Package: solvebio (via r-universe)

October 19, 2026

Type Package

Title The Official SolveBio API Client

Version 2.15.1

Date 2025-02-06

Author Tobi Guennel [aut, cre]

Maintainer Tobi Guennel <tobi.guennel@precisionformedicine.com>

Description R language bindings for SolveBio's API (DEPRECATED: This package will no longer be maintained after March 31, 2026). SolveBio is a biomedical knowledge hub that enables life science organizations to collect and harmonize the complex, disparate ``multi-omic'' data essential for today's R&D and BI needs.

URL https://github.com/solvebio/solvebio-r

Imports httr, jsonlite, mime

License MIT + file LICENSE

Collate solvebio.R annotation.R application.R beacon.R beacon_set.R dataset.R dataset_commit.R dataset_export.R dataset_field.R dataset_import.R dataset_migration.R dataset_template.R global_search.R object.R saved_query.R shiny.R task.R user.R utils.R vault.R zzz.R

RoxygenNote 7.3.2

Encoding UTF-8

Suggests testthat, shiny, shinyjs, openssl

Config/pak/sysreqs libssl-dev

Repository https://solvebio.r-universe.dev

Date/Publication 2025-10-15 10:57:06 UTC

RemoteUrl https://github.com/solvebio/solvebio-r

RemoteRef HEAD

RemoteSha ed132b8ed18ef992ae3b154646e72bb0edb8eb8d

Contents

Annotator.annotate . . . 4
Application.all . . . 5
Application.create . . . 5
Application.delete . . . 6
Application.retrieve . . . 7
Application.update . . . 7
Beacon.all . . . 8
Beacon.create . . . 9
Beacon.delete . . . 10
Beacon.query . . . 10
Beacon.retrieve . . . 11
Beacon.update . . . 12
BeaconSet.all . . . 12
BeaconSet.create . . . 13
BeaconSet.delete . . . 14
BeaconSet.query . . . 15
BeaconSet.retrieve . . . 15
BeaconSet.update . . . 16
createEnv . . . 17
Dataset.activity . . . 17
Dataset.all . . . 18
Dataset.count . . . 19
Dataset.create . . . 19
Dataset.data . . . 20
Dataset.delete . . . 21
Dataset.disable_global_beacon . . . 21
Dataset.enable_global_beacon . . . 22
Dataset.facets . . . 23
Dataset.fields . . . 23
Dataset.get_by_full_path . . . 24
Dataset.get_global_beacon_status . . . 25
Dataset.get_or_create_by_full_path . . . 25
Dataset.query . . . 26
Dataset.retrieve . . . 27
Dataset.template . . . 27
Dataset.update . . . 28
DatasetCommit.all . . . 29
DatasetCommit.delete . . . 29
DatasetCommit.retrieve . . . 30
DatasetExport.all . . . 31
DatasetExport.create . . . 31
DatasetExport.delete . . . 32
DatasetExport.get_download_url . . . 33
DatasetExport.retrieve . . . 33
DatasetField.all . . . 34
DatasetField.create . . . 35

DatasetField.facets . . . 36
DatasetField.retrieve . . . 36
DatasetField.update . . . 37
DatasetImport.all . . . 38
DatasetImport.create . . . 38
DatasetImport.delete . . . 39
DatasetImport.retrieve . . . 40
DatasetMigration.all . . . 40
DatasetMigration.create . . . 41
DatasetMigration.delete . . . 42
DatasetMigration.retrieve . . . 43
DatasetTemplate.all . . . 43
DatasetTemplate.create . . . 44
DatasetTemplate.delete . . . 45
DatasetTemplate.retrieve . . . 45
DatasetTemplate.update . . . 46
Expression.evaluate . . . 47
GlobalSearch.facets . . . 48
GlobalSearch.request . . . 48
GlobalSearch.search . . . 50
GlobalSearch.subjects . . . 51
GlobalSearch.subjects_count . . . 51
login . . . 52
Object.all . . . 53
Object.create . . . 53
Object.data . . . 54
Object.delete . . . 55
Object.disable_global_beacon . . . 56
Object.enable_global_beacon . . . 56
Object.fields . . . 57
Object.get_by_full_path . . . 58
Object.get_by_path . . . 58
Object.get_download_url . . . 59
Object.get_global_beacon_status . . . 60
Object.get_or_upload_file . . . 60
Object.query . . . 61
Object.retrieve . . . 62
Object.update . . . 63
Object.upload_file . . . 63
protectedServer . . . 64
protectedServerJS . . . 65
SavedQuery.all . . . 66
SavedQuery.create . . . 66
SavedQuery.delete . . . 67
SavedQuery.retrieve . . . 68
SavedQuery.update . . . 68
Task.all . . . 69
Task.follow . . . 70

Task.retrieve 70
User.retrieve 71
Vault.all 72
Vault.create 72
Vault.create_dataset 73
Vault.create_folder 74
Vault.datasets 75
Vault.delete 75
Vault.files 76
Vault.folders 77
Vault.get_by_full_path 77
Vault.get_or_create_by_full_path 78
Vault.get_personal_vault 79
Vault.objects 79
Vault.retrieve 80
Vault.search 81
Vault.update 81

Index **83**

`Annotator.annotate` *Annotator.annotate*

Description

Annotate a data table/frame with additional fields.

Usage

```
Annotator.annotate(
  records,
  fields,
  include_errors = FALSE,
  raw = FALSE,
  env = solvebio:::.solveEnv
)
```

Arguments

`records`	The data table or data frame to annotate.
`fields`	The fields to add.
`include_errors`	Set to TRUE to include errors in the output (default: FALSE).
`raw`	Set to TRUE to return the raw response (default: FALSE).
`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Annotator.annotate(records=tbl, fields=fields)

## End(Not run)
```

`Application.all` *Application.all*

Description

Retrieves the metadata about all application on SolveBio available to the current user.

Usage

```
Application.all(env = solvebio:::.solveEnv, ...)
```

Arguments

`env`	(optional) Custom client environment.
`...`	(optional) Additional query parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Application.all()

## End(Not run)
```

`Application.create` *Application.create*

Description

Create a new SolveBio application.

Usage

```
Application.create(name, redirect_uris, env = solvebio:::.solveEnv, ...)
```

Arguments

name	The name of the application.
redirect_uris	A list of space-separated OAuth2 redirect URIs.
env	(optional) Custom client environment.
...	(optional) Additional application attributes.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Application.create(
                   name="My new application",
                   redirect_uris="http://localhost:3838/"
                   )

## End(Not run)
```

Application.delete *Application.delete*

Description

Delete a specific application from SolveBio.

Usage

```
Application.delete(client_id, env = solvebio:::.solveEnv)
```

Arguments

client_id	The client ID for the application.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Application.delete("abcd1234")

## End(Not run)
```

Application.retrieve *Application.retrieve*

Description

Retrieves the metadata about a specific application SolveBio.

Usage

```
Application.retrieve(client_id, env = solvebio:::.solveEnv)
```

Arguments

client_id	The client ID for the application.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Application.retrieve("abcd1234")

## End(Not run)
```

Application.update *Application.update*

Description

Updates the attributes of an existing application.

Usage

```
Application.update(client_id, env = solvebio:::.solveEnv, ...)
```

Arguments

client_id	The client ID for the application.
env	(optional) Custom client environment.
...	Application attributes to change.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Application.update(
                   "abcd1234",
                   name="New app name"
                   )

## End(Not run)
```

`Beacon.all` *Beacon.all*

Description

Retrieves the metadata about all beacons on SolveBio accessible to the current user.

Usage

```
Beacon.all(env = solvebio:::.solveEnv, ...)
```

Arguments

`env`	(optional) Custom client environment.
`...`	(optional) Additional query parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Beacon.all()

## End(Not run)
```

`Beacon.create` *Beacon.create*

Description

Add a new beacon to an existing beacon set. The beacon set must already exist in order to add beacons.

Usage

```
Beacon.create(
  beacon_set_id,
  vault_object_id,
  title,
  env = solvebio:::.solveEnv,
  ...
)
```

Arguments

`beacon_set_id`	The ID of the parent beacon set.
`vault_object_id`	The ID of the vault object (i.e. dataset) queried by the beacon.
`title`	The title displayed for the beacon.
`env`	(optional) Custom client environment.
`...`	(optional) Additional beacon attributes (such as description and params).

References

https://docs.solvebio.com/

Examples

```
## Not run:
Beacon.create(
             beacon_set_id="1234",
             vault_object_id="1234567890",
             title="My new beacon"
             )

## End(Not run)
```

Beacon.delete *Beacon.delete*

Description

Delete a specific beacon from SolveBio.

Usage

```
Beacon.delete(id, env = solvebio:::.solveEnv)
```

Arguments

id	The ID of the beacon.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Beacon.delete("1234567890")

## End(Not run)
```

Beacon.query *Beacon.query*

Description

Query an individual beacon.

Usage

```
Beacon.query(id, query, entity_type, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of the beacon.
query	The entity ID or query string.
entity_type	(optional) A valid SolveBio entity type.
env	(optional) Custom client environment.
...	(optional) Additional query parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Beacon.query(
             id="1234",
             query="BRCA2",
             entity_type="gene"
             )

## End(Not run)
```

Beacon.retrieve *Beacon.retrieve*

Description

Retrieves the metadata about a specific beacon on SolveBio.

Usage

```
Beacon.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

id	The ID of the beacon.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Beacon.retrieve("1234")

## End(Not run)
```

Beacon.update *Beacon.update*

Description

Updates the attributes of an existing beacon.

Usage

```
Beacon.update(id, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of the beacon to update.
env	(optional) Custom client environment.
...	Beacon attributes to change.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Beacon.update(
              id="1234",
              title="New Beacon Title"
             )

## End(Not run)
```

BeaconSet.all *BeaconSet.all*

Description

Retrieves the metadata about all beacon sets on SolveBio accessible to the current user.

Usage

```
BeaconSet.all(env = solvebio:::.solveEnv, ...)
```

Arguments

env	(optional) Custom client environment.
...	(optional) Additional query parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
BeaconSet.all()

## End(Not run)
```

BeaconSet.create *BeaconSet.create*

Description

Create a new beacon set.

Usage

```
BeaconSet.create(
  title,
  description,
  is_shared = FALSE,
  env = solvebio:::.solveEnv,
  ...
)
```

Arguments

title	The title displayed for the beacon set.
description	(optional) An optional description for the new beacon set.
is_shared	If TRUE, everyone else in your account will be able to see and query the beacon set, but will not be able to edit it. (Default: FALSE)
env	(optional) Custom client environment.
...	(optional) Additional beacon set attributes.

References

https://docs.solvebio.com/

Examples

```
## Not run:
BeaconSet.create(
                 title="My new beacon set",
                 )

## End(Not run)
```

BeaconSet.delete *BeaconSet.delete*

Description

Delete a specific beacon set (including all its beacons) from SolveBio.

Usage

```
BeaconSet.delete(id, env = solvebio:::.solveEnv)
```

Arguments

id	The ID of the beacon set.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
BeaconSet.delete("1234")

## End(Not run)
```

`BeaconSet.query` *BeaconSet.query*

Description

Query a beacon set (i.e. all the beacons within a beacon set).

Usage

```
BeaconSet.query(id, query, entity_type, env = solvebio:::.solveEnv, ...)
```

Arguments

`id`	The ID of the beacon set.
`query`	The entity ID or query string.
`entity_type`	(optional) A valid SolveBio entity type.
`env`	(optional) Custom client environment.
`...`	(optional) Additional query parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
BeaconSet.query(
                id="1234",
                query="BRCA2",
                entity_type="gene"
                )

## End(Not run)
```

`BeaconSet.retrieve` *BeaconSet.retrieve*

Description

Retrieves the metadata about a specific beacon set on SolveBio.

Usage

```
BeaconSet.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

id	The ID of the beacon set.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
BeaconSet.retrieve("1234")

## End(Not run)
```

BeaconSet.update *BeaconSet.update*

Description

Updates the attributes of an existing beacon set.

Usage

```
BeaconSet.update(id, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of the beacon set to update.
env	(optional) Custom client environment.
...	Beacon set attributes to change.

References

https://docs.solvebio.com/

Examples

```
## Not run:
BeaconSet.update(
                 id="1234",
                 title="New Beacon Set Title"
                )

## End(Not run)
```

createEnv *createEnv*

Description

Create a new SolveBio environment.

Usage

```
createEnv(token, token_type = "Token", host = .solveEnv$host)
```

Arguments

token	A SolveBio API key or OAuth2 token
token_type	SolveBio token type (default: Token)
host	(optional) The SolveBio API host (default: https://api.solvebio.com)

References

https://docs.solvebio.com/

Examples

```
## Not run:
env <- createEnv("MyAPIkey")
User.retrieve(env = myEnv)

## End(Not run)
```

Dataset.activity *Dataset.activity*

Description

A helper function to get or follow the current activity on a dataset.

Usage

```
Dataset.activity(id, follow = TRUE, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of a SolveBio dataset
follow	Follow active tasks until they complete.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.activity("1234567890")

## End(Not run)
```

`Dataset.all` *Dataset.all*

Description

Retrieves the metadata about datasets on SolveBio.

Usage

```
Dataset.all(env = solvebio:::.solveEnv, ...)
```

Arguments

`env`	(optional) Custom client environment.
`...`	(optional) Additional query parameters (e.g. page).

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.all()

## End(Not run)
```

Dataset.count *Dataset.count*

Description

Returns the total number of records for a given SolveBio dataset.

Usage

```
Dataset.count(id, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of a SolveBio dataset, or a Dataset object.
env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. filters, limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
dataset <- Dataset.get_by_full_path("solvebio:public:/ClinVar/3.7.4-2017-01-30/Variants-GRCh37")
Dataset.count(dataset)
Dataset.count(dataset, filters='[["gene_symbol", "BRCA2"]]')

## End(Not run)
```

Dataset.create *Dataset.create*

Description

Create an empty SolveBio dataset.

Usage

```
Dataset.create(
  vault_id,
  vault_parent_object_id,
  name,
  env = solvebio:::.solveEnv,
  ...
)
```

Arguments

vault_id	The ID of the vault.
vault_parent_object_id	The parent object (folder) ID in the vault.
name	The name of the dataset in the parent folder.
env	(optional) Custom client environment.
...	(optional) Additional dataset attributes.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.create(vault_id=vault$id, vault_parent_object_id=NULL, name="My Dataset")

## End(Not run)
```

Dataset.data *Dataset.data*

Description

Returns one page of documents from a SolveBio dataset and processes the response.

Usage

```
Dataset.data(id, filters, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of a SolveBio dataset, or a Dataset object.
filters	(optional) Query filters.
env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.data("1234567890")

## End(Not run)
```

Dataset.delete *Dataset.delete*

Description

Delete a specific dataset from SolveBio.

Usage

```
Dataset.delete(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of a SolveBio dataset
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.delete("1234567890")

## End(Not run)
```

Dataset.disable_global_beacon *Dataset.disable_global_beacon*

Description

Disables Global Beacon for the dataset.

Usage

```
Dataset.disable_global_beacon(id, env = solvebio:::.solveEnv)
```

Arguments

id	The ID of a SolveBio dataset.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.disable_global_beacon("1234567890")

## End(Not run)
```

Dataset.enable_global_beacon *Dataset.enable_global_beacon*

Description

Enables Global Beacon for the the dataset.

Usage

```
Dataset.enable_global_beacon(id, env = solvebio:::.solveEnv)
```

Arguments

id	The ID of a SolveBio dataset.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.enable_global_beacon("1234567890")

## End(Not run)
```

`Dataset.facets` *Dataset.facets*

Description

Retrieves aggregated statistics or term counts for one or more fields in a SolveBio dataset. Returns a list of data frames, one for each requested facet.

Usage

```
Dataset.facets(id, facets, env = solvebio:::.solveEnv, ...)
```

Arguments

`id`	The ID of a SolveBio dataset, or a Dataset object.
`facets`	A list of one or more field facets.
`env`	(optional) Custom client environment.
`...`	(optional) Additional query parameters (e.g. filters, limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.facets("1234567890", list("clinical_significance", "gene_symbol"))

## End(Not run)
```

`Dataset.fields` *Dataset.fields*

Description

Retrieves the list of fields and field metadata for a dataset.

Usage

```
Dataset.fields(id, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of a SolveBio dataset, or a Dataset object.
env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.fields("1234567890")

## End(Not run)
```

Dataset.get_by_full_path *Dataset.get_by_full_path*

Description

A helper function to get a dataset by its full path.

Usage

```
Dataset.get_by_full_path(full_path, env = solvebio:::.solveEnv)
```

Arguments

full_path	A valid full path to a dataset.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.get_by_full_path("solvebio:public:/ClinVar/3.7.4-2017-01-30/Variants-GRCh37")

## End(Not run)
```

Dataset.get_global_beacon_status
Dataset.get_global_beacon_status

Description

Retrieves the global beacon status for the dataset.

Usage

```
Dataset.get_global_beacon_status(
  id,
  raise_on_disabled = FALSE,
  env = solvebio:::.solveEnv
)
```

Arguments

id	The ID of a SolveBio dataset.
raise_on_disabled	Whether to raise an exception if Global Beacon is disabled or to return NULL.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.get_global_beacon_status("1234567890")

## End(Not run)
```

Dataset.get_or_create_by_full_path
Dataset.get_or_create_by_full_path

Description

A helper function to get or create a dataset by its full path.

Usage

```
Dataset.get_or_create_by_full_path(full_path, env = solvebio:::.solveEnv, ...)
```

Arguments

full_path	A valid full path to a dataset.
env	(optional) Custom client environment.
...	Additional dataset creation parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.get_or_create_by_full_path("MyVault:/folder/sub-folder/dataset")

## End(Not run)
```

Dataset.query *Dataset.query*

Description

Queries a SolveBio dataset and returns an R data frame containing all records. Returns a single page of results otherwise (default).

Usage

```
Dataset.query(
  id,
  paginate = FALSE,
  use_field_titles = TRUE,
  env = solvebio:::.solveEnv,
  ...
)
```

Arguments

id	The ID of a SolveBio dataset, or a Dataset object.
paginate	When set to TRUE, retrieves all records (memory permitting).
use_field_titles	(optional) Use field title instead of field name for query.
env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. filters, limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.query("12345678790", paginate=TRUE)

## End(Not run)
```

`Dataset.retrieve` *Dataset.retrieve*

Description

Retrieves the metadata about a specific dataset from SolveBio.

Usage

```
Dataset.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

`id`	String The ID of a SolveBio dataset
`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.retrieve("1234567890")

## End(Not run)
```

`Dataset.template` *Dataset.template*

Description

Retrieves the template for a dataset.

Usage

```
Dataset.template(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of a SolveBio dataset
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.template("1234567890")

## End(Not run)
```

Dataset.update *Dataset.update*

Description

Updates the attributes of an existing dataset.

Usage

```
Dataset.update(id, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of the dataset to update.
env	(optional) Custom client environment.
...	Dataset attributes to change.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Dataset.update(
               id="1234",
               name="New Dataset Name",
              )

## End(Not run)
```

DatasetCommit.all *DatasetCommit.all*

Description

Retrieves the metadata about all dataset commits on SolveBio.

Usage

```
DatasetCommit.all(env = solvebio:::.solveEnv, ...)
```

Arguments

env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. page).

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetCommit.all()

## End(Not run)
```

DatasetCommit.delete *DatasetCommit.delete*

Description

Deletes a specific dataset commit on SolveBio.

Usage

```
DatasetCommit.delete(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID or full name of a SolveBio dataset commit.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetCommit.delete(<ID>)

## End(Not run)
```

`DatasetCommit.retrieve` *DatasetCommit.retrieve*

Description

Retrieves the metadata about a specific dataset commit on SolveBio.

Usage

```
DatasetCommit.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

`id`	String The ID of a SolveBio dataset commit.
`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetCommit.retrieve(<ID>)

## End(Not run)
```

DatasetExport.all *DatasetExport.all*

Description

Retrieves the metadata about all dataset exports on SolveBio.

Usage

```
DatasetExport.all(env = solvebio:::.solveEnv, ...)
```

Arguments

env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. page).

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetExport.all()

## End(Not run)
```

DatasetExport.create *DatasetExport.create*

Description

Create a new dataset export.

Usage

```
DatasetExport.create(
  dataset_id,
  format = "json",
  params = list(),
  follow = FALSE,
  env = solvebio:::.solveEnv,
  ...
)
```

Arguments

dataset_id	The target dataset ID.
format	(optional) The export format (default: json).
params	(optional) Query parameters for the export.
follow	(default: FALSE) Follow the export task until it completes.
env	(optional) Custom client environment.
...	(optional) Additional dataset export parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetExport.create(dataset_id=<ID>, format='json', params=list(fields=c("field_1"), limit=100))

## End(Not run)
```

DatasetExport.delete *DatasetExport.delete*

Description

Deletes a specific dataset export on SolveBio.

Usage

```
DatasetExport.delete(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of a SolveBio dataset export.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetExport.delete(<ID>)

## End(Not run)
```

DatasetExport.get_download_url
DatasetExport.get_download_url

Description

Helper method to get the download URL for a dataset export.

Usage

```
DatasetExport.get_download_url(id, env = solvebio:::.solveEnv)
```

Arguments

id	The ID of the dataset export.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetExport.get_download_url("1234567890")

## End(Not run)
```

DatasetExport.retrieve
DatasetExport.retrieve

Description

Retrieves the metadata about a specific dataset export on SolveBio.

Usage

```
DatasetExport.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of a SolveBio dataset export.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetExport.retrieve(<ID>)

## End(Not run)
```

`DatasetField.all` *DatasetField.all*

Description

Retrieves the metadata about all dataset fields on SolveBio.

Usage

```
DatasetField.all(env = solvebio:::.solveEnv, ...)
```

Arguments

`env`	(optional) Custom client environment.
`...`	(optional) Additional query parameters (e.g. page).

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetField.all()

## End(Not run)
```

DatasetField.create *DatasetField.create*

Description

Create a new dataset field.

Usage

```
DatasetField.create(
  dataset_id,
  name,
  data_type = "auto",
  env = solvebio:::.solveEnv,
  ...
)
```

Arguments

dataset_id	The dataset ID.
name	The name of the dataset field.
data_type	(optional) The data type for the field (default: auto).
env	(optional) Custom client environment.
...	(optional) Additional dataset import attributes.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetField.create(dataset_id=<ID>, name="my_field", title="My Field", data_type="string")

## End(Not run)
```

DatasetField.facets *DatasetField.facets*

Description

Returns the facets for a given dataset field.

Usage

```
DatasetField.facets(id, env = solvebio:::.solveEnv, ...)
```

Arguments

id	String The ID of a dataset field.
env	(optional) Custom client environment.
...	(optional) Additional query parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetField.facets(691)

## End(Not run)
```

DatasetField.retrieve *DatasetField.retrieve*

Description

Retrieves the metadata about a specific dataset field.

Usage

```
DatasetField.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of a dataset field.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetField.retrieve(691)

## End(Not run)
```

DatasetField.update *DatasetField.update*

Description

Updates the attributes of an existing dataset field. NOTE: The data_type of a field cannot be changed.

Usage

```
DatasetField.update(id, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of the dataset field to update.
env	(optional) Custom client environment.
...	Dataset field attributes to change.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetField.update(
                    id="1234",
                    title="New Field Title"
                   )

## End(Not run)
```

DatasetImport.all *DatasetImport.all*

Description

Retrieves the metadata about all dataset imports on SolveBio.

Usage

```
DatasetImport.all(env = solvebio:::.solveEnv, ...)
```

Arguments

env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. page).

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetImport.all()

## End(Not run)
```

DatasetImport.create *DatasetImport.create*

Description

Create a new dataset import. Either an object_id, manifest, or data_records is required.

Usage

```
DatasetImport.create(
  dataset_id,
  commit_mode = "append",
  env = solvebio:::.solveEnv,
  ...
)
```

Arguments

dataset_id	The target dataset ID.
commit_mode	(optional) The commit mode (default: append).
env	(optional) Custom client environment.
...	(optional) Additional dataset import attributes.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetImport.create(dataset_id=<ID>, upload_id=<ID>)

## End(Not run)
```

DatasetImport.delete *DatasetImport.delete*

Description

Deletes a specific dataset import on SolveBio.

Usage

```
DatasetImport.delete(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of a SolveBio dataset import.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetImport.delete(<ID>)

## End(Not run)
```

DatasetImport.retrieve *DatasetImport.retrieve*

Description

Retrieves the metadata about a specific dataset import on SolveBio.

Usage

```
DatasetImport.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of a SolveBio dataset import.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetImport.retrieve(<ID>)

## End(Not run)
```

DatasetMigration.all *DatasetMigration.all*

Description

Retrieves the metadata about all dataset migrations on SolveBio.

Usage

```
DatasetMigration.all(env = solvebio:::.solveEnv, ...)
```

Arguments

env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. page).

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetMigration.all()

## End(Not run)
```

DatasetMigration.create

DatasetMigration.create

Description

Create a new dataset migration.

Usage

```
DatasetMigration.create(
  source_id,
  target_id,
  commit_mode = "append",
  source_params = NULL,
  target_fields = NULL,
  include_errors = FALSE,
  env = solvebio:::.solveEnv,
  ...
)
```

Arguments

source_id	The source dataset ID.
target_id	The target dataset ID.
commit_mode	(optional) The commit mode (default: append).
source_params	(optional) The query parameters used on the source dataset.
target_fields	(optional) A list of valid dataset fields to add or override in the target dataset.
include_errors	(optional) If TRUE, a new field (_errors) will be added to each record containing expression evaluation errors (default: FALSE).
env	(optional) Custom client environment.
...	(optional) Additional dataset migration attributes.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetMigration.create(dataset_id=<ID>, upload_id=<ID>)

## End(Not run)
```

DatasetMigration.delete

DatasetMigration.delete

Description

Deletes specific dataset migration on SolveBio.

Usage

```
DatasetMigration.delete(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of a SolveBio dataset migration.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetMigration.delete(<ID>)

## End(Not run)
```

DatasetMigration.retrieve *DatasetMigration.retrieve*

Description

Retrieves the metadata about a specific dataset migration on SolveBio.

Usage

```
DatasetMigration.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of a SolveBio dataset migration.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetMigration.retrieve(<ID>)

## End(Not run)
```

DatasetTemplate.all *DatasetTemplate.all*

Description

Retrieves the metadata about all dataset templates on SolveBio.

Usage

```
DatasetTemplate.all(env = solvebio:::.solveEnv, ...)
```

Arguments

env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. page).

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetTemplate.all()

## End(Not run)
```

DatasetTemplate.create

DatasetTemplate.create

Description

Create a SolveBio dataset template.

Usage

```
DatasetTemplate.create(env = solvebio:::.solveEnv, ...)
```

Arguments

env	(optional) Custom client environment.
...	(optional) Dataset template attributes.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetTemplate.create(name="My Dataset Template")

## End(Not run)
```

DatasetTemplate.delete

DatasetTemplate.delete

Description

Deletes a specific dataset template on SolveBio.

Usage

```
DatasetTemplate.delete(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID or full name of a SolveBio dataset template.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetTemplate.delete(<ID>)

## End(Not run)
```

DatasetTemplate.retrieve

DatasetTemplate.retrieve

Description

Retrieves the metadata about a specific dataset template on SolveBio.

Usage

```
DatasetTemplate.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of a SolveBio dataset template.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetTemplate.retrieve(<ID>)

## End(Not run)
```

DatasetTemplate.update

DatasetTemplate.update

Description

Updates the attributes of an existing dataset template.

Usage

```
DatasetTemplate.update(id, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of the dataset template to update.
env	(optional) Custom client environment.
...	Dataset template attributes to change.

References

https://docs.solvebio.com/

Examples

```
## Not run:
DatasetTemplate.update(
               id="1234",
               name="New Template Name",
              )

## End(Not run)
```

`Expression.evaluate` *Expression.evaluate*

Description

Evaluate a SolveBio expression.

Usage

```
Expression.evaluate(
  expression,
  data_type = "string",
  is_list = FALSE,
  data = NULL,
  raw = FALSE,
  env = solvebio:::.solveEnv
)
```

Arguments

`expression`	The SolveBio expression string.
`data_type`	The data type to cast the expression result (default: string).
`is_list`	Set to TRUE if the result is expected to be a list (default: FALSE).
`data`	Variables used in the expression (default: NULL).
`raw`	Set to TRUE to return the raw response (default: FALSE).
`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Expression.evaluate("1 + 1", data_type="integer", is_list=FALSE)

## End(Not run)
```

GlobalSearch.facets *GlobalSearch.facets*

Description

Performs a Global Search based on provided filters, entities, queries, and returns an R data frame containing the facets results from API response.

Usage

```
GlobalSearch.facets(facets, env = solvebio:::.solveEnv, ...)
```

Arguments

facets	Facets list.
env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. filters, entities, query).

References

https://docs.solvebio.com/

Examples

```
## Not run:
GlobalSearch.facets(facets="study")

## End(Not run)
```

GlobalSearch.request *GlobalSearch.request*

Description

Performs a single Global Search API request with the provided filters, queries and entities. A single request will only retrieve one page of results (based on the ‘limit‘ parameter). Use ‘GlobalSearch.search()‘ to retrieve all pages of results. Returns the full API response (containing attributes: results, vaults, subjects, subjects_count, total)

Usage

```
GlobalSearch.request(
  query = NULL,
  filters,
  entities,
  env = solvebio:::.solveEnv,
  ...
)
```

Arguments

query	(optional) Advanced search query.
filters	(optional) Low-level filter specification.
entities	(optional) Low-level entity specification.
env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
# No filters are applied
GlobalSearch.request()

# Global Beacon search
GlobalSearch.request(entities = '[["gene","BRCA2"]]')

# Type filter (only vaults)
GlobalSearch.request(filters = '[{"and":[["type__in",["vault"]]]}]')

# Advanced search
GlobalSearch.request(query = "fuji")


# Multiple filters and entities
GlobalSearch.request(
  entities = '[["gene","BRCA2"]]',
  filters = '[{
               "and": [
                      {"and": [
                         ["created_at__range",["2021-11-28","2021-12-28"]]]},
                         ["type__in",["dataset"]]
                      ]
             }]'
)

## End(Not run)
```

GlobalSearch.search *GlobalSearch.search*

Description

Performs a Global Search based on provided filters, entities, queries, and returns an R data frame containing results from API response. Returns a single page of results otherwise (default).

Usage

```
GlobalSearch.search(paginate = FALSE, env = solvebio:::.solveEnv, ...)
```

Arguments

paginate	When set to TRUE, retrieves all records (memory permitting).
env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. filters, entities, query, limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
# No filters applied
GlobalSearch.search()

#Global Beacon search
GlobalSearch.search(entities = '[["gene","BRCA2"]]')

GlobalSearch.search(filters = '[{"and":[["type__in",["vault"]]]}]')

# Advanced search
GlobalSearch.search(query = "fuji")

## End(Not run)
```

GlobalSearch.subjects *GlobalSearch.subjects*

Description

Performs a Global Search based on provided filters, entities, queries, and returns an R data frame containing subjects from API response.

Usage

```
GlobalSearch.subjects(env = solvebio:::.solveEnv, ...)
```

Arguments

env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. filters, entities, query, limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
GlobalSearch.subjects(entities = '[["gene","BRCA2"]]')

## End(Not run)
```

GlobalSearch.subjects_count *GlobalSearch.subjects_count*

Description

Performs a Global Search based on provided filters, entities, queries, and returns the total number of subjects from API response.

Usage

```
GlobalSearch.subjects_count(env = solvebio:::.solveEnv, ...)
```

Arguments

env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. filters, entities, query, limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
GlobalSearch.subjects_count(entities = '[["gene","BRCA2"]]')

## End(Not run)
```

---

login *login*

Description

Store and verify your SolveBio credentials.

Usage

```
login(api_key, api_host, env = solvebio:::.solveEnv)
```

Arguments

api_key	Your SolveBio API key
api_host	SolveBio API host (default: https://api.solvebio.com)
env	(optional) The R environment used to store API credentials.

References

https://docs.solvebio.com/

Examples

```
## Not run:
login()

## End(Not run)
```

Object.all *Object.all*

Description

Retrieves the metadata about all objects on SolveBio accessible to the current user.

Usage

```
Object.all(env = solvebio:::.solveEnv, ...)
```

Arguments

env	(optional) Custom client environment.
...	(optional) Additional query parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.all()

## End(Not run)
```

Object.create *Object.create*

Description

Create a SolveBio object.

Usage

```
Object.create(
  vault_id,
  parent_object_id,
  object_type,
  filename,
  env = solvebio:::.solveEnv,
  ...
)
```

Arguments

vault_id	The target vault ID.
parent_object_id	The ID of the parent object (folder) or NULL for the vault root.
object_type	The type of object (i.e. "folder").
filename	The filename (i.e. the name) of the object.
env	(optional) Custom client environment.
...	(optional) Additional object attributes.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.create(
              vault_id="1234567890",
              parent_object_id=NULL,
              object_type="folder",
              filename="My Folder"
              )

## End(Not run)
```

Object.data *Object.data*

Description

Returns one page of documents from a SolveBio file (object) and processes the response.

Usage

```
Object.data(id, filters, col.names = NULL, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of a SolveBio file (vault object).
filters	(optional) Query filters.
col.names	(optional) Force data frame column name ordering.
env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.data("1234567890")

## End(Not run)
```

`Object.delete` *Object.delete*

Description

Delete a specific object from SolveBio.

Usage

```
Object.delete(id, env = solvebio:::.solveEnv)
```

Arguments

`id`	The ID of the object.
`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.delete("1234567890")

## End(Not run)
```

`Object.disable_global_beacon`
Object.disable_global_beacon

Description

Disables Global Beacon for the specified dataset.

Usage

```
Object.disable_global_beacon(id, env = solvebio:::.solveEnv)
```

Arguments

`id`	The ID of a SolveBio dataset.
`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.disable_global_beacon("1234567890")

## End(Not run)
```

`Object.enable_global_beacon`
Object.enable_global_beacon

Description

Enables Global Beacon for the specified dataset.

Usage

```
Object.enable_global_beacon(id, env = solvebio:::.solveEnv)
```

Arguments

`id`	The ID of a SolveBio dataset.
`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.enable_global_beacon("1234567890")

## End(Not run)
```

`Object.fields` *Object.fields*

Description

Retrieves the list of fields for a file (JSON, CSV, or TSV).

Usage

```
Object.fields(id, env = solvebio:::.solveEnv, ...)
```

Arguments

`id`	The ID of a SolveBio file (vault object).
`env`	(optional) Custom client environment.
`...`	(optional) Additional query parameters (e.g. limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.fields("1234567890")

## End(Not run)
```

Object.get_by_full_path *Object.get_by_full_path*

Description

A helper function to get an object on SolveBio by its full path.

Usage

```
Object.get_by_full_path(full_path, env = solvebio:::.solveEnv, ...)
```

Arguments

full_path	The full path to the object.
env	(optional) Custom client environment.
...	(optional) Additional query parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.get_by_full_path("solvebio:public:/ClinVar")

## End(Not run)
```

Object.get_by_path *Object.get_by_path*

Description

A helper function to get an object on SolveBio by its path. Used as a pass-through function from some Vault methods.

Usage

```
Object.get_by_path(path, env = solvebio:::.solveEnv, ...)
```

Arguments

path	The path to the object, relative to a vault.
env	(optional) Custom client environment.
...	(optional) Additional query parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.get_by_path("/ClinVar")

## End(Not run)
```

`Object.get_download_url` *Object.get_download_url*

Description

Helper method to get the download URL for a file object.

Usage

```
Object.get_download_url(id, env = solvebio:::.solveEnv)
```

Arguments

`id`	The ID of the object.
`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.get_download_url("1234567890")

## End(Not run)
```

Object.get_global_beacon_status *Object.get_global_beacon_status*

Description

Retrieves the global beacon status for the specified dataset.

Usage

```
Object.get_global_beacon_status(
  id,
  raise_on_disabled = FALSE,
  env = solvebio:::.solveEnv
)
```

Arguments

id	The ID of a SolveBio dataset.
raise_on_disabled	Whether to raise an exception if Global Beacon is disabled or to return NULL.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.get_global_beacon_status("1234567890")
Object.get_global_beacon_status("1234567890", raise_on_disabled=TRUE)

## End(Not run)
```

Object.get_or_upload_file *Object.get_or_upload_file*

Description

Upload a local file to a vault on SolveBio only if it does not yet exist (by name, at the provided path). The vault path provided is the parent directory for uploaded file. Accepts the same arguments as 'Object.upload_file'.

Usage

```
Object.get_or_upload_file(
  local_path,
  vault_id,
  vault_path,
  filename,
  env = solvebio:::.solveEnv
)
```

Arguments

local_path	The path to the local file
vault_id	The SolveBio vault ID
vault_path	The remote path in the vault
filename	(optional) The filename for the uploaded file in the vault (default: the basename of the local_path)
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.get_or_upload_file("my_file.json.gz", vault$id, "/parent/directory/")

## End(Not run)
```

Object.query *Object.query*

Description

Queries a SolveBio file (vault object) and returns an R data frame containing all records. Returns a single page of results otherwise (default).

Usage

```
Object.query(id, paginate = FALSE, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of a SolveBio file (vault object).
paginate	When set to TRUE, retrieves all records (memory permitting).
env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. filters, limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.query("12345678790", paginate=TRUE)

## End(Not run)
```

`Object.retrieve` *Object.retrieve*

Description

Retrieves the metadata about a specific object on SolveBio.

Usage

```
Object.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

`id`	The ID of the object.
`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.retrieve("1234567890")

## End(Not run)
```

`Object.update` *Object.update*

Description

Updates the attributes of an existing vault object.

Usage

```
Object.update(id, env = solvebio:::.solveEnv, ...)
```

Arguments

`id`	The ID of the vault to update.
`env`	(optional) Custom client environment.
`...`	Object attributes to change.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.update(
              id="1234",
              filename="New Name",
             )

## End(Not run)
```

`Object.upload_file` *Object.upload_file*

Description

Upload a local file to a vault on SolveBio. The vault path provided is the parent directory for uploaded file.

Usage

```
Object.upload_file(
  local_path,
  vault_id,
  vault_path,
  filename,
  env = solvebio:::.solveEnv
)
```

Arguments

local_path	The path to the local file
vault_id	The SolveBio vault ID
vault_path	The remote path in the vault
filename	(optional) The filename for the uploaded file in the vault (default: the basename of the local_path)
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Object.upload_file("my_file.json.gz", vault$id, "/parent/directory/")

## End(Not run)
```

---

protectedServer *protectedServer*

Description

Wraps an existing Shiny server in an OAuth2 flow.

Usage

```
protectedServer(
  server,
  client_id,
  client_secret = NULL,
  base_url = "https://my.solvebio.com"
)
```

Arguments

server	Your original Shiny server function.
client_id	Your application's client ID.
client_secret	(optional) Your application's client secret.
base_url	(optional) Override the default login host (default: https://my.solvebio.com).

References

https://docs.solvebio.com/

Examples

```
## Not run:
protectedServer(
                server=server,
                client_id="abcd1234"
                )

## End(Not run)
```

---

protectedServerJS *protectedServerUI*

Description

Returns ShinyJS-compatible JS code to support cookie-based token storage.

Usage

```
protectedServerJS()
```

References

https://docs.solvebio.com/

Examples

```
## Not run:
jscookie_src <- "https://cdnjs.cloudflare.com/ajax/libs/js-cookie/2.2.0/js.cookie.js"
ui <- fluidPage(
    shiny::tags$head(
        shiny::tags$script(src = jscookie_src)
    ),
    useShinyjs(),
    extendShinyjs(text = solvebio::protectedServerJS(),
                functions = c("enableCookieAuth", "getCookie", "setCookie", "rmCookie"))
)
```

```
## End(Not run)
```

SavedQuery.all *SavedQuery.all*

Description

Retrieves the all saved queries on SolveBio.

Usage

```
SavedQuery.all(env = solvebio:::.solveEnv, ...)
```

Arguments

env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. page).

References

https://docs.solvebio.com/

Examples

```
## Not run:
SavedQuery.all()

## End(Not run)
```

SavedQuery.create *SavedQuery.create*

Description

Create a SolveBio saved query.

Usage

```
SavedQuery.create(env = solvebio:::.solveEnv, ...)
```

Arguments

env	(optional) Custom client environment.
...	(optional) Saved query attributes.

References

https://docs.solvebio.com/

Examples

```
## Not run:
SavedQuery.create(name="My Dataset Template")

## End(Not run)
```

SavedQuery.delete *SavedQuery.delete*

Description

Deletes a specific saved query on SolveBio.

Usage

```
SavedQuery.delete(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of the SolveBio saved query.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
SavedQuery.delete(<ID>)

## End(Not run)
```

SavedQuery.retrieve *SavedQuery.retrieve*

Description

Retrieves a specific saved query on SolveBio by ID.

Usage

```
SavedQuery.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

id	String The ID of a SolveBio saved query.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
SavedQuery.retrieve(<ID>)

## End(Not run)
```

SavedQuery.update *SavedQuery.update*

Description

Updates the attributes of an existing saved query.

Usage

```
SavedQuery.update(id, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of the saved query to update.
env	(optional) Custom client environment.
...	Saved query attributes to change.

References

https://docs.solvebio.com/

Examples

```
## Not run:
SavedQuery.update(
               id="1234",
               name="New query Name",
              )

## End(Not run)
```

`Task.all` *Task.all*

Description

Retrieves the metadata about all tasks on SolveBio accessible to the current user.

Usage

```
Task.all(env = solvebio:::.solveEnv, ...)
```

Arguments

`env`	(optional) Custom client environment.
`...`	(optional) Additional query parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Task.all()

## End(Not run)
```

Task.follow *Task.follow*

Description

A helper function to follow a specific task until it gets completed.

Usage

```
Task.follow(id, env = solvebio:::.solveEnv, interval = 2)
```

Arguments

id	String The ID of a task.
env	(optional) Custom client environment.
interval	2. Delay in seconds between each completion status query

Value

the task object

References

https://docs.solvebio.com/

Examples

```
## Not run:
Task.follow("1234567890")

## End(Not run)
```

Task.retrieve *Task.retrieve*

Description

Retrieves the metadata about a specific task on SolveBio.

Usage

```
Task.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

id	The ID of a task.
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Task.retrieve("1234567890")

## End(Not run)
```

`User.retrieve` *User.retrieve*

Description

Retrieves information about the current user.

Usage

```
User.retrieve(env = solvebio:::.solveEnv)
```

Arguments

`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
User.retrieve()

## End(Not run)
```

Vault.all *Vault.all*

Description

Retrieves the metadata about all accessible vaults.

Usage

```
Vault.all(..., env = solvebio:::.solveEnv)
```

Arguments

...	(optional) Additional query parameters (e.g. limit, offset).
env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Vault.all()

## End(Not run)
```

Vault.create *Vault.create*

Description

Create a new SolveBio vault.

Usage

```
Vault.create(name, env = solvebio:::.solveEnv, ...)
```

Arguments

name	The unique name of the vault.
env	(optional) Custom client environment.
...	(optional) Additional vault attributes.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Vault.create(name="my-domain:MyVault")

## End(Not run)
```

`Vault.create_dataset` *Vault.create_dataset*

Description

Create a new dataset in a vault.

Usage

```
Vault.create_dataset(id, path, name, env = solvebio:::.solveEnv, ...)
```

Arguments

`id`	The ID of the vault.
`path`	The path to the dataset, within the vault.
`name`	The name (filename) for the dataset.
`env`	(optional) Custom client environment.
`...`	(optional) Additional dataset creation parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
vault = Vault.get_personal_vault()
Vault.create_dataset(vault$id, path="/", name="My Dataset")

## End(Not run)
```

Vault.create_folder *Vault.create_folder*

Description

Create a new folder in a vault.

Usage

```
Vault.create_folder(
  id,
  path,
  recursive = FALSE,
  env = solvebio:::.solveEnv,
  ...
)
```

Arguments

id	The ID of the vault.
path	The path to the folder, within the vault.
recursive	Create all parent directories that do not yet exist (default: FALSE).
env	(optional) Custom client environment.
...	(optional) Additional folder creation parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
vault = Vault.get_personal_vault()
Vault.create_folder(vault$id, "/My Folder")

## End(Not run)
```

`Vault.datasets` *Vault.datasets*

Description

Retrieves all datasets in a specific vault.

Usage

```
Vault.datasets(id, env = solvebio:::.solveEnv, ...)
```

Arguments

`id`	The ID of the vault.
`env`	(optional) Custom client environment.
`...`	(optional) Additional query parameters (e.g. limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
vault = Vault.get_personal_vault()
Vault.datasets(vault$id)

## End(Not run)
```

`Vault.delete` *Vault.delete*

Description

Delete a specific vault from SolveBio. This operation cannot be undone.

Usage

```
Vault.delete(id, env = solvebio:::.solveEnv)
```

Arguments

`id`	String The ID of a SolveBio vault.
`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Vault.delete("1")

## End(Not run)
```

`Vault.files` *Vault.files*

Description

Retrieves all files in a specific vault.

Usage

```
Vault.files(id, env = solvebio:::.solveEnv, ...)
```

Arguments

`id`	The ID of the vault.
`env`	(optional) Custom client environment.
`...`	(optional) Additional query parameters (e.g. limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
vault = Vault.get_personal_vault()
Vault.files(vault$id)

## End(Not run)
```

`Vault.folders` *Vault.folders*

Description

Retrieves all folders in a specific vault.

Usage

```
Vault.folders(id, env = solvebio:::.solveEnv, ...)
```

Arguments

`id`	The ID of the vault.
`env`	(optional) Custom client environment.
`...`	(optional) Additional query parameters (e.g. limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
vault = Vault.get_personal_vault()
Vault.folders(vault$id)

## End(Not run)
```

`Vault.get_by_full_path` *Vault.get_by_full_path*

Description

Retrieves a specific vault by its full path (domain:vault).

Usage

```
Vault.get_by_full_path(full_path, verbose = TRUE, env = solvebio:::.solveEnv)
```

Arguments

`full_path`	The full path of a SolveBio vault.
`verbose`	Print warning/error messages (default: TRUE).
`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Vault.get_by_full_path("SolveBio:Public")

## End(Not run)
```

`Vault.get_or_create_by_full_path`
Vault.get_or_create_by_full_path

Description

Retrieves or creates a specific vault by its full path (domain:vault).

Usage

```
Vault.get_or_create_by_full_path(full_path, env = solvebio:::.solveEnv, ...)
```

Arguments

`full_path`	The full path of a SolveBio vault.
`env`	(optional) Custom client environment.
`...`	(optional) Additional parameters.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Vault.get_or_create_by_full_path("My New Vault")

## End(Not run)
```

Vault.get_personal_vault *Vault.get_personal_vault*

Description

Retrieves the current users"s personal, private vault.

Usage

```
Vault.get_personal_vault(env = solvebio:::.solveEnv)
```

Arguments

env	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Vault.get_personal_vault()

## End(Not run)
```

Vault.objects *Vault.objects*

Description

Retrieves all objects in a specific vault.

Usage

```
Vault.objects(id, env = solvebio:::.solveEnv, ...)
```

Arguments

id	The ID of the vault.
env	(optional) Custom client environment.
...	(optional) Additional query parameters (e.g. limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
vault = Vault.get_personal_vault()
Vault.objects(vault$id)

## End(Not run)
```

`Vault.retrieve` *Vault.retrieve*

Description

Retrieves the metadata about a specific SolveBio vault.

Usage

```
Vault.retrieve(id, env = solvebio:::.solveEnv)
```

Arguments

`id`	String The ID of a SolveBio vault
`env`	(optional) Custom client environment.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Vault.retrieve("1234567890")

## End(Not run)
```

`Vault.search` *Vault.search*

Description

Search for objects in a specific vault.

Usage

```
Vault.search(id, query, env = solvebio:::.solveEnv, ...)
```

Arguments

`id`	The ID of the vault.
`query`	The search query.
`env`	(optional) Custom client environment.
`...`	(optional) Additional query parameters (e.g. limit, offset).

References

https://docs.solvebio.com/

Examples

```
## Not run:
vault = Vault.get_personal_vault()
Vault.search('test')

## End(Not run)
```

`Vault.update` *Vault.update*

Description

Updates the attributes of an existing vault.

Usage

```
Vault.update(id, env = solvebio:::.solveEnv, ...)
```

Arguments

`id`	The ID of the vault to update.
`env`	(optional) Custom client environment.
`...`	Vault attributes to change.

References

https://docs.solvebio.com/

Examples

```
## Not run:
Vault.update(
             id="1234",
             name="New Vault Name",
            )

## End(Not run)
```

Index

Annotator.annotate, 4
Application.all, 5
Application.create, 5
Application.delete, 6
Application.retrieve, 7
Application.update, 7

Beacon.all, 8
Beacon.create, 9
Beacon.delete, 10
Beacon.query, 10
Beacon.retrieve, 11
Beacon.update, 12
BeaconSet.all, 12
BeaconSet.create, 13
BeaconSet.delete, 14
BeaconSet.query, 15
BeaconSet.retrieve, 15
BeaconSet.update, 16

createEnv, 17

Dataset.activity, 17
Dataset.all, 18
Dataset.count, 19
Dataset.create, 19
Dataset.data, 20
Dataset.delete, 21
Dataset.disable_global_beacon, 21
Dataset.enable_global_beacon, 22
Dataset.facets, 23
Dataset.fields, 23
Dataset.get_by_full_path, 24
Dataset.get_global_beacon_status, 25
Dataset.get_or_create_by_full_path, 25
Dataset.query, 26
Dataset.retrieve, 27
Dataset.template, 27
Dataset.update, 28
DatasetCommit.all, 29
DatasetCommit.delete, 29
DatasetCommit.retrieve, 30
DatasetExport.all, 31
DatasetExport.create, 31
DatasetExport.delete, 32
DatasetExport.get_download_url, 33
DatasetExport.retrieve, 33
DatasetField.all, 34
DatasetField.create, 35
DatasetField.facets, 36
DatasetField.retrieve, 36
DatasetField.update, 37
DatasetImport.all, 38
DatasetImport.create, 38
DatasetImport.delete, 39
DatasetImport.retrieve, 40
DatasetMigration.all, 40
DatasetMigration.create, 41
DatasetMigration.delete, 42
DatasetMigration.retrieve, 43
DatasetTemplate.all, 43
DatasetTemplate.create, 44
DatasetTemplate.delete, 45
DatasetTemplate.retrieve, 45
DatasetTemplate.update, 46

Expression.evaluate, 47

GlobalSearch.facets, 48
GlobalSearch.request, 48
GlobalSearch.search, 50
GlobalSearch.subjects, 51
GlobalSearch.subjects_count, 51

login, 52

Object.all, 53
Object.create, 53
Object.data, 54
Object.delete, 55

Object.disable_global_beacon, 56
Object.enable_global_beacon, 56
Object.fields, 57
Object.get_by_full_path, 58
Object.get_by_path, 58
Object.get_download_url, 59
Object.get_global_beacon_status, 60
Object.get_or_upload_file, 60
Object.query, 61
Object.retrieve, 62
Object.update, 63
Object.upload_file, 63

protectedServer, 64
protectedServerJS, 65

SavedQuery.all, 66
SavedQuery.create, 66
SavedQuery.delete, 67
SavedQuery.retrieve, 68
SavedQuery.update, 68

Task.all, 69
Task.follow, 70
Task.retrieve, 70

User.retrieve, 71

Vault.all, 72
Vault.create, 72
Vault.create_dataset, 73
Vault.create_folder, 74
Vault.datasets, 75
Vault.delete, 75
Vault.files, 76
Vault.folders, 77
Vault.get_by_full_path, 77
Vault.get_or_create_by_full_path, 78
Vault.get_personal_vault, 79
Vault.objects, 79
Vault.retrieve, 80
Vault.search, 81
Vault.update, 81