (civis.resources.\_resources.Reports  
    method), 162  
put\_shares\_users() (civis.resources.\_resources.Files  
    method), 39  
put\_shares\_users() (civis.resources.\_resources.Imports  
    method), 69  
put\_shares\_users() (civis.resources.\_resources.Jobs  
    method), 80  
put\_shares\_users() (civis.resources.\_resources.Models  
    method), 109  
put\_shares\_users() (civis.resources.\_resources.Projects  
    method), 133  
put\_shares\_users() (civis.resources.\_resources.Reports  
    method), 163  
put\_sql() (civis.resources.\_resources.Scripts method),  
    364  
put\_sql\_archive() (civis.resources.\_resources.Scripts  
    method), 370  
put\_sql\_projects() (civis.resources.\_resources.Scripts  
    method), 374  
put\_sql\_shares\_groups() (civis.resources.\_resources.Scripts  
    method), 374  
put\_sql\_shares\_users() (civis.resources.\_resources.Scripts  
    method), 375  
put\_syncs() (civis.resources.\_resources.Imports method),  
    70

## Q

Queries (class in `civis.resources._resources`), [134](#)  
`query_civis()` (in module `civis.io`), [21](#)

## R

`read_civis()` (in module `civis.io`), [15](#)  
`read_civis_sql()` (in module `civis.io`), [16](#)  
Reports (class in `civis.resources._resources`), [142](#)  
Response (class in `civis.response`), [25](#)

## S

Scripts (class in `civis.resources._resources`), [164](#)

## T

Tables (class in `civis.resources._resources`), [376](#)  
`transfer_table()` (in module `civis.io`), [20](#)

## U

`username` (`civis.APIClient` attribute), [24](#)  
Users (class in `civis.resources._resources`), [394](#)