

Package: databraryr (via r-universe)

July 2, 2024

Type Package

Title Interact with the 'Databrary.org' API

Version 0.6.5

Maintainer Rick O. Gilmore <rog1@psu.edu>

Description 'Databrary.org' is a restricted access repository for research data, especially video and audio. This package provides commands to interact with the data stored on 'Databrary.org'.

License MIT + file LICENSE

URL <https://databrary.github.io/databraryr/>,
<https://github.com/databrary/databraryr>

BugReports <https://github.com/databrary/databraryr/issues>

Imports assertthat (>= 0.2.1), dplyr (>= 1.1.4), getPass (>= 0.2.4),
httr (>= 1.4.7), httr2 (>= 1.0.0), jsonlite (>= 1.8.8), keyring
(>= 1.3.2), lifecycle (>= 1.0.4), magick (>= 2.8.2), magrittr,
methods (>= 4.3.2), purrr (>= 1.0.2), stringr (>= 1.5.1),
tibble (>= 3.2.1), xfun (>= 0.41)

Suggests av (>= 0.9.0), ggplot2, knitr, openssl, readr, rmarkdown (>= 2.26), testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

RoxygenNote 7.3.1

NeedsCompilation no

Author Rick O. Gilmore [aut, cre, cph], Jeffrey Spies [aut], National Science Foundation OAC-2032713 [fnd], National Institutes of Health R01HD094830 [fnd]

Repository CRAN

Date/Publication 2024-07-01 19:00:02 UTC

Contents

API_CONSTANTS	3
assign_constants	3
check_ssl_certs	4
download_party	5
download_party_avatar	5
download_session_asset	6
download_session_assets_fr_df	7
download_session_csv	8
download_session_zip	9
download_single_session_asset_fr_df	10
download_video	12
download_volume_zip	13
get_assets_from_session	14
get_asset_segment_range	14
GET_db_contents	15
get_db_stats	16
get_file_duration	17
get_party_as_df	17
get_party_by_id	18
get_permission_levels	19
get_release_levels	20
get_session_by_id	20
get_session_by_name	21
get_supported_file_types	22
get_volume_by_id	22
HHMMSSmmm_to_ms	23
is_institution	24
is_person	24
list_affiliates	25
list_asset_formats	26
list_authorized_investigators	26
list_individual_sponsors	27
list_institutional_sponsors	28
list_party	28
list_party_affiliates	29
list_party_sponsors	30
list_party_volumes	30
list_people	31
list_sessions	32
list_sessions_in_volume	32
list_session_activity	33
list_session_assets	34
list_sponsors	34
list_volume	35
list_volume_activity	36
list_volume_assets	36

list_volume_excerpts	37
list_volume_funding	38
list_volume_links	38
list_volume_owners	39
list_volume_sessions	40
list_volume_session_assets	41
list_volume_tags	42
login_db	42
logout_db	43
make_default_request	44
make_fn_portable	44
make_login_client	45
search_for_funder	46
search_for_keywords	47
search_for_tags	47
Index	49

API_CONSTANTS	<i>Load Package-wide Constants into Local Environment</i>
---------------	---

Description

Load Package-wide Constants into Local Environment

Usage

API_CONSTANTS

Format

An object of class character of length 1.

assign_constants	<i>Download Databrary Constants From API.</i>
------------------	---

Description

Download Databrary Constants From API.

Usage

assign_constants(vb = FALSE, rq = NULL)

Arguments

- vb A Boolean value. If TRUE provides verbose output.
- rq An httr2 request object. Defaults to NULL.

Value

A data frame with the constants.

Examples

```
assign_constants()
```

check_ssl_certs *Check SSL Certificates For nyu.databrary.org.*

Description

check_ssl_certs checks the SSL certificates for nyu.databrary.org and returns a data frame with the relevant information.

Usage

```
check_ssl_certs(host = "nyu.databrary.org")
```

Arguments

- host Target URL. Defaults to 'nyu.databrary.org'.

Value

A data frame with information about the SSL certificates.

Examples

```
check_ssl_certs()
```

download_party	<i>Download Information About a Party on Databrary.</i>
----------------	---

Description

[Deprecated]

This function has been deprecated and may be removed in a future release. See `get_party_by_id()` for similar, and more general functionality.

Usage

```
download_party(party_id = 6, convert_JSON = TRUE, vb = FALSE)
```

Arguments

party_id	Party number to retrieve information about.
convert_JSON	A Boolean value if TRUE converts the JSON download
vb	A Boolean value if TRUE returns verbose output.

Value

A data frame with information about the party.

Examples

```
download_party()  
#'
```

download_party_avatar	<i>Returns the Avatar(s) (images) for Authorized User(s).</i>
-----------------------	---

Description

Returns the Avatar(s) (images) for Authorized User(s).

Usage

```
download_party_avatar(  
  party_id = 6,  
  show_party_info = TRUE,  
  vb = FALSE,  
  rq = NULL  
)
```

Arguments

party_id	A number or range of numbers. Party number or numbers to retrieve information about. Default is 6 (Rick Gilmore).
show_party_info	A logical value. Show the person's name and affiliation in the output. Default is TRUE.
vb	A Boolean value. If TRUE returns verbose output. Default is FALSE.
rq	An http2 request object. If not provided, a new request is generated via make_default_request().

Value

An list with the avatar (image) file and a name_affil string.

Examples

```
## Not run:
download_party_avatar() # Show Rick Gilmore's (party 6) avatar.

# Download avatars from Databrary's founders (without name/affiliations)
download_party_avatar(5:7, show_party_info = FALSE)

# Download NYU logo
download_party_avatar(party = 8)

## End(Not run)
```

download_session_asset

Download Asset From Databrary.

Description

Databrary stores file types (assets) of many types. This function downloads an asset based on its system-unique integer identifier (asset_id) and system-unique session (slot) identifier (session_id).

Usage

```
download_session_asset(
  asset_id = 1,
  session_id = 9807,
  file_name = NULL,
  target_dir = tempdir(),
  timeout_secs = REQUEST_TIMEOUT,
  vb = FALSE,
  rq = NULL
)
```

Arguments

asset_id	An integer. Asset id for target file. Default is 1.
session_id	An integer. Slot/session number where target file is stored. Default is 9807.
file_name	A character string. Name for downloaded file. Default is NULL.
target_dir	A character string. Directory to save the downloaded file. Default is a temporary directory given by a call to tempdir().
timeout_secs	An integer constant. The default value, defined in CONSTANTS.R is REQUEST_TIMEOUT. This value determines the default timeout value for the httr2 request object. When downloading large files, it can be useful to set this value to a large number.
vb	A logical value. If TRUE provides verbose output. Default is FALSE.
rq	A list in the form of an httr2 request object. Default is NULL.

Value

Full file name to the asset or NULL.

Examples

```
## Not run:
download_session_asset() # Download's 'numbers' file from volume 1.
download_session_asset(asset_id = 11643, session_id = 9825, file_name = "rdk.mp4")
# Downloads a display with a random dot kinematogram (RDK).

## End(Not run)
```

download_session_assets_fr_df

Download Asset From A Databrary Session.

Description

Databrary stores file types (assets) of many types. This function downloads assets in a data frame generated by list_session_assets().

Usage

```
download_session_assets_fr_df(
  session_df = list_session_assets(),
  target_dir = tempdir(),
  add_session_subdir = TRUE,
  overwrite = TRUE,
  make_portable_fn = FALSE,
  timeout_secs = REQUEST_TIMEOUT_VERY_LONG,
  vb = FALSE,
  rq = NULL
)
```

Arguments

session_df	A data frame as generated by list_session_assets_2().
target_dir	A character string. Directory to save the downloaded file. Default is directory named after the session_id.
add_session_subdir	A logical value. Add add the session name to the file path so that files are in a subdirectory specific to the session. Default is TRUE.
overwrite	A logical value. Overwrite an existing file. Default is TRUE.
make_portable_fn	A logical value. Replace characters in file names that are not broadly portable across file systems. Default is FALSE.
timeout_secs	An integer. The seconds an httr2 request will run before timing out. Default is 600 (10 min). This is to handle very large files.
vb	A logical value. If TRUE provides verbose output. Default is FALSE.
rq	A list in the form of an httr2 request object. Default is NULL.

Value

Full file names to the downloaded assets or NULL.

Examples

```
## Not run:
download_session_assets_fr_df() # Downloads all of the files from session
9807 in Databrary volume 1.

# Just the CSVs
v1 <- list_session_assets()
v1_csv <- dplyr::filter(v1, format_extension == "csv")
download_session_assets_fr_df(v1_csv, vb = TRUE)

## End(Not run)
```

download_session_csv *Download Session Spreadsheet As CSV*

Description

Databrary generates a CSV-formatted spreadsheet that summarizes information about individual sessions. This command downloads that CSV file as a temporary file or with a name specified by the user.

Usage

```
download_session_csv(
  vol_id = 1,
  file_name = "test.csv",
  target_dir = tempdir(),
  as_df = FALSE,
  vb = FALSE,
  rq = NULL
)
```

Arguments

vol_id	An integer. Target volume number. Default is 1.
file_name	A character string. Name for the output file. Default is 'test.csv'.
target_dir	A character string. Directory to save downloaded file. Default is tempdir().
as_df	A logical value. Convert the data from a list to a data frame. Default is FALSE.
vb	A logical value. Provide verbose feedback. Default is FALSE.
rq	An httr2 request object. Default is NULL.

Value

A character string that is the name of the downloaded file or a data frame if as_df is TRUE.

Examples

```
## Not run:
download_session_csv() # Downloads "session" CSV for volume 1

## End(Not run)
```

download_session_zip *Download Zip Archive From Databrary Session.*

Description

Download Zip Archive From Databrary Session.

Usage

```
download_session_zip(
  vol_id = 31,
  session_id = 9803,
  out_dir = tempdir(),
  file_name = "test.zip",
```

```

    vb = FALSE,
    rq = NULL
  )

```

Arguments

vol_id	Volume number.
session_id	Slot/session number.
out_dir	Directory to save output file.
file_name	Name for downloaded file, default is 'test.zip'.
vb	A Boolean value. If TRUE provides verbose output.
rq	An <code>httr2</code> request object. Default is NULL.

Value

Full filename of the downloaded file.

Examples

```

## Not run:
download_session_zip() # Downloads Zip Archive from volume 31, session 9803

## End(Not run)

```

download_single_session_asset_fr_df

Download Single Asset From Databrary

Description

Databrary stores file types (assets) of many types. This function downloads an asset based on its system-unique integer identifier (`asset_id`) and system-unique session (slot) identifier (`session_id`). It is designed to work with `download_session_assets_fr_df()` so that multiple files can be downloaded simultaneously.

Usage

```

download_single_session_asset_fr_df(
  i = NULL,
  session_df = NULL,
  target_dir = tempdir(),
  add_session_subdir = TRUE,
  overwrite = TRUE,
  make_portable_fn = FALSE,

```

```

    timeout_secs = REQUEST_TIMEOUT_VERY_LONG,
    vb = FALSE,
    rq = NULL
  )

```

Arguments

<code>i</code>	An integer. Index into a row of the session asset data frame. Default is NULL.
<code>session_df</code>	A row from a data frame from <code>list_session_assets()</code> or <code>list_volume_assets()</code> . Default is NULL.
<code>target_dir</code>	A character string. Directory to save the downloaded file. Default is a temporary directory given by a call to <code>tempdir()</code> .
<code>add_session_subdir</code>	A logical value. Add the session name to the file path so that files are in a subdirectory specific to the session. Default is TRUE.
<code>overwrite</code>	A logical value. Overwrite an existing file. Default is TRUE.
<code>make_portable_fn</code>	A logical value. Replace characters in file names that are not broadly portable across file systems. Default is FALSE.
<code>timeout_secs</code>	An integer. The seconds an http2 request will run before timing out. Default is 600 (10 min). This is to handle very large files.
<code>vb</code>	A logical value. If TRUE provides verbose output. Default is FALSE.
<code>rq</code>	A list in the form of an http2 request object. Default is NULL.

Value

Full file name to the asset or NULL.

Examples

```

## Not run:
vol_1 <- list_session_assets(session_id = 9807)
a_1 <- vol_1[1,]
tmp_dir <- tempdir()
fn <- file.path(tmp_dir, paste0(a_1$asset_name, ".", a_1$format_extension))
download_single_session_asset_fr_df(a_1$asset_id,
  fn,
  session_id = a_1$session_id,
  vb = TRUE)

## End(Not run)

```

download_video	<i>Download Video From Databrary.</i>
----------------	---------------------------------------

Description

Download Video From Databrary.

Usage

```
download_video(
  asset_id = 1,
  session_id = 9807,
  file_name = tempfile(paste0(session_id, "_", asset_id, "_"), fileext = ".mp4"),
  target_dir = tempdir(),
  vb = FALSE,
  rq = NULL
)
```

Arguments

asset_id	Asset id for target file.
session_id	Slot/session number where target file is stored.
file_name	Name for downloaded file.
target_dir	Directory to save the downloaded file. Default is a temporary directory given by a call to tempdir().
vb	A Boolean value. If TRUE provides verbose output.
rq	An httr2 request object.

Value

Full file name to the asset.

Examples

```
## Not run:
download_video() # Download's 'numbers' file from volume 1.
download_video(asset_id = 11643, session_id = 9825, file_name = "rdk.mp4")
#' # Downloads a display with a random dot kinematogram (RDK).

## End(Not run)
```

download_volume_zip *Download Zip Archive of All Data in a Volume.*

Description

Download Zip Archive of All Data in a Volume.

Usage

```
download_volume_zip(  
  vol_id = 31,  
  out_dir = tempdir(),  
  file_name = "test.zip",  
  vb = FALSE,  
  rq = NULL  
)
```

Arguments

vol_id	Volume number.
out_dir	Directory to save output file.
file_name	Name for downloaded file, default is 'test.mp4'.
vb	A Boolean value. If TRUE provides verbose output.
rq	An httr2 request object. Default is NULL.

Value

Full filename of the downloaded file.

Examples

```
## Not run:  
download_volume_zip() # Zip file of all data from volume 31, the default.  
  
## End(Not run)
```

get_assets_from_session

Helper function for list_volume_assets

Description

Helper function for list_volume_assets

Usage

```
get_assets_from_session(volume_container, ignore_materials = TRUE)
```

Arguments

volume_container
The 'container' list from a volume.

ignore_materials
A logical value.

get_asset_segment_range

Get Time Range For An Asset.

Description

Get Time Range For An Asset.

Usage

```
get_asset_segment_range(  
  vol_id = 1,  
  session_id = 9807,  
  asset_id = 1,  
  convert_JSON = TRUE,  
  segment_only = TRUE,  
  vb = FALSE,  
  rq = NULL  
)
```

Arguments

vol_id	Volume ID
session_id	Slot/session number.
asset_id	Asset number.
convert_JSON	A Boolean value. If TRUE, convert JSON to a data frame. Default is TRUE.
segment_only	A Boolean value. If TRUE, returns only the segment values. Otherwise returns a data frame with two fields, segment and permission. Default is TRUE.
vb	A Boolean value. If TRUE provides verbose output.
rq	An httr2 request object. Default is NULL.

Value

The time range (in ms) for an asset, if one is indicated.

Examples

```
get_asset_segment_range()
```

GET_db_contents

Queries the Databrary API with an HTML GET command

Description

GET_db_contents is a helper function that retrieves information from the 'Databrary.org' API in a structured way.

[Deprecated]

This function has been deprecated and may be removed in a future version.

Usage

```
GET_db_contents(  
  base_URL = "https://nyu.databrary.org",  
  URL_components = "/api/volume/1",  
  convert_JSON = TRUE,  
  vb = FALSE  
)
```

Arguments

base_URL	Base URL for API call. Default is https://nyu.databrary.org.
URL_components	Other components of API URL usually assembled using a paste0() command.
convert_JSON	A Boolean value. If TRUE, convert JSON to a data frame. Default is TRUE.
vb	A Boolean value. If TRUE provides verbose output.

Value

Output from GET command with the specified parameters, usually a string or a data frame.

Examples

```
GET_db_contents() # Returns a list with data from volume 1.
```

get_db_stats	<i>Get Stats About Databrary.</i>
--------------	-----------------------------------

Description

get_db_stats returns basic summary information about the institutions, people, and data hosted on 'Databrary.org'.

Usage

```
get_db_stats(type = "stats", vb = FALSE, rq = NULL)
```

Arguments

type	Type of Databrary report to run "institutions", "people", "data"
vb	A Boolean value. If TRUE provides verbose output.
rq	An httr2 request object.

Value

A data frame with the requested data or NULL if there is no new information.

Examples

```
get_db_stats()
get_db_stats("stats")
get_db_stats("people") # Information about the newest authorized investigators.
get_db_stats("places") # Information about the newest institutions.
```

get_file_duration *Get Duration (In ms) Of A File.*

Description

Get Duration (In ms) Of A File.

Usage

```
get_file_duration(  
  asset_id = 1,  
  types_w_durations = c("-600", "-800"),  
  vb = FALSE,  
  rq = NULL  
)
```

Arguments

asset_id Asset number.
types_w_durations Asset types that have valid durations.
vb A Boolean value. If TRUE provides verbose output.
rq An http2 request object. Default is NULL.

Value

Duration of a file in ms.

Examples

```
get_file_duration() # default is the test video from databrary.org/volume/1
```

get_party_as_df *Retrieve Party (Person/Institution) Info as Data Frame.*

Description

[Deprecated]

This function has been deprecated and may be removed in a future release. See `get_party_by_id()` for similar, and more general functionality.

Usage

```
get_party_as_df(party_id = 6, convert_JSON = TRUE, vb = FALSE)
```

Arguments

party_id	Party number(s) to retrieve information about.
convert_JSON	A Boolean value if TRUE converts the JSON download.
vb	A Boolean value if TRUE returns verbose output.

Value

A data frame with information about the selected party or parties.

Examples

```
get_party_as_df() # Info about Rick Gilmore.
get_party_as_df(party_id = 8) # Info about NYU.
get_party_as_df(5:7) # Info about Databrary's founders
#'
```

get_party_by_id *Download Information About a Party on Databrary as JSON*

Description

Download Information About a Party on Databrary as JSON

Usage

```
get_party_by_id(
  party_id = 6,
  parents_children_access = TRUE,
  vb = FALSE,
  rq = NULL
)
```

Arguments

party_id	An integer. The party number to retrieve information about.
parents_children_access	A logical value. If TRUE (the default), returns <i>all</i> of the data about the party. If FALSE, only a minimum amount of information about the party is returned.
vb	A logical value if TRUE returns verbose output.
rq	An <code>httr2</code> -style request object. If NULL, then a new request will be generated using <code>make_default_request()</code> .

Value

A nested list with information about the party. This can be readily parsed by other functions.

Examples

```
## Not run:  
get_party_by_id()  
  
## End(Not run)
```

get_permission_levels *Extract Databrary Permission Levels.*

Description

Extract Databrary Permission Levels.

Usage

```
get_permission_levels(vb = FALSE)
```

Arguments

vb A Boolean value. If TRUE provides verbose output.

Value

An array with the permission levels that can be assigned to data.

Examples

```
get_permission_levels()
```

get_release_levels *Show Databrary Release Levels*

Description

Show Databrary Release Levels

Usage

```
get_release_levels(vb = FALSE)
```

Arguments

vb A Boolean value. If TRUE provides verbose output.

Value

A data frame with Databrary's release levels.

Examples

```
get_release_levels()
```

get_session_by_id *Get Session (Slot) Data From A Databrary Volume*

Description

Get Session (Slot) Data From A Databrary Volume

Usage

```
get_session_by_id(session_id = 9807, vol_id = 1, vb = FALSE, rq = NULL)
```

Arguments

session_id An integer indicating a valid session/slot identifier linked to a volume. Default value is 9807, the materials folder for volume 1.

vol_id An integer indicating the volume identifier. Default is 1.

vb A logical value. Show verbose feedback. Default is FALSE.

rq An httr2 request object.

Value

A JSON blob with the session data. If the user has previously logged in to Databrary via `login_db()`, then session(s) that have restricted access can be downloaded, subject to the sharing release levels on those session(s).

Examples

```
## Not run:
get_session_by_id() # Default is Volume 1, Session 9807, the Materials folder

## End(Not run)
```

get_session_by_name *Get Session (Slot) Data From A Databrary Volume By Session Name.*

Description

Get Session (Slot) Data From A Databrary Volume By Session Name.

Usage

```
get_session_by_name(
  session_name = "Advisory Board Meeting",
  vol_id = 1,
  vb = FALSE,
  rq = NULL
)
```

Arguments

session_name	A string. The name of the target session. Defaults to "Advisory Board Meeting", the name of several of the sessions in the public Volume 1.
vol_id	An integer indicating the volume identifier. Default is 1.
vb	A logical value. Show verbose feedback. Default is FALSE.
rq	An htr2 request, such as that generated by <code>make_default_request()</code> .

Value

One or more JSON blobs (as lists) whose session name(s) match name in the given volume.

Examples

```
## Not run:  
# Returns multiple JSON blobs with the name "Advisory Board Meeting" in Volume 1.  
get_session_by_name()  
  
get_session_by_name(name = 'Top-level materials')  
  
## End(Not run)
```

get_supported_file_types

Extracts File Types Supported by Databrary.

Description

Extracts File Types Supported by Databrary.

Usage

```
get_supported_file_types(vb = FALSE)
```

Arguments

vb A Boolean value. If TRUE provides verbose output.

Value

A data frame with the file types permitted on Databrary.

Examples

```
get_supported_file_types()
```

get_volume_by_id

Get Data From A Databrary Volume

Description

Get Data From A Databrary Volume

Usage

```
get_volume_by_id(vol_id = 1, vb = FALSE, rq = NULL)
```

Arguments

vol_id	Volume ID.
vb	A logical value. Show verbose messages.
rq	An httr2 request object. If NULL (the default), a new request is generated using <code>make_default_request()</code> . To access restricted data, the user must login with a specific request object using <code>login_db()</code> .

Value

A JSON blob with the volume data. If the user has previously logged in to Databrary via `login_db()`, then volume(s) that have restricted access can be downloaded, subject to the sharing release levels on those volume(s).

Examples

```
get_volume_by_id() # Default is Volume 1
```

HHMMSSmmm_to_ms *Convert Timestamp String To ms.*

Description

Convert Timestamp String To ms.

Usage

```
HHMMSSmmm_to_ms(HHMMSSmmm = "01:01:01:333")
```

Arguments

HHMMSSmmm a string in the format "HH:MM:SS:mmm"

Value

A numeric value in ms from the input string.

Examples

```
HHMMSSmmm_to_ms() # 01:01:01:333 in ms
```

is_institution	<i>Is This Party An Institution?</i>
----------------	--------------------------------------

Description

Is This Party An Institution?

Usage

```
is_institution(party_id = 8, vb = FALSE, rq = NULL)
```

Arguments

party_id	Databrary party ID
vb	A Boolean value. If TRUE provides verbose output.
rq	An httr2 request object.

Value

TRUE if the party is an institution, FALSE otherwise.

Examples

```
is_institution() # Is party 8 (NYU) an institution.
```

is_person	<i>Is This Party A Person?</i>
-----------	--------------------------------

Description

Is This Party A Person?

Usage

```
is_person(party_id = 7, vb = FALSE, rq = NULL)
```

Arguments

party_id	Databrary party ID
vb	A boolean value. If TRUE provides verbose output.
rq	An httr2 request object.

Value

TRUE if the party is a person, FALSE otherwise.

Examples

```
is_person()
```

list_affiliates	<i>List Affiliates for A Databrary Party.</i>
-----------------	---

Description

#' @description [**Superseded**]

Usage

```
list_affiliates(party_id = 6, report_target_party = FALSE, vb = FALSE)
```

Arguments

party_id	Target volume number.
report_target_party	Report data about the target party. Default is FALSE.
vb	A Boolean value. If TRUE provides verbose output.

Details

This function has been superseded by `list_party_affiliates()`.

Value

A data frame with information about a party's affiliates (sponsored investigators).

Examples

```
list_affiliates() # Lists Rick Gilmore's affiliates  
#'
```

list_asset_formats *List Stored Assets (Files) By Type.*

Description

List Stored Assets (Files) By Type.

Usage

```
list_asset_formats(vb = FALSE)
```

Arguments

vb A boolean value. If TRUE provides verbose output.

Value

A data frame with information about the data formats Databrary supports.

Examples

```
list_asset_formats()
```

list_authorized_investigators
 List Authorized Investigators at Institution

Description

List Authorized Investigators at Institution

Usage

```
list_authorized_investigators(party_id = 12, vb = FALSE, rq = NULL)
```

Arguments

party_id Target party ID.
vb A Boolean value. If TRUE provides verbose output.
rq An httr2-style request object. If NULL, then a new request will be generated using make_default_request().

Value

A data frame with information the institution's authorized investigators.

Examples

```
## Not run:  
list_institutional_affiliates() # Default is Penn State (party 12)  
  
## End(Not run)
```

list_individual_sponsors

List A Party's Individual Sponsors.

Description

[Deprecated]

This function has been deprecated and may be removed in a future release. See `list_party_sponsors()` for similar functionality.

A party (person or institution) may have a sponsor that is another person or an institution. This function lists the *person(s)* who sponsor a party.

Usage

```
list_individual_sponsors(party_id = 406, vb = FALSE, rq = NULL)
```

Arguments

party_id	Party number. Default is 406 (Kasey Soska)
vb	A Boolean value. If TRUE provides verbose output.
rq	An <code>httr2</code> request object. If NULL (the default) a request will be generated, but this will only permit public information to be returned.

Value

A data frame with information about a party.

Examples

```
## Not run:  
list_individual_sponsors() # Default is Kasey Soska (party 406)  
  
## End(Not run)
```

```
list_institutional_sponsors
```

List Institutional Sponsors For A Party.

Description

[Deprecated]

This function has been deprecated and may be removed in a future release.

A party (person or institution) may have a sponsor that is another person or an institution. This function lists the *institution(s)* that sponsor a party.

Usage

```
list_institutional_sponsors(
  party_id = 6,
  report_target_party = FALSE,
  vb = FALSE
)
```

Arguments

party_id	Target volume number.
report_target_party	A Boolean value. Print info about the target party.
vb	A Boolean value. If TRUE provides verbose output.

Value

A data frame with information about a party's *institutional* sponsor(s).

Examples

```
list_institutional_sponsors() # Defaults to Rick Gilmore (party 6)
```

```
list_party
```

List Data For A Party (person or institution).

Description

[Deprecated]

This function has been deprecated and may be removed in a future release. See `get_party_by_id()` for similar functionality.

Usage

```
list_party(party_id = 8, component = "all", vb = FALSE)
```

Arguments

party_id	Target volume number. Default is volume 8 (NYU).
component	Which data to return 'children', 'parents', or 'all'. Default is all.
vb	A Boolean value. If TRUE provides verbose output. Default is FALSE.

Value

A data frame with information about a party (person or institution).

Examples

```
list_party() # Default is New York University (party 8)
#'
```

`list_party_affiliates` *List Affiliates For A Party*

Description

List Affiliates For A Party

Usage

```
list_party_affiliates(party_id = 6, vb = FALSE, rq = NULL)
```

Arguments

party_id	Target party ID.
vb	A Boolean value. If TRUE provides verbose output.
rq	An httr2 request object. Defaults to NULL.

Value

A data frame with information about a party's affiliates.

Examples

```
list_party_affiliates() # Default is Rick Gilmore (party 6)
```

list_party_sponsors *List Sponsors For A Party*

Description

List Sponsors For A Party

Usage

```
list_party_sponsors(party_id = 6, vb = FALSE, rq = NULL)
```

Arguments

party_id	Target party ID.
vb	A Boolean value. If TRUE provides verbose output.
rq	An http2-style request object. If NULL, then a new request will be generated using make_default_request().

Value

A data frame with information about a party's sponsors.

Examples

```
## Not run:  
list_party_sponsors() # Default is Rick Gilmore (party 6)  
  
## End(Not run)
```

list_party_volumes *List Volumes For A Party*

Description

List Volumes For A Party

Usage

```
list_party_volumes(party_id = 6, vb = FALSE, rq = NULL)
```

Arguments

party_id	Target party ID.
vb	A Boolean value. If TRUE provides verbose output.
rq	An httr2-style request object. If NULL, then a new request will be generated using make_default_request().

Value

A data frame with information about a party's sponsors.

Examples

```
## Not run:
list_party_volumes() # Default is Rick Gilmore (party 6)

## End(Not run)
```

list_people	<i>List Info About People.</i>
-------------	--------------------------------

Description**[Deprecated]**

This function has been deprecated and may be removed in a future release. See `get_party_by_id()` for similar functionality.

Usage

```
list_people(people_list = 5:7, vb = FALSE)
```

Arguments

people_list	Party number(s) for people to list.
vb	A Boolean value if TRUE returns verbose output.

Value

A data frame with information about the people in the specified range of party ID's.

Examples

```
list_people() # Lists people with party IDs 5:7, Databrary's founders
#'
```

list_sessions	<i>Download Information About A Volume or Volume's Sessions.</i>
---------------	--

Description

[Deprecated] list_sessions() has been deprecated in favor of list_volume_sessions().

Usage

```
list_sessions(vol_id = 1, vb = FALSE)
```

Arguments

vol_id	Target volume ID. Defaults to 1.
vb	A boolean value.

Value

Information about sessions in a volume.

Examples

```
list_sessions() # Lists sessions in volume 1.
```

list_sessions_in_volume	<i>List Sessions in a Volume.</i>
-------------------------	-----------------------------------

Description

[Superseded] list_sessions_in_volume() has been superseded in list_sessions().

Usage

```
list_sessions_in_volume(vol_id = 1, vb = FALSE)
```

Arguments

vol_id	Target volume ID. Defaults to 1.
vb	Provide verbose output. Default is FALSE.

Value

A data frame with information about the sessions in a volume.

Examples

```
list_sessions_in_volume() # Lists sessions in volume 1.
```

list_session_activity *List Activity History in Databrary Session.*

Description

If a user has access to a volume and session, this function returns the history of modifications to that session.

Usage

```
list_session_activity(session_id = 6256, vb = FALSE, rq = NULL)
```

Arguments

session_id	Selected session/slot number.
vb	A Boolean value. If TRUE provides verbose output.
rq	An httr2 request object. Defaults to NULL. To access the activity history on a volume a user has privileges on. Create a request (rq <- make_default_request()); login using make_login_client(rq = rq); then run list_session_activity(session_id = <YOUR_S

Value

A list with the activity history on a session/slot.

Examples

```
## Not run:  
# The following will only return output if the user has write privileges  
# on the session.  
  
list_session_activity(session_id = 6256, vb = FALSE) # Activity for session 6256.  
  
## End(Not run)
```

list_session_assets *List Assets in a Databrary Session.*

Description

#' @description [**Experimental**]

Usage

```
list_session_assets(session_id = 9807, vb = FALSE, rq = NULL)
```

Arguments

session_id An integer. A Databrary session number. Default is 9807, the "materials" folder from Databrary volume 1.

vb A logical value. Show verbose feedback. Default is FALSE.

rq An httr2 request object. If NULL, a default request is generated from databraryr::make_default_request()

Details

As of v0.6.3 `list_session_assets()` replaces an older function that is now named `list_volume_session_assets()`. The older function requires both a volume ID and a session ID. The new function requires only a session ID.

Value

A data frame with information about all assets in a volume.

Examples

```
## Not run:
list_session_assets() # Session 9807 in volume 1

## End(Not run)
```

list_sponsors *List Sponsors For A Party*

Description

List Sponsors For A Party

Usage

```
list_sponsors(party_id = 6, vb = FALSE, rq = NULL)
```

Arguments

party_id	Target party ID.
vb	A Boolean value. If TRUE provides verbose output.
rq	An http2-style request object. If NULL, then a new request will be generated using make_default_request().

Value

A data frame with information about a party's sponsors.

Examples

```
## Not run:
list_sponsors() # Default is Rick Gilmore (party 6)

## End(Not run)
```

list_volume	<i>List Info for a Databrary Volume.</i>
-------------	--

Description

[Deprecated] list_volume() has been deprecated in lieu of other, more specific, volume-related functions. It will be dropped from a future release.

Usage

```
list_volume(vol_id = 1, search_str = "", vb = FALSE)
```

Arguments

vol_id	Target volume number.
search_str	The specific data to retrieve for the volume
vb	A Boolean value. If TRUE provides verbose output.

Value

A data frame with information about a volume.

Examples

```
## Not run:
list_volume() # Default is volume 1

## End(Not run)
```

list_volume_activity *List Activity In A Databrary Volume*

Description

If a user has access to a volume, this command lists the modification history of the volume as a

Usage

```
list_volume_activity(vol_id = 1, vb = FALSE, rq = NULL)
```

Arguments

vol_id	Selected volume number.
vb	A Boolean value. If TRUE provides verbose output.
rq	An httr2 request object. Defaults to NULL.

Value

A list with the activity history on a volume.

Examples

```
## Not run:  
# The following will only return output if the user has write privileges  
# on the volume.  
  
list_volume_activity(vol_id = 1) # Activity on volume 1.  
  
## End(Not run)
```

list_volume_assets *List Assets in Databrary Volume.*

Description

List Assets in Databrary Volume.

Usage

```
list_volume_assets(vol_id = 1, vb = FALSE, rq = NULL)
```

Arguments

vol_id	Target volume number. Default is 1.
vb	A logical value. Default is FALSE.
rq	An http2 request object. Default is NULL.

Value

A data frame with information about all assets in a volume.

Examples

```
## Not run:  
list_volume_assets() # Assets in volume 1  
  
## End(Not run)
```

list_volume_excerpts *List Image or Video Excerpts On A Databrary Volume.*

Description

List Image or Video Excerpts On A Databrary Volume.

Usage

```
list_volume_excerpts(vol_id = 1, vb = FALSE, rq = NULL)
```

Arguments

vol_id	Target volume number.
vb	A Boolean value. If TRUE provides verbose output.
rq	An http2 request object. Default is NULL.

Value

A list with information about any available excerpts.

Examples

```
list_volume_excerpts()
```

list_volume_funding *Lists Funders Associated With a Databrary Volume.*

Description

Lists Funders Associated With a Databrary Volume.

Usage

```
list_volume_funding(vol_id = 1, add_id = TRUE, vb = FALSE, rq = NULL)
```

Arguments

vol_id	Target volume number.
add_id	A logical value. Include the volume ID in the output. Default is TRUE.
vb	A Boolean value. If TRUE provides verbose output.
rq	An httr2 request object.

Value

A data frame with funder information for the volume.

Examples

```
## Not run:  
list_volume_funding() # Funding for volume 1  
  
list_volume_funding(vol_id = c(1:10))  
  
## End(Not run)
```

list_volume_links *Retrieves URL Links From A Databrary Volume.*

Description

Retrieves URL Links From A Databrary Volume.

Usage

```
list_volume_links(vol_id = 1, vb = FALSE, rq = NULL)
```

Arguments

vol_id	Target volume number.
vb	A Boolean value. If TRUE provides verbose output.
rq	An httr2 request object.

Value

A data frame with the requested data.

Examples

```
## Not run:  
list_volume_links() # Links from volume 1  
  
## End(Not run)
```

list_volume_owners *List Owners of a Databrary Volume.*

Description

List Owners of a Databrary Volume.

Usage

```
list_volume_owners(vol_id = 1, vb = FALSE, rq = NULL)
```

Arguments

vol_id	Selected volume number. Default is volume 1.
vb	A boolean value. If TRUE provides verbose output.
rq	An httr2 request object. If NULL (the default) a request will be generated, but this will only permit public information to be returned.

Value

A data frame with information about a volume's owner(s).

Examples

```
list_volume_owners() # Lists information about the owners of volume 1.
```

list_volume_sessions *List Sessions in Databrary Volume.*

Description

List Sessions in Databrary Volume.

Usage

```
list_volume_sessions(  
  vol_id = 1,  
  include_vol_data = FALSE,  
  vb = FALSE,  
  rq = NULL  
)
```

Arguments

vol_id	Target volume number.
include_vol_data	A Boolean value. Include volume-level metadata or not. Default is FALSE.
vb	A Boolean value. Show verbose output. Default is FALSE.
rq	An http2 request object. If NULL (the default) a request will be generated, but this will only permit public information to be returned.

Value

A data frame with information about all assets in a volume.

Examples

```
## Not run:  
list_volume_sessions() # Sessions in Volume 1  
  
## End(Not run)
```

`list_volume_session_assets`

List Assets in a Session from a Databrary volume.

Description

#' @description [**Experimental**]

Usage

```
list_volume_session_assets(  
  vol_id = 1,  
  session_id = 9807,  
  vb = FALSE,  
  rq = NULL  
)
```

Arguments

<code>vol_id</code>	Target volume number.
<code>session_id</code>	The session number in the selected volume.
<code>vb</code>	A boolean value.
<code>rq</code>	An httr2 request object.

Details

`list_volume_session_assets()` is a new name for the <v0.6.2 `list_session_assets()` as function. There is a new `list_session_assets()` function that does not require the volume ID. The `list_volume_session_assets()` *requires* a volume ID.

Value

A data frame with information about all assets in a volume.

Examples

```
## Not run:  
list_volume_session_assets() # Session 9807 in volume 1  
  
## End(Not run)
```

list_volume_tags	<i>Lists Keywords And Tags For A Volume.</i>
------------------	--

Description

Lists Keywords And Tags For A Volume.

Usage

```
list_volume_tags(vol_id = 1, vb = FALSE, rq = NULL)
```

Arguments

vol_id	Target volume number.
vb	A Boolean value. If TRUE provides verbose output.
rq	An httr2 request object.

Value

A data frame with the requested data.

Examples

```
list_volume_tags()
```

login_db	<i>Log In To Databrary.org.</i>
----------	---------------------------------

Description

Log In To Databrary.org.

Usage

```
login_db(  
  email = NULL,  
  password = NULL,  
  store = FALSE,  
  overwrite = FALSE,  
  vb = FALSE,  
  SERVICE = KEYRING_SERVICE,  
  rq = NULL  
)
```

Arguments

email	Databrary account email address.
password	Databrary password (not recommended as it will be displayed as you type)
store	A boolean value. If TRUE store/retrieve credentials from the system keyring/keychain.
overwrite	A boolean value. If TRUE and store is TRUE, overwrite/ update stored credentials in keyring/keychain.
vb	A boolean value. If TRUE provides verbose output.
SERVICE	A character label for stored credentials in the keyring. Default is "databrary"
rq	An http request object. Defaults to NULL.

Value

Logical value indicating whether log in is successful or not.

Examples

```
login_db() # Queries user for email and password interactively.

## Not run:
# The following shows how to use credentials that have been stored previously.

login_db(email = "you@provider.com", store = TRUE)

## End(Not run)
```

logout_db	<i>Log Out of Databrary.org.</i>
-----------	----------------------------------

Description

Log Out of Databrary.org.

Usage

```
logout_db(vb = FALSE, rq = NULL)
```

Arguments

vb	A boolean value. If TRUE provides verbose output.
rq	An http request object. Defaults to NULL.

Value

TRUE if logging out succeeds, FALSE otherwise.

Examples

```
logout_db()
```

```
make_default_request  Set default httr request parameters.
```

Description

make_default_request sets default parameters for httr requests.

Usage

```
make_default_request()
```

Value

An httr2 request object

Examples

```
make_default_request()
```

```
make_fn_portable      Make Portable File Names
```

Description

Make Portable File Names

Usage

```
make_fn_portable(
  fn,
  vb = FALSE,
  replace_regex = "[ &!\\|\\)\\(\\|\\}\\|\\{\\|\\[\\|\\]|+\\|=\\|#\\|\\$%\\|^\\|\\*]",
  replacement_char = "_"
)
```

Arguments

fn Databrary party ID

vb A boolean value. If TRUE provides verbose output.

replace_regex A character string. A regular expression to capture the "non-portable" characters in fn.

replacement_char A character string. The character(s) that will replace the non-portable characters.

Value

A "cleaned" portable file name

make_login_client	<i>Log In To Databrary.org.</i>
-------------------	---------------------------------

Description

Log In To Databrary.org.

Usage

```
make_login_client(
  email = NULL,
  password = NULL,
  store = FALSE,
  overwrite = FALSE,
  vb = FALSE,
  SERVICE = KEYRING_SERVICE,
  rq = NULL
)
```

Arguments

email	Databrary account email address.
password	Databrary password (not recommended as it will displayed as you type)
store	A boolean value. If TRUE store/retrieve credentials from the system keyring/keychain.
overwrite	A boolean value. If TRUE and store is TRUE, overwrite/ update stored credentials in keyring/keychain.
vb	A boolean value. If TRUE provides verbose output.
SERVICE	A character label for stored credentials in the keyring. Default is "databrary"
rq	An httr2 request object. Defaults to NULL.

Value

Logical value indicating whether log in is successful or not.

Examples

```
make_login_client() # Queries user for email and password interactively.

## Not run:
# The following shows how to use credentials that have been stored previously.

make_login_client(email = "you@provider.com", store = TRUE)
```

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

search_for_funder *Report Information About A Funder.*

Description

Report Information About A Funder.

Usage

```
search_for_funder(  
  search_string = "national+science+foundation",  
  vb = FALSE,  
  rq = NULL  
)
```

Arguments

search_string String to search.

vb A Boolean value. If TRUE provides verbose output.

rq An `httr2` request object. Default is NULL.

Value

A data frame with information about the funder.

Examples

```
search_for_funder("national+science+foundation")
```

search_for_keywords *Search For Keywords in Databrary Volumes.*

Description

Search For Keywords in Databrary Volumes.

Usage

```
search_for_keywords(search_string = "locomotion", vb = FALSE, rq = NULL)
```

Arguments

search_string String to search.
vb A Boolean value. If TRUE provides verbose output.
rq An httr2 request object. Default is NULL.

Value

A list with the volumes that contain the keyword.

Examples

```
## Not run:  
search_for_keywords() # searches for volumes with "locomotion" as a keyword.  
  
search_for_keywords("adult") # searches for volumes with "adult" as a keyword.  
  
## End(Not run)
```

search_for_tags *Search For Tags on Volumes or Sessions.*

Description

Search For Tags on Volumes or Sessions.

Usage

```
search_for_tags(search_string = "ICIS", vb = FALSE, rq = NULL)
```

Arguments

search_string String to search.
vb A Boolean value. If TRUE provides verbose output.
rq An httr2 request object. Default is NULL.

Value

An array of tags that match the tag_string.

Examples

```
## Not run:  
search_for_tags() # Searches for volumes that have the tag "ICIS"  
  
## End(Not run)
```


Index

* datasets

- API_CONSTANTS, 3
- API_CONSTANTS, 3
- assign_constants, 3
- check_ssl_certs, 4
- download_party, 5
- download_party_avatar, 5
- download_session_asset, 6
- download_session_assets_fr_df, 7
- download_session_csv, 8
- download_session_zip, 9
- download_single_session_asset_fr_df, 10
- download_video, 12
- download_volume_zip, 13
- get_asset_segment_range, 14
- get_assets_from_session, 14
- GET_db_contents, 15
- get_db_stats, 16
- get_file_duration, 17
- get_party_as_df, 17
- get_party_by_id, 18
- get_permission_levels, 19
- get_release_levels, 20
- get_session_by_id, 20
- get_session_by_name, 21
- get_supported_file_types, 22
- get_volume_by_id, 22
- HHMSSmmm_to_ms, 23
- is_institution, 24
- is_person, 24
- list_affiliates, 25
- list_asset_formats, 26
- list_authorized_investigators, 26
- list_individual_sponsors, 27
- list_institutional_sponsors, 28
- list_party, 28
- list_party_affiliates, 29
- list_party_sponsors, 30
- list_party_volumes, 30
- list_people, 31
- list_session_activity, 33
- list_session_assets, 34
- list_sessions, 32
- list_sessions_in_volume, 32
- list_sponsors, 34
- list_volume, 35
- list_volume_activity, 36
- list_volume_assets, 36
- list_volume_excerpts, 37
- list_volume_funding, 38
- list_volume_links, 38
- list_volume_owners, 39
- list_volume_session_assets, 41
- list_volume_sessions, 40
- list_volume_tags, 42
- login_db, 42
- logout_db, 43
- make_default_request, 44
- make_fn_portable, 44
- make_login_client, 45
- search_for_funder, 46
- search_for_keywords, 47
- search_for_tags, 47