

Package ‘galvanizer’

October 13, 2022

Title Interface to Galvanize 'Highbond' Internal Audit Software

Version 0.5.3

Description An R interface to the Galvanize 'Highbond' API <<https://docs-apis.highbond.com>>.

License GPL-3

Encoding UTF-8

RoxygenNote 7.1.1

Imports magrittr, dplyr, httr, jsonlite, lubridate, rlang, tidyjson,
tibble, tidyr, tidyrselect

Suggests testthat

URL <https://jonlinca.github.io/galvanizer/>,
<https://github.com/jonlinca/galvanizer>

BugReports <https://github.com/jonlinca/galvanizer/issues>

NeedsCompilation no

Author Jonathan Lin [aut, cre] (<<https://orcid.org/0000-0001-8664-6846>>)

Maintainer Jonathan Lin <hello@jonlin.ca>

Repository CRAN

Date/Publication 2021-05-28 15:10:07 UTC

R topics documented:

create_results_analyses	2
create_results_collections	3
create_results_columns	5
create_results_tables	6
get_projects	7
get_project_actions	8
get_project_collaborators	9
get_project_controls	10
get_project_control_tests	11
get_project_control_test_plans	12

get_project_entities	13
get_project_entity_categories	13
get_project_issues	14
get_project_mitigations	15
get_project_narratives	16
get_project_objectives	17
get_project_planning_files	18
get_project_request_items	19
get_project_result_files	19
get_project_risks	20
get_project_types	21
get_project_type_custom_attributes	22
get_project_walkthroughs	23
get_results_analyses	23
get_results_collections	24
get_results_columns	24
get_results_records	25
get_results_tables	26
post_results_records	26
setup_highbond	28

Index **29**

create_results_analyses

Create, Update or Delete Results - Analyses

Description

Create, Update or Delete Results - Analyses

Usage

create_results_analyses(auth, collection_id, name, ...)

update_results_analyses(auth, analysis_id, ...)

delete_results_analyses(auth, analysis_id)

Arguments

auth	Highbond authentication credentials, created from setup_highbond
collection_id	The ID number of the collection
name	The name of the Results object to create
...	List(s) to add additional data
analysis_id	The ID number of the analyses

Details

Each endpoint has a list of required elements, as listed in the parameters.

For optional data, such as attributes, you may pass additional lists. The top level name of this optional list should match the name of the json level to be added, with key-value pairs within the list. Ultimately, this list object will be added as a child json level, under the 'data' top level json.

For example: attributes is a json is nested under data, and therefore should be created as the name of the object. Then additional information can be added to it as a key-value pair to the list.

Value

If creating or updating, a data frame with the new details. When deleting, JSON response indicating success or failure.

Functions

- update_results_analyses: Update an analyses
- delete_results_analyses: Delete an analyses

create_results_collections

Create, Update or Delete Results - Collections

Description

Create, Update or Delete Results - Collections

Usage

```
create_results_collections(auth, name, ...)
```

```
update_results_collections(auth, collection_id, ...)
```

```
delete_results_collections(auth, collection_id)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
name	The name of the Results object to create
...	List(s) to add additional data
collection_id	The ID number of the collection

Details

Each endpoint has a list of required elements, as listed in the parameters.

For optional data, such as attributes, you may pass additional lists. The top level name of this optional list should match the name of the json level to be added, with key-value pairs within the list. Ultimately, this list object will be added as a child json level, under the 'data' top level json.

For example: attributes is a json is nested under data, and therefore should be created as the name of the object. Then additional information can be added to it as a key-value pair to the list.

Value

If creating or updating, a data frame with the new details. When deleting, JSON response indicating success or failure.

Functions

- update_results_collections: Update a collection
- delete_results_collections: Delete a collection

Examples

```
## Not run:
# Create and delete a Collection
auth <- setup_highbond(Sys.getenv('highbond_openapi'),
  Sys.getenv('highbond_org'),
  Sys.getenv('highbond_datacenter'))

name <- 'galvanizer Test Collection'
response <- create_results_collections(auth, name)
collection_id <- response$id[[1]]
response <- delete_results_collections(auth, collection_id)

# Create a Collection with attributes
myattr <- list(description = 'My first description')
response <- create_results_collections(auth, name, attributes = myattr)
collection_id <- response$id[[1]]

# Update a Collection with attributes
myattr <- list(name = 'galvanizer Super Test Collection', description = 'My second description')
response <- update_results_collections(auth, collection_id, attributes = myattr)

# Delete a Collection
response <- delete_results_collections(auth, collection_id)

## End(Not run)
```

`create_results_columns`*Create or Delete Results - Columns*

Description

`post_results_records` will upload a data frame with the correct specification for the fields uploaded. You can use this function to add columns manually if preferred, especially if a non-standard column type.

Usage

```
create_results_columns(auth, table_id, columns)
```

Arguments

<code>auth</code>	Highbond authentication credentials, created from setup_highbond
<code>table_id</code>	The ID number of the table
<code>columns</code>	A data frame with columns to be added

Details

When creating columns, the argument will accept a data frame. This data frame must only have three columns - `field_name`, `display_name`, and `data_type`. See API for allowable `data_type`.

When deleting columns, the argument will accept a data frame. This data frame must only have one column - `field_name`

Value

If creating or updating, a data frame with the new details. When deleting, JSON response indicating success or failure.

Examples

```
## Not run:
auth <- setup_highbond(Sys.getenv('highbond_openapi'),
  Sys.getenv('highbond_org'),
  Sys.getenv('highbond_datacenter'))

field_name <- c("a", "b", "c", "d", "e", "f", "g")
display_name <- c("field_one", "field_two", "field_three",
  "field_four", "field_five", "field_six", "field_seven")
data_type <- c("character", "numeric", "logical",
  "date", "datetime", "file", "digital_signature")
columns <- data.frame(field_name, display_name, data_type)

table_id <- 12345
response <- create_results_columns(auth, table_id, columns)
```

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

create_results_tables *Create, Update or Delete Results - Tables*

Description

Create, Update or Delete Results - Tables

Usage

```
create_results_tables(auth, analysis_id, name, ...)
```

```
update_results_tables(auth, table_id, ...)
```

```
delete_results_tables(auth, table_id)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
analysis_id	The ID number of the analyses
name	The name of the Results object to create
...	List(s) to add additional data
table_id	The ID number of the table

Details

Each endpoint has a list of required elements, as listed in the parameters.

For optional data, such as attributes, you may pass additional lists. The top level name of this optional list should match the name of the json level to be added, with key-value pairs within the list. Ultimately, this list object will be added as a child json level, under the 'data' top level json.

For example: attributes is a json is nested under data, and therefore should be created as the name of the object. Then additional information can be added to it as a key-value pair to the list.

Value

If creating or updating, a data frame with the new details. When deleting, JSON response indicating success or failure.

Functions

- update_results_tables: Update an analyses
- delete_results_tables: Delete an analyses

get_projects	<i>Retrieve Highbond Projects - Projects</i>
--------------	--

Description

Downloads the primary details of one or all projects

Usage

```
get_projects(  
  auth,  
  project_id = NULL,  
  fields = NULL,  
  pagesize = 50,  
  waittime = 0.2  
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
project_id	Project ID number. NULL will default to all items.
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Details

Fields allowed: name, state, status, created_at, updated_at, description, background, budget, position, certification, control_performance, risk_assurance, management_response, max_sample_size, number_of_testing_rounds, opinion, opinion_description, purpose, scope, start_date, target_date, tag_list, project_type, entities

Value

A tibble of projects

Examples

```
## Not run:  
projects <- get_projects(auth)  
projects <- get_projects(auth, fields = c('name', 'state', 'status'))  
  
## End(Not run)
```

get_project_actions *Retrieve Highbond Project - Actions*

Description

"An action is a specific follow-up measure that is associated with an identified issue. You can add actions and assign action owners. You can also set up reminders for yourself to retest issues or track hours spent on retesting by self-assigning actions."

Usage

```
get_project_actions(
  auth,
  issue_id = NULL,
  action_id = NULL,
  fields = NULL,
  pagesize = 50,
  waittime = 0.2
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
issue_id	Required if other parameter is blank. May obtain multiple rows.
action_id	Required if other parameter is blank. Will get only one.
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Details

possible fields: title, created_at, updated_at, owner_name, owner_email, send_recurring_reminder, include_issue_details, include_remediation_details, description, due_date, priority, closed, completed_date, status, submitted_on, custom_attributes, issue, assigned_by, cc_users

Value

A tibble of actions

`get_project_collaborators`*Retrieve Highbond Projects - Collaborators*

Description

Get the collaborators, aka user roles in Project or Framework.

Usage

```
get_project_collaborators(  
  auth,  
  project_id = NULL,  
  encoded_uid = NULL,  
  fields = NULL,  
  pagesize = 50,  
  waittime = 0.2  
)
```

Arguments

<code>auth</code>	Highbond authentication credentials, created from setup_highbond
<code>project_id</code>	Required if other parameter is blank.
<code>encoded_uid</code>	Required if the other parameter is blank. Base64 encoded parent resource id (project or framework) and user uid, encoded from format <code>parent_resource_id:user_uid..</code>
<code>fields</code>	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
<code>pagesize</code>	Defaults to 50. Maximum is 100.
<code>waittime</code>	Time in seconds to wait between requests.

Details

Fields allowed: `project`, `framework`, `user`, `project_role`, `effective_role`, `organization_role`, `created_at`, `updated_at`, `group`

Value

A tibble of collaborators

get_project_controls *Retrieve Highbond Project - Controls*

Description

"A control is a program, policy, routine, or activity that is intended to mitigate a risk. Controls are organized by objectives, and can be associated with one or more risks. The combination of identified risks and corresponding controls defines the Risk Control Matrix. Controls are also known as procedures."

Usage

```
get_project_controls(  
  auth,  
  objective_id = NULL,  
  control_id = NULL,  
  fields = NULL,  
  pagesize = 50,  
  waittime = 0.2  
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
objective_id	Required if other parameter is blank. May obtain multiple rows.
control_id	Required if other parameter is blank. Will get only one.
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Details

possible fields: title, description, control_id, owner, frequency, control_type, prevent_detect, method, status, position, created_at, updated_at, custom_attributes, objective, walkthrough, control_test_plan, control_tests, mitigations, owner_user, entities

Value

A tibble of controls

`get_project_control_tests`*Retrieve Highbond Project - Control Tests*

Description

"Control tests evaluate the operating effectiveness of a control."

Usage

```
get_project_control_tests(  
  auth,  
  control_test_id,  
  fields = NULL,  
  pagesize = 50,  
  waittime = 0.2  
)
```

Arguments

<code>auth</code>	Highbond authentication credentials, created from setup_highbond
<code>control_test_id</code>	Will get only one.
<code>fields</code>	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
<code>pagesize</code>	Defaults to 50. Maximum is 100.
<code>waittime</code>	Time in seconds to wait between requests.

Details

Only applicable to test plans with Internal Control types

Value

A tibble of control tests

`get_project_control_test_plans`*Retrieve Highbond Project - Test Plans*

Description

"A test plan is a document that details how controls are assessed. Test plans identify the testing method or type of evidence obtained, specify the total sample size (split amongst testing rounds), and illustrate test steps or attributes."

Usage

```
get_project_control_test_plans(  
  auth,  
  control_test_plan_id,  
  fields = NULL,  
  pagesize = 50,  
  waittime = 0.2  
)
```

Arguments

<code>auth</code>	Highbond authentication credentials, created from setup_highbond
<code>control_test_plan_id</code>	Will get only one.
<code>fields</code>	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
<code>pagesize</code>	Defaults to 50. Maximum is 100.
<code>waittime</code>	Time in seconds to wait between requests.

Details

Only applicable to test plans with Internal Control types

possible fields: `testing_round_number`, `not_applicable`, `sample_size`, `testing_results`, `testing_conclusion`, `created_at`, `updated_at`, `control`, `assigned_user`

Value

A tibble of control test plans

get_project_entities *Retrieve Highbond Projects - Entities*

Description

Get all the entities

Usage

```
get_project_entities(  
  auth,  
  entity_id = NULL,  
  fields = NULL,  
  pagesize = 50,  
  waittime = 0.2  
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
entity_id	NULL will default to all entities.
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Details

Fields allowed: title, description, created_at, updated_at, parent, children_count, entity_category

Value

A tibble of entities

get_project_entity_categories
Retrieve Highbond Projects - Entity Categories

Description

Get all the entity categories

Usage

```
get_project_entity_categories(
  auth,
  entity_category_id = NULL,
  fields = NULL,
  pagesize = 50,
  waittime = 0.2
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
entity_category_id	NULL will default to all entity categories
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Details

Fields allowed: title, description, position, entities_count, created_at, updated_at, entities

Value

A tibble of entities

get_project_issues	<i>Retrieve Highbond Project - Issues</i>
--------------------	---

Description

"An issue is a problem, control gap, or exception that has been identified within a project. Adding an issue involves recording basic information about the issue and assigning the issue to an owner. Issues may also be known as deficiencies, findings, or another customized term."

Usage

```
get_project_issues(
  auth,
  project_id = NULL,
  issue_id = NULL,
  fields = NULL,
  pagesize = 50,
  waittime = 0.2
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
project_id	Required if other parameter is blank. May obtain multiple rows.
issue_id	Required if other parameter is blank. Will get only one.
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Details

possible fields: title, description, creator_name, created_at, updated_at, position, owner, recommendation, deficiency_type, severity, published, identified_at, reference, reference_prefix, risk, scope, escalation, cause, effect, cost_impact, executive_summary, executive_owner, project_owner, closed, remediation_status, remediation_plan, remediation_date, actual_remediation_date, retest_deadline_date, actual_retest_date, retesting_results_overview, custom_attributes, project, entities

Value

A tibble of issues

```
get_project_mitigations
```

Retrieve Highbond Project - Mitigations

Description

The mapping between controls and risks

Usage

```
get_project_mitigations(
  auth,
  mitigation_id,
  fields = NULL,
  pagesize = 50,
  waittime = 0.2
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
mitigation_id	Will get only one.
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Value

A tibble of mitigations

```
get_project_narratives
```

Retrieve Highbond Project - Narratives

Description

"A narrative is a description of a business process or area under review. Narratives are also known as policies, process narratives, process descriptions, or control guides."

Usage

```
get_project_narratives(  
  auth,  
  objective_id = NULL,  
  narrative_id = NULL,  
  fields = NULL,  
  pagesize = 50,  
  waittime = 0.2  
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
objective_id	Required if other parameter is blank. May obtain multiple rows.
narrative_id	Required if other parameter is blank. Will get only one row.
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Details

possible fields: title, description, created_at, updated_at, objective

Value

A tibble of narratives

`get_project_objectives`*Retrieve Highbond Project - Objectives*

Description

Downloads the primary details of one or multiple objectives for a project. Also known as sections, processes, cycles, functional areas, application systems, or another custom term.

Usage

```
get_project_objectives(  
  auth,  
  project_id = NULL,  
  objective_id = NULL,  
  fields = NULL,  
  pagesize = 50,  
  waittime = 0.2  
)
```

Arguments

<code>auth</code>	Highbond authentication credentials, created from setup_highbond
<code>project_id</code>	Required if other parameter is blank. May obtain multiple rows.
<code>objective_id</code>	Required if other parameter is blank. Will get only one row.
<code>fields</code>	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
<code>pagesize</code>	Defaults to 50. Maximum is 100.
<code>waittime</code>	Time in seconds to wait between requests.

Details

possible fields: title, description, reference, division_department, owner, executive_owner, created_at, updated_at, project, assigned_user, custom_attributes, position, risk_control_matrix_id, walkthrough_summary_id, testing_round_1_id, testing_round_2_id, testing_round_3_id, testing_round_4_id, entities

Value

A tibble of objectives

`get_project_planning_files`*Retrieve Highbond Project - Planning Files*

Description

Downloads the primary details of one or multiple planning files for a project.

Usage

```
get_project_planning_files(  
  auth,  
  project_id = NULL,  
  planning_file_id = NULL,  
  fields = NULL,  
  pagesize = 50,  
  waittime = 0.2  
)
```

Arguments

<code>auth</code>	Highbond authentication credentials, created from setup_highbond
<code>project_id</code>	Required if other parameter is blank. May obtain multiple rows.
<code>planning_file_id</code>	Required if other parameter is blank. Will get only one row.
<code>fields</code>	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
<code>pagesize</code>	Defaults to 50. Maximum is 100.
<code>waittime</code>	Time in seconds to wait between requests.

Details

possible fields: name, reference_id, description, position, created_at, updated_at, custom_attributes, project

Value

A tibble of planning files.

`get_project_request_items`*Retrieve Highbond Project - Request Items*

Description

Capture what you asked for

Usage

```
get_project_request_items(  
  auth,  
  request_id,  
  fields = NULL,  
  pagesize = 50,  
  waittime = 0.2  
)
```

Arguments

<code>auth</code>	Highbond authentication credentials, created from setup_highbond
<code>request_id</code>	Will get only one.
<code>fields</code>	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
<code>pagesize</code>	Defaults to 50. Maximum is 100.
<code>waittime</code>	Time in seconds to wait between requests.

Value

A tibble of walkthroughs

`get_project_result_files`*Retrieve Highbond Project - Result Files*

Description

Downloads the primary details of one or multiple result files for a project.

Usage

```
get_project_result_files(
  auth,
  project_id = NULL,
  results_file_id = NULL,
  fields = NULL,
  pagesize = 50,
  waittime = 0.2
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
project_id	Required if other parameter is blank. May obtain multiple rows.
results_file_id	Required if other parameter is blank. Will get only one row.
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Details

possible fields: name, reference_id, description, position, created_at, updated_at, custom_attributes, project

Value

A tibble of result files

get_project_risks	<i>Retrieve Highbond Project - Risks</i>
-------------------	--

Description

"A narrative is a description of a business process or area under review. Narratives are also known as policies, process narratives, process descriptions, or control guides."

Usage

```
get_project_risks(
  auth,
  objective_id = NULL,
  risk_id = NULL,
  fields = NULL,
  pagesize = 50,
  waittime = 0.2
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
objective_id	Required if other parameter is blank. May obtain multiple rows.
risk_id	Required if other parameter is blank. Will get only one.
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Details

possible fields: title, description, risk_id, position, impact, likelihood, custom_attributes, created_at, updated_at, objective, mitigations, entities

Value

A tibble of Risks

get_project_types	<i>Retrieve Highbond Projects - Project Types</i>
-------------------	---

Description

Downloads the primary details of one or all project types.

Usage

```
get_project_types(
  auth,
  project_type_id = NULL,
  fields = NULL,
  pagesize = 50,
  waittime = 0.2
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
project_type_id	NULL Defaults to all project types
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Details

Possible fields include: name, description, workflow, project_terms, certification_terms, control_terms, finding_terms, control_test_terms, finding_action_terms, narrative_terms, objective_terms, planning_terms, results_terms, risk_terms, test_plan_terms, walkthrough_terms

Value

A tibble of project types

get_project_type_custom_attributes

Retrieve Highbond Projects - Project Type Custom Attributes

Description

Get the custom attributes set within a project type. Note these are different than the custom terms used to rename fields. For those, see [get_project_types](#)

Usage

```
get_project_type_custom_attributes(
  auth,
  project_type_id = NULL,
  custom_attributes_id = NULL,
  fields = NULL,
  pagesize = 50,
  waittime = 0.2
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
project_type_id	Required if other parameter is blank.
custom_attributes_id	Required if the other parameter is blank. Base64 encoded parent resource id (project or framework) and user uid, encoded from format parent_resource_id:user_uid..
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Value

A tibble of custom attributes

 get_project_walkthroughs

Retrieve Highbond Project - Walkthroughs / Execute Procedures

Description

"A walkthrough is a series of steps you perform to establish the reliability of controls and test the design of controls. Each control you define has a corresponding walkthrough that is used to verify that the control is designed appropriately. In a Workplan workflow project, a walkthrough is called an execute procedure."

Usage

```
get_project_walkthroughs(
  auth,
  walkthrough_id,
  fields = NULL,
  pagesize = 50,
  waittime = 0.2
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
walkthrough_id	Will get only one.
fields	OPTIONAL. A character vector each field requested within the project. NULL will default to all fields.
pagesize	Defaults to 50. Maximum is 100.
waittime	Time in seconds to wait between requests.

Value

A tibble of walkthroughs

 get_results_analyses *Retrieve Highbond Results - Analyses*

Description

Downloads a list of Analyses in a Collection

Usage

```
get_results_analyses(auth, collection_id = NULL, analysis_id = NULL)
```

Arguments

auth Highbond authentication credentials, created from [setup_highbond](#)
collection_id Collection ID. Required if other parameter is blank.
analysis_id Analyses ID. Required if other parameter is blank.

Value

A data frame of Analyses in a Collection

get_results_collections

Retrieve Highbond Results - Collections

Description

Downloads a list of collections

Usage

```
get_results_collections(auth, collection_id = NULL)
```

Arguments

auth Highbond authentication credentials, created from [setup_highbond](#)
collection_id Collection ID. Optional.

Value

A data frame of Collections

get_results_columns

Retrieve Highbond Results - Columns

Description

Gets the schema of a single table.

Usage

```
get_results_columns(auth, table_id)
```

Arguments

auth Highbond authentication credentials, created from [setup_highbond](#)
table_id Table ID. Required.

Value

A data frame column name and types of a single Table

get_results_records	<i>Retrieve Highbond Results - Records</i>
---------------------	--

Description

Downloads the content in the structure as saved by Highbond Results, including metadata and questionnaire responses (if applicable).

Usage

```
get_results_records(auth, table_id, timezone = Sys.timezone())
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
table_id	The table to be downloaded
timezone	Defaults to Sys.timezone(). See OlsonNames() for a list of allowable time-zones.

Details

Table ID can be found in the path of your Result table. https://<company_name>.results.highbond.com/projects/<COLLECTI

Value

A list containing the contents, table requested and the original response

Examples

```
## Not run:  
response <- get_results_records(auth, 567890)  
  
## End(Not run)
```

get_results_tables *Retrieve Highbond Results - Tables*

Description

Downloads the primary details of one or all tables in an Analyses

Usage

```
get_results_tables(auth, analysis_id = NULL, table_id = NULL)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
analysis_id	Analysis ID. Required if other parameter is blank.
table_id	Table ID. Required if other parameter is blank.

Value

A data frame of Tables in an Analyses

post_results_records *Upload Highbond Results - Records*

Description

Uploads the content in to Highbond Results.

Usage

```
post_results_records(  
  auth,  
  table_id,  
  upload = NULL,  
  purge = FALSE,  
  skipquestions = FALSE,  
  sizelimit = 75000  
)
```

Arguments

auth	Highbond authentication credentials, created from setup_highbond
table_id	The table to be downloaded
upload	A data frame to be uploaded. Classes supported are c('character', 'numeric', 'logical', 'Date', 'POSIXct', 'POSIXlt')
purge	FALSE by default. Whether or not to delete the contents of the table before overwriting.
skipquestions	FALSE by default. Set TRUE if not purging and you don't want to overwrite the responses within the Results table already.
sizelimit	75000 bytes (~75 kb) by default. Used to estimate chunk size. Reduce size if upload chunks tend to fail.

Details

Current Results in the table can be purged or not. If data is not purged, a check is performed that compatible types are being upload and all fields exist. If purged, there will be no checks done beforehand, which may alter structure of existing table.

If the Result table contains questionnaire responses, then any records that are being updated may need to have the responses also included as well. These fields are usually prefixed with a 'q'.

A result row is generally appended, however a Primary Key field within the Data Analytic Settings for that table may be specified so a record with the same primary key may be merged, rather than duplicated.

Value

No return value, although errors will be verbose

Examples

```
## Not run:
upload <- data.frame(field_one = c('A','B','C'),
  field_two = c(1, 2, 3),
  field_three = c(10L, 11L, 12L),
  field_four = c(TRUE, FALSE, TRUE),
  field_five = c(as.Date('2019-01-01'), as.Date('2020-01-01'), as.Date('2021-12-31')),
  field_six = c(as.POSIXct(Sys.time()), as.POSIXct(Sys.time()), as.POSIXct(Sys.time())))
post_results_records(auth, upload = upload, purge = TRUE)

## End(Not run)
```

setup_highbond	<i>Highbond Authentication credentials</i>
----------------	--

Description

Assembles all the core authentication needed to connect to a Highbond instance.

Usage

```
setup_highbond(apikey, instance, datacenter)
```

Arguments

apikey	Highbond API token
instance	The HighBond instance ID, also known as the organization number
datacenter	The region code. Can be 'us', 'ca', 'eu', 'ap', 'au'

Details

Requires a Highbond API token. Activate a token from https://help.highbond.com/helpdocs/highbond/en-us/Content/launchpad/getting_started/managing_access_tokens.html.

The Instance (Organization) number and datacenter can both be found from the Highbond launchpad, then accessing Options and Organization. https://accounts.highbond.com/orgs/<ORG_ID>/details

Value

A Highbond Authentication credentials object to pass to your requests

Index

create_results_analyses, [2](#)
create_results_collections, [3](#)
create_results_columns, [5](#)
create_results_tables, [6](#)

delete_results_analyses
 (create_results_analyses), [2](#)
delete_results_collections
 (create_results_collections), [3](#)
delete_results_tables
 (create_results_tables), [6](#)

get_project_actions, [8](#)
get_project_collaborators, [9](#)
get_project_control_test_plans, [12](#)
get_project_control_tests, [11](#)
get_project_controls, [10](#)
get_project_entities, [13](#)
get_project_entity_categories, [13](#)
get_project_issues, [14](#)
get_project_mitigations, [15](#)
get_project_narratives, [16](#)
get_project_objectives, [17](#)
get_project_planning_files, [18](#)
get_project_request_items, [19](#)
get_project_result_files, [19](#)
get_project_risks, [20](#)
get_project_type_custom_attributes, [22](#)
get_project_types, [21](#), [22](#)
get_project_walkthroughs, [23](#)
get_projects, [7](#)
get_results_analyses, [23](#)
get_results_collections, [24](#)
get_results_columns, [24](#)
get_results_records, [25](#)
get_results_tables, [26](#)

post_results_records, [5](#), [26](#)

setup_highbond, [2](#), [3](#), [5–27](#), [28](#)

update_results_analyses
 (create_results_analyses), [2](#)
update_results_collections
 (create_results_collections), [3](#)
update_results_tables
 (create_results_tables), [6](#)