

Package ‘skilljaR’

July 23, 2025

Title Connect to Your 'Skilljar' Data

Version 0.1.2

Description Functions that simplify calls to the 'Skilljar' API. See
<<https://api.skilljar.com/docs/>> for documentation on the 'Skilljar' API.
This package is not supported by 'Skilljar'.

Depends R (>= 4.1.0)

License CC0

Encoding UTF-8

Imports httr (>= 1.4.2), jsonlite (>= 1.7.2), purrr (>= 0.3.4), dplyr
(>= 1.1.0), magrittr (>= 2.0.1)

RoxygenNote 7.2.3

NeedsCompilation no

Author Chris Umphlett [aut, cre]

Maintainer Chris Umphlett <christopher.umphlett@gmail.com>

Repository CRAN

Date/Publication 2023-09-19 16:00:02 UTC

Contents

get_course_paths	2
get_course_progress	3
get_published_courses	4
get_users	5
get_user_domain	6
Index	7

get_course_paths	<i>Get Course Paths</i>
------------------	-------------------------

Description

Returns a data.frame with all paths and the items in those paths.

Usage

```
get_course_paths(api_token)
```

Arguments

api_token Your personalized token provided by 'Skilljar'

Details

Utilizing the API requires a token. This must be obtained by logging in at dashboard.skilljar.com and going to Organization -> API Credentials. There are different strategies for storing api tokens securely. It is an unnecessary risk to store the token in the script!

Value

A data frame with users and user data

See Also

See <https://api.skilljar.com/docs/> for documentation on the 'Skilljar' API.

Examples

```
## Not run:  
# Get course progress data  
paths <- get_course_paths(api_token = "my-token")  
  
## End(Not run)
```

get_course_progress	<i>Get Course Progress for a List of Users</i>
---------------------	--

Description

Returns a data.frame with all course progress details for a list of user id's.

Usage

```
get_course_progress(user_ids, api_token)
```

Arguments

user_ids	Vector of user id's for which you want course progress data
api_token	Your personalized token provided by 'Skilljar'

Details

The API call only returns one user at a time. If you have many users then it may take a long time for the function to return. You may also exceed the 'Skilljar' API rate limit.

Utilizing the API requires a token. This must be obtained by logging in at dashboard.skilljar.com and going to Organization -> API Credentials. There are different strategies for storing api tokens securely. It is an unnecessary risk to store the token in the script!

Value

A data frame with users and user data

See Also

See <https://api.skilljar.com/docs/> for documentation on the 'Skilljar' API.

Examples

```
## Not run:
# Get some user id's
users <- get_users(domain = "training.mycompany.com",
  users_desired = 10,
  api_token = "my-token")

# Get course progress data
progress <- get_course_progress(user_ids = unique(users$user.id),
  api_token = "my-token")

## End(Not run)
```

get_published_courses *Get All Published Courses*

Description

Returns a data.frame with all published courses from a particular domain.

Usage

```
get_published_courses(domain, api_token)
```

Arguments

domain	Domain of 'Skilljar' account
api_token	Your personalized token provided by 'Skilljar'

Details

Utilizing the API requires a token. This must be obtained by logging in at dashboard.skilljar.com and going to Organization -> API Credentials. There are different strategies for storing api tokens securely. It is an unnecessary risk to store the token in the script!

Value

A data frame with all published course data

See Also

See <https://api.skilljar.com/docs/> for documentation on the 'Skilljar' API.

Examples

```
## Not run:  
# Retrieve all published courses  
courses <- get_published_courses(domain = "training.mycompany.com",  
  api_token = "my-token")  
  
## End(Not run)
```

`get_users`*Get All Users*

Description

Returns a data.frame with all users from a particular domain. If you have many users and do not want to return them all at once, you may request fewer users to save time while doing development. The API returns up 10,000 users at a time– if you request more than 10,000, it will return in full page increments (a multiple of 10,000).

Usage

```
get_users(users_desired = 1e+08, api_token)
```

Arguments

<code>users_desired</code>	Number of user records to return
<code>api_token</code>	Your personalized token provided by 'Skilljar'

Details

Utilizing the API requires a token. This must be obtained by logging in at dashboard.skilljar.com and going to Organization -> API Credentials. There are different strategies for storing api tokens securely. It is an unnecessary risk to store the token in the script!

Value

A data frame with users and user data

See Also

See <https://api.skilljar.com/docs/> for documentation on the 'Skilljar' API.

Examples

```
## Not run:  
# Retrieve 1000 users  
my_users <- get_users(users_desired = 1000,  
  api_token = "my-token")  
  
## End(Not run)
```

get_user_domain	<i>Determine if User has joined a domain</i>
-----------------	--

Description

Accepts a vector of user id's, and a domain, and returns a data.frame that indicates if each id is found in that domain.

Usage

```
get_user_domain(domain, user_ids, api_token)
```

Arguments

domain	Skilljar domain you want to search in
user_ids	Vector of user id's
api_token	Your personalized token provided by 'Skilljar'

Details

Utilizing the API requires a token. This must be obtained by logging in at dashboard.skilljar.com and going to Organization -> API Credentials. There are different strategies for storing api tokens securely. It is an unnecessary risk to store the token in the script!

Value

A data.frame that indicates if each id is found in that domain.

See Also

See <https://api.skilljar.com/docs/> for documentation on the 'Skilljar' API.

Examples

```
## Not run:
# Retrieve user data and domain for 100 users
my_users <- get_users(users_desired = 100,
  api_token = "my-token")
user_vector <- users %>% select(user_id) %>% pull()
my_users2 <- get_user_domain(user_vector, "my-domain.skilljar.com",
  api_token = "my-token")

## End(Not run)
```

Index

`get_course_paths`, [2](#)
`get_course_progress`, [3](#)
`get_published_courses`, [4](#)
`get_user_domain`, [6](#)
`get_users`, [5](#)