

Package ‘PhotosynQ’

May 7, 2026

Title Connect to PhotosynQ

Version 0.2.3

Description Connect R to the PhotosynQ platform (<<https://photosynq.org>>). It allows to login and logout, as well as receive project information and project data. Further it transforms the received JSON objects into a data frame, which can be used for the final data analysis.

License MIT + file LICENCE

Encoding UTF-8

Language en-US

URL <https://github.com/Photosynq/PhotosynQ-R>

BugReports <https://github.com/Photosynq/PhotosynQ-R/issues>

RoxygenNote 7.1.1

Depends R (>= 3.2.4)

Imports httr (>= 1.4.0), getPass (>= 0.2-2)

NeedsCompilation no

Author Sebastian Kuhlert [aut, cre]

Maintainer Sebastian Kuhlert <sebastian.kuhlert@gmail.com>

Repository CRAN

Date/Publication 2021-07-13 14:50:07 UTC

Contents

| | |
|---------------------------|----------|
| createDataframe | 2 |
| getProject | 2 |
| getProjectData | 3 |
| getProjectInfo | 4 |
| login | 5 |
| logout | 6 |
| Index | 7 |

| | |
|-----------------|--|
| createDataframe | <i>Generate a Data Frame from 'PhotosynQ' project data</i> |
|-----------------|--|

Description

This function creates a data frame with the data for a single 'PhotosynQ' Project.

Usage

```
createDataframe(project_info = NULL, project_data = NULL)
```

Arguments

| | |
|--------------|---|
| project_info | Object returned by getProjectInfo |
| project_data | Object returned by getProjectData |

Details

This function allows to create a data frame based on the Project's information ([getProjectInfo](#)) and the Project's data ([getProjectData](#)). If more than one Protocol was used in a Project, the data frame contains multiple frames, each one named after the corresponding measurement Protocol.

Value

Data frame(s) with Project's data for subsequent analysis. In case of issues it will return NULL.

Examples

```
project_info <- getProjectInfo(1566)
project_data <- getProjectData(1566)
createDataframe(project_info, project_data)
```

| | |
|------------|--|
| getProject | <i>Get Project data in a Data Frame from 'PhotosynQ'</i> |
|------------|--|

Description

Get Project data using the Project's ID and create data frame(s).

Usage

```
getProject(projectID = "", processedData = TRUE, rawTraces = FALSE)
```

Arguments

| | |
|---------------|--|
| projectID | The ID of your Project (Just copy the Project ID from the project page or your user page) |
| processedData | (optional) Receive the processed data when set to TRUE, receive raw Data when set to FALSE. |
| rawTraces | (optional) Adds raw traces to processed data. It is ignored when processedData is set to TRUE. |

Details

This function produces a data frame for a 'PhotosynQ' Project using the Project's ID. The ID can be found on the Project's page on the 'PhotosynQ' website. This function call includes the Project data as well as information. In case multiple protocols were used, each protocol is in a separate data frame. By default the processed data is not included. The parameter rawTraces can be set to TRUE to include the rawTraces in combination with the processed data. By default the recorded traces are not received. When the original data structure is needed, separate calls have to be used including [getProjectInfo](#) and [getProjectData](#).

Value

Separate data frame per protocol including measurement data and answers to Project.

Note

Including the raw data and/or the traces will increase the data frame size significantly.

Examples

```
getProject(1566)
```

| | |
|----------------|--|
| getProjectData | <i>Get Project Data from 'PhotosynQ'</i> |
|----------------|--|

Description

Get Project Data using the Project's ID

Usage

```
getProjectData(projectID = "", processedData = TRUE, rawTraces = FALSE)
```

Arguments

| | |
|---------------|--|
| projectID | The ID of your Project (Just copy the Project's ID from the project page or your user page) |
| processedData | (optional) Receive the processed data when set to TRUE, receive raw Data when set to FALSE. |
| rawTraces | (optional) Adds raw traces to processed data. It is ignored when processedData is set to TRUE. |

Details

This function receives the data from 'PhotosynQ' for a specific Project. The received data is in the original 'JSON' structure. When setting the processedDate to FALSE the raw data will be received instead of the processed data. By default the processed data is not included. The parameter rawTraces can be set to TRUE to include the rawTraces in combination with the processed data. By default the traces are not received.

Value

Project data is returned in the 'JSON' format. In case of issues it will return NULL.

Note

Including the raw data and/or the traces will increase the data frame size significantly.

Examples

```
getProjectData(1566)
```

| | |
|----------------|---|
| getProjectInfo | <i>Get Project Information from 'PhotosynQ'</i> |
|----------------|---|

Description

Get Project Information using the Project's ID

Usage

```
getProjectInfo(projectID = "")
```

Arguments

| | |
|-----------|---|
| projectID | The ID of your Project (Just copy the Project ID from the project page or your user page) |
|-----------|---|

Details

This function receives the information about a project from 'PhotosynQ' including used Protocols and Project Questions. The information is also required to process the Project's data for the data frame used in the [createDataframe](#) function.

Value

Project information is returned in the 'JSON' format. In case of issues it will return NULL.

Examples

```
getProjectInfo(1566)
```

| | |
|-------|-----------------------------|
| login | <i>Login to 'PhotosynQ'</i> |
|-------|-----------------------------|

Description

Login to 'PhotosynQ' to allow data access.

Usage

```
login(email = "", url = photosynq.env$DEFAULT_API_DOMAIN)
```

Arguments

| | |
|-------|--|
| email | Your email address you use to login |
| url | (optional) Change the default URL to point to another instance |

Details

This function allows a user to login to 'PhotosynQ' and start a session. The functions [getProjectInfo](#), [getProjectData](#), [getProject](#) and [logout](#) require a session started by login in. A login is only required once at the beginning of a session.

Value

Session key is received and session data is stored as a global variable. Otherwise nothing is returned.

Note

The password needs to be entered in a dialog and gets never saved nor should it be saved anywhere in the code.

Examples

```
## Not run:  
login("john.doe@domain.com")  
  
## End(Not run)
```

| | |
|--------|--------------------------------|
| logout | <i>Logout from 'PhotosynQ'</i> |
|--------|--------------------------------|

Description

Logout from 'PhotosynQ' and end session

Usage

```
logout()
```

Details

This function ends the current session and logs out the current user out from 'PhotosynQ'. Use the login function to sign in again and start a new session if needed.

Value

Session data is removed from the global variables. Nothing is returned

Examples

```
logout()
```

Index

- * **Data**

- getProject, 2
 - getProjectData, 3

- * **Information**

- getProjectInfo, 4

- * **Project**

- createDataframe, 2
 - getProject, 2
 - getProjectData, 3
 - getProjectInfo, 4

- * **data**

- createDataframe, 2

- * **frame**

- createDataframe, 2

- * **login**

- login, 5

- * **logout**

- logout, 6

createDataframe, 2, 4

getProject, 2, 5

getProjectData, 2, 3, 3, 5

getProjectInfo, 2, 3, 4, 5

login, 5

logout, 5, 6