

Package ‘zoomr’

July 21, 2025

Title Connect to Your 'Zoom' Data

Version 0.3.0

Description Facilitates making a connection to the 'Zoom' API and executing various queries. You can use it to get data on 'Zoom' webinars and 'Zoom' meetings. The 'Zoom' documentation is available at <https://developers.zoom.us/docs/api/>. This package is not supported by 'Zoom' (owner of the software).

URL <https://github.com/chrisumphlett/zoomr>

BugReports <https://github.com/chrisumphlett/zoomr/issues>

License CC0

Encoding UTF-8

Imports dplyr (>= 1.0.0), magrittr (>= 1.5), jsonlite (>= 1.6.1), httr (>= 1.4.1), glue (>= 1.4.2), rlang (>= 1.0.4), tidyr (>= 1.1.4), janitor (>= 2.1.0), tidyselect (>= 1.1.1), purrr (>= 0.3.4)

RoxygenNote 7.2.3

NeedsCompilation no

Author Chris Umphlett [aut, cre]

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

Repository CRAN

Date/Publication 2024-02-23 18:30:02 UTC

Contents

get_account_users	2
get_panelists	3
get_registration_questions	4
get_tracking_sources	5
get_webinar_details	6
get_webinar_participants	7
get_webinar_polls	8

get_webinar_qanda	9
get_webinar_registrants	10
list_webinars	11
Index	12

get_account_users	<i>Get All Users in an Account</i>
-------------------	------------------------------------

Description

Get all users in a zoom account. The user id is required to get a list of webinars hosted by a particular user.

Usage

```
get_account_users(account_id, client_id, client_secret)
```

Arguments

- account_id Account Id granted by the Zoom developer app.
- client_id Client Id granted by the Zoom developer app.
- client_secret Client secret granted by the Zoom developer app.

Value

A data frame with the list of all users in the account.

See Also

See <<https://marketplace.zoom.us/docs/api-reference/zoom-api/>> for documentation on the Zoom API.

Examples

```
## Not run:
dat <- get_account_users(your_account_id,
  your_client_id,
  your_client_secret)

## End(Not run)
```

get_panelists	<i>Get Webinar Panelists</i>
---------------	------------------------------

Description

Get the list of attendees who were also panelists from a single webinar.

Usage

```
get_panelists(webinar_id, account_id, client_id, client_secret)
```

Arguments

webinar_id	Zoom Webinar Id, typically an 11 digit number.
account_id	Account Id granted by the Zoom developer app.
client_id	Client Id granted by the Zoom developer app.
client_secret	Client secret granted by the Zoom developer app.

Value

A data frame with the list of panelists from that webinar.

See Also

See <<https://marketplace.zoom.us/docs/api-reference/zoom-api/>> for documentation on the Zoom API.

Examples

```
## Not run:
dat <- get_webinar_panelists(webinar_id = "99911112222",
  your_account_id,
  your_client_id,
  your_client_secret)

## End(Not run)
```

`get_registration_questions`*Get Webinar Registration Question Responses*

Description

Get registration question answers from all registrants for a single webinar.

Usage

```
get_registration_questions(  
  webinar_id,  
  account_id,  
  client_id,  
  client_secret,  
  registrant_status = c("approved")  
)
```

Arguments

<code>webinar_id</code>	Zoom Webinar Id, typically an 11 digit number.
<code>account_id</code>	Account Id granted by the Zoom developer app.
<code>client_id</code>	Client Id granted by the Zoom developer app.
<code>client_secret</code>	Client secret granted by the Zoom developer app.
<code>registrant_status</code>	One or more of "approved", "pending", or "denied". Default is "approved" only.

Value

A data frame with the registration question answers.

See Also

See <<https://marketplace.zoom.us/docs/api-reference/zoom-api/>> for documentation on the Zoom API.

Examples

```
## Not run:  
dat <- get_registration_questions(webinar_id = "99911112222",  
  your_account_id,  
  your_client_id,  
  your_client_secret,  
  c("approved", "denied", "pending"))  
  
## End(Not run)
```

get_tracking_sources	<i>Get Webinar Tracking Sources</i>
----------------------	-------------------------------------

Description

Get a summary of registrations and visitors by tracking source for a specific webinar.

Usage

```
get_tracking_sources(webinar_id, account_id, client_id, client_secret)
```

Arguments

webinar_id	Zoom Webinar Id, typically an 11 digit number.
account_id	Account Id granted by the Zoom developer app.
client_id	Client Id granted by the Zoom developer app.
client_secret	Client secret granted by the Zoom developer app.

Value

A data frame with the list of panelists from that webinar.

See Also

See <<https://marketplace.zoom.us/docs/api-reference/zoom-api/>> for documentation on the Zoom API.

Examples

```
## Not run:
dat <- get_tracking_sources(webinar_id = "99911112222",
  your_account_id,
  your_client_id,
  your_client_secret)

## End(Not run)
```

get_webinar_details	<i>Get Webinar Details</i>
---------------------	----------------------------

Description

Get metadata about a single webinar.

Usage

```
get_webinar_details(webinar_id, account_id, client_id, client_secret)
```

Arguments

webinar_id	Zoom Webinar Id, typically an 11 digit number.
account_id	Account Id granted by the Zoom developer app.
client_id	Client Id granted by the Zoom developer app.
client_secret	Client secret granted by the Zoom developer app.

Value

A data frame with the metadata about a webinar.

See Also

See <<https://marketplace.zoom.us/docs/api-reference/zoom-api/>> for documentation on the Zoom API.

Examples

```
## Not run:
dat <- get_webinar_details(webinar_id = "99911112222",
  your_account_id,
  your_client_id,
  your_client_secret)

## End(Not run)
```

get_webinar_participants
Get Webinar Participants

Description

Get participant info about a single webinar.

Usage

```
get_webinar_participants(webinar_id, account_id, client_id, client_secret)
```

Arguments

webinar_id	Zoom Webinar Id, typically an 11 digit number.
account_id	Account Id granted by the Zoom developer app.
client_id	Client Id granted by the Zoom developer app.
client_secret	Client secret granted by the Zoom developer app.

Value

A data frame with data on each participant at a webinar.

See Also

See <<https://marketplace.zoom.us/docs/api-reference/zoom-api/>> for documentation on the Zoom API.

Examples

```
## Not run:  
dat <- get_webinar_participants(webinar_id = "9991111222",  
  your_account_id,  
  your_client_id,  
  your_client_secret)  
  
## End(Not run)
```

get_webinar_polls	<i>Get Webinar Polls</i>
-------------------	--------------------------

Description

Get the polls summary from a single webinar.

Usage

```
get_webinar_polls(webinar_id, account_id, client_id, client_secret)
```

Arguments

webinar_id	Zoom Webinar Id, typically an 11 digit number.
account_id	Account Id granted by the Zoom developer app.
client_id	Client Id granted by the Zoom developer app.
client_secret	Client secret granted by the Zoom developer app.

Value

A data frame with poll results from a webinar.

See Also

See <<https://marketplace.zoom.us/docs/api-reference/zoom-api/>> for documentation on the Zoom API.

Examples

```
## Not run:
dat <- get_webinar_polls(webinar_id = "99911112222",
  your_account_id,
  your_client_id,
  your_client_secret)

## End(Not run)
```

get_webinar_qanda	<i>Get Webinar Q & A</i>
-------------------	------------------------------

Description

Get the Q&A summary from a single webinar.

Usage

```
get_webinar_qanda(webinar_id, account_id, client_id, client_secret)
```

Arguments

webinar_id	Zoom Webinar Id, typically an 11 digit number.
account_id	Account Id granted by the Zoom developer app.
client_id	Client Id granted by the Zoom developer app.
client_secret	Client secret granted by the Zoom developer app.

Value

A data frame with all of the Q&A data from a webinar.

See Also

See <<https://marketplace.zoom.us/docs/api-reference/zoom-api/>> for documentation on the Zoom API.

Examples

```
## Not run:
dat <- get_webinar_qanda(webinar_id = "99911112222",
  your_account_id,
  your_client_id,
  your_client_secret)

## End(Not run)
```

`get_webinar_registrants`*Get Webinar Registrants*

Description

Get registrant info about a single webinar.

Usage

```
get_webinar_registrants(  
  webinar_id,  
  account_id,  
  client_id,  
  client_secret,  
  registrant_status = c("approved")  
)
```

Arguments

<code>webinar_id</code>	Zoom Webinar Id, typically an 11 digit number.
<code>account_id</code>	Account Id granted by the Zoom developer app.
<code>client_id</code>	Client Id granted by the Zoom developer app.
<code>client_secret</code>	Client secret granted by the Zoom developer app.
<code>registrant_status</code>	One or more of "approved", "pending", or "denied". Default is "approved" only.

Value

A data frame with data on all the registrants for a webinar.

See Also

See <<https://marketplace.zoom.us/docs/api-reference/zoom-api/>> for documentation on the Zoom API.

Examples

```
## Not run:  
dat <- get_webinar_registrants(webinar_id = "99911112222",  
  your_account_id,  
  your_client_id,  
  your_client_secret,  
  c("approved", "denied", "pending"))  
  
## End(Not run)
```

list_webinars	<i>Get List of Webinars for a User.</i>
---------------	---

Description

Get list of webinars for a User. This is used to get the webinar Id's to pass into other functions.

Usage

```
list_webinars(user_id, account_id, client_id, client_secret)
```

Arguments

user_id	Zoom User Id.
account_id	Account Id granted by the Zoom developer app.
client_id	Client Id granted by the Zoom developer app.
client_secret	Client secret granted by the Zoom developer app.

Value

A data frame with all of the webinars hosted by a specific user.

See Also

See <<https://marketplace.zoom.us/docs/api-reference/zoom-api/>> for documentation on the Zoom API.

Examples

```
## Not run:
dat <- get_webinar_details(user_id = "user_id_string",
  your_account_id,
  your_client_id,
  your_client_secret)

## End(Not run)
```

Index

`get_account_users`, [2](#)
`get_panelists`, [3](#)
`get_registration_questions`, [4](#)
`get_tracking_sources`, [5](#)
`get_webinar_details`, [6](#)
`get_webinar_participants`, [7](#)
`get_webinar_polls`, [8](#)
`get_webinar_qanda`, [9](#)
`get_webinar_registrants`, [10](#)

`list_webinars`, [11](#)