Package ‘brickset’

January 11, 2024

Type Package

Title Interface with the Brickset API for Getting Data About LEGO Sets

Version 2024.0.0

Date 2024-01-11

Maintainer Jason Bryer <jason@bryer.org>

Description Interface with the 'Brickset' API
<https://brickset.com/article/52664/api-version-3-documentation> for
getting data about LEGO sets. Data sets that
can be used for teaching and learning without the need of a 'Brickset'
account and API key are also included. Includes all LEGO since through
the end of 2023.

License GPL (>= 3)

URL https://github.com/jbryer/brickset,
https://jbryer.github.io/brickset/

BugReports https://github.com/jbryer/brickset/issues

Imports dplyr, httr, jsonlite, piggyback

Suggests DT, ggplot2, knitr, shiny

Encoding UTF-8

LazyData TRUE

RoxygenNote 7.2.3

NeedsCompilation no

Author Jason Bryer [aut, cre] (<https://orcid.org/0000-0002-2454-0402>)

Depends R (>= 3.5.0)

Repository CRAN

Date/Publication 2024-01-11 12:40:03 UTC

R topics documented:

---

brickset *R package to interface with the Brickset API for getting data about LEGO.*

Description

R package to interface with the Brickset API for getting data about LEGO.

Details

You can request an API key on the Brickset website here: https://brickset.com/tools/webservices/requestkey/

The API key can be passed as function parameter or may be set globally using:

`options(brickset_key = YOUR_API_KEY)`

Author(s)

`<jason@bryer.org>`

checkKey *Check the Brickset API key.*

Description

You can request an API key on the Brickset website here: https://brickset.com/tools/webservices/requestkey/

Usage

```
checkKey(key = getOption("brickset_key"))
```

Arguments

key	the Brickset API key

Details

The API key can be passed as function parameter or may be set globally using:

`options(brickset_key = YOUR_API_KEY)`

Brickset API documentation is available here: https://brickset.com/article/52664/api-version-3-documentation

Value

TRUE if the API key is valid.

Examples

```
## Not run:
options(brickset_key = 'BRICKSET_KEY',
        brickset_username = 'BRICKSET_UERNAME',
        brickset_password = 'BRICKSET_PASSWORD')
checkKey() # Will return TRUE if the credentials are correct

## End(Not run)
```

checkUserHash *Check the Brickset API key.*

Description

You can request an API key on the Brickset website here: https://brickset.com/tools/webservices/requestkey/

Usage

```
checkUserHash(key = getOption("brickset_key"), userHash)
```

Arguments

`key`	the API key
`userHash`	the user hash returned from login.

Details

Brickset API documentation is available here: https://brickset.com/article/52664/api-version-3-documentation/

Value

TRUE if the API key is fine.

`download_reviews` *Downloads a pre-built reviews data frame.*

Description

This function will return a data frame with all the reviews as of last package build. Since this data frame is larger than what is allowed in CRAN packages, the data files are saved as releases on Github. The `piggyback::pb_upload()` function is used to upload the data frame.

Usage

```
download_reviews(dest = tempdir(), ...)
```

Arguments

`dest`	directory to download the `reviews.rda` file to.
`...`	other parameters passed to `piggyback::pb_download()`.

Details

To see what versions are available use the `piggyback::pb_list(repo = 'jbryer/brickset')` function call. By default the latest version will be returned. For reproducibility you can use the `tag` parameter to return a specific version of the data frame.

Value

a data frame with all the reviews as of the tag date.

getInstructions *Download list of instructions for given set.*

Description

Brickset API documentation is available here: https://brickset.com/article/52664/api-version-3-documentation/

Usage

```
getInstructions(setID, setNumber, key = getOption("brickset_key"), ...)
```

Arguments

setID	the ID of the set (see data(legosets))
setNumber	the set number from on the LEGO box
key	the Brickset API key.
...	other parameters passed to getUserHash including the Brickset username and password if they are not available from getOption('brickset_username') and getOption('brickset_password').

Value

a data.frame with the instructions.

getKeyUsageStats *Get information about frequency of API usage for the given API key.*

Description

Provides information about how frequently the API key has been used.

Usage

```
getKeyUsageStats(key = getOption("brickset_key"))
```

Arguments

key	the API key

Value

a data.frame with the number of times the key has been used.

dateStamp The date

count The number of times the key was used for the given date

Examples

```
## Not run:
options(brickset_key = 'BRICKSET_KEY',
        brickset_username = 'BRICKSET_UERNAME',
        brickset_password = 'BRICKSET_PASSWORD')
getKeyUsageStats()

## End(Not run)
```

getReviews *Downloads reviews for a LEGO set.*

Description

Downloads reviews for a LEGO set.

Usage

```
getReviews(setID, key = getOption("brickset_key"), ...)
```

Arguments

setID	the ID of the set (see data(legosets))
key	the Brickset API key.
...	other parameters passed to getUserHash including the Brickset username and password if they are not available from getOption('brickset_username') and getOption('brickset_password').

Value

a data.frame with the reviews.

author Author of the review

datePosted Date of the review

title Title of the review

review The text of the review

HTML TRUE if the review contains HTML

overall overall rating by the reviewer

parts rating for the parts

buildingExperience rating for the building experience

playability rating for the playability

valueForMoney rating for the value for money

Examples

```
## Not run:
options(brickset_key = 'BRICKSET_KEY',
        brickset_username = 'BRICKSET_UERNAME',
        brickset_password = 'BRICKSET_PASSWORD')
getReviews('31728') # Will return TRUE if the credentials are correct

## End(Not run)
```

---

getSets *Downloads LEGO data from Brickset.*

Description

Brickset API documentation is available here: https://brickset.com/article/52664/api-version-3-documentation/

Usage

```
getSets(year, key = getOption("brickset_key"), ...)
```

Arguments

`year`	the year of data to download.
`key`	the Brickset key.
`...`	other parameters passed to `getUserHash` including the Brickset username and password if they are not available from `getOption('brickset_username')` and `getOption('brickset_password')`.

Value

a data.frame with all sets from the given year.

setID integer; 19409 unique values

number character; 17997 unique values

numberVariant integer; 25 unique values

name character; 16206 unique values

year integer; 54 unique values

theme character; 158 unique values

themeGroup character; 17 unique values

subtheme character; 957 unique values

category character; 7 unique values

released logical; 2 unique values

pieces integer; 1461 unique values

minifigs integer; 34 unique values

bricksetURL character; 19409 unique values

rating numeric; 30 unique values

reviewCount integer; 63 unique values

packagingType character; 19 unique values

availability character; 10 unique values

agerange_min integer; 17 unique values

thumbnailURL character; 18355 unique values

imageURL character; 18355 unique values

US_retailPrice numeric; 154 unique values

US_dateFirstAvailable Date; 979 unique values

US_dateLastAvailable Date; 2197 unique values

UK_retailPrice numeric; 226 unique values

UK_dateFirstAvailable Date; 927 unique values

UK_dateLastAvailable Date; 2068 unique values

CA_retailPrice numeric; 177 unique values

CA_dateFirstAvailable Date; 745 unique values

CA_dateLastAvailable Date; 1880 unique values

DE_retailPrice numeric; 173 unique values

DE_dateFirstAvailable Date; 514 unique values

DE_dateLastAvailable Date; 1252 unique values

height numeric; 248 unique values

width numeric; 290 unique values

depth numeric; 282 unique values

weight numeric; 1107 unique values

getSubthemes *Returns a table of sub-themes for a given theme with number of sets and years active.*

Description

Brickset API documentation is available here: https://brickset.com/article/52664/api-version-3-documentation/

Usage

```
getSubthemes(theme, key = getOption("brickset_key"), ...)
```

Arguments

`theme`	the theme.
`key`	the Brickset API key.
`...`	other parameters passed to `getUserHash` including the Brickset username and password if they are not available from `getOption('brickset_username')` and `getOption('brickset_password')`.

Value

a data.frame with the subthemes.

`getThemes` *Returns a table of themes with number of sets and years active.*

Description

Brickset API documentation is available here: https://brickset.com/article/52664/api-version-3-documentation/

Usage

```
getThemes(key = getOption("brickset_key"), ...)
```

Arguments

`key`	the Brickset API key.
`...`	other parameters passed to `getUserHash` including the Brickset username and password if they are not available from `getOption('brickset_username')` and `getOption('brickset_password')`.

Value

a data.frame with the themes.

Examples

```
## Not run:
options(brickset_key = 'BRICKSET_KEY',
        brickset_username = 'BRICKSET_UERNAME',
        brickset_password = 'BRICKSET_PASSWORD')
getThemes()

## End(Not run)
```

getUserHash *Returns the user hash from Brickset.*

Description

Many of the Brickset API calls require a user to login. This function wraps the login function to managing the user hash returned from login across multiple API calls. It will also ensure that the hash is still valid and if it expired a new hash will be requested.

Usage

```
getUserHash(
  username = getOption("brickset_username"),
  password = getOption("brickset_password"),
  key = getOption("brickset_key")
)
```

Arguments

`username`	the Brickset username.
`password`	the Brickset password.
`key`	the Brickset API key.

Details

Brickset API documentation is available here: https://brickset.com/article/52664/api-version-3-documentation/

Value

the user hash for the current API session.

getYears *Get a list of years for a given theme, with the total number of sets in each.*

Description

Get a list of years for a given theme, with the total number of sets in each.

Usage

```
getYears(theme, key = getOption("brickset_key"), ...)
```

Arguments

`theme`	the theme
`key`	the Brickset API key.
`...`	other parameters passed to `getUserHash` including the Brickset username and password if they are not available from `getOption('brickset_username')` and `getOption('brickset_password')`.

Value

a data.frame with the years of a given theme.

theme Name of the theme

year Year

setCount Number of sets released in the given year and theme

Examples

```
## Not run:
options(brickset_key = 'BRICKSET_KEY',
        brickset_username = 'BRICKSET_UERNAME',
        brickset_password = 'BRICKSET_PASSWORD')
getYears('Architecture')

## End(Not run)
```

`legosets` *Lego sets from 1970 through 2022.*

Description

This dataset was built using the `getSets` function. For working with the LEGO sets data frame this pre-built data is preferred as it will minimize the API requests. Note that the only disadvantage is that the `rating` and `reviewCount` may be out-of-date and inaccurate. The remaining variables are relatively static.

Format

A data.frame.

setID integer; 19409 unique values

number character; 17997 unique values

numberVariant integer; 25 unique values

name character; 16206 unique values

year integer; 54 unique values

theme character; 158 unique values

themeGroup character; 17 unique values

subtheme character; 957 unique values

category character; 7 unique values

released logical; 2 unique values

pieces integer; 1461 unique values

minifigs integer; 34 unique values

bricksetURL character; 19409 unique values

rating numeric; 30 unique values

reviewCount integer; 63 unique values

packagingType character; 19 unique values

availability character; 10 unique values

agerange_min integer; 17 unique values

thumbnailURL character; 18355 unique values

imageURL character; 18355 unique values

US_retailPrice numeric; 154 unique values

US_dateFirstAvailable Date; 979 unique values

US_dateLastAvailable Date; 2197 unique values

UK_retailPrice numeric; 226 unique values

UK_dateFirstAvailable Date; 927 unique values

UK_dateLastAvailable Date; 2068 unique values

CA_retailPrice numeric; 177 unique values

CA_dateFirstAvailable Date; 745 unique values

CA_dateLastAvailable Date; 1880 unique values

DE_retailPrice numeric; 173 unique values

DE_dateFirstAvailable Date; 514 unique values

DE_dateLastAvailable Date; 1252 unique values

height numeric; 248 unique values

width numeric; 290 unique values

depth numeric; 282 unique values

weight numeric; 1107 unique values

Source

https://brickset.com

login *Login to the Brickset API.*

Description

Brickset API documentation is available here: https://brickset.com/article/52664/api-version-3-documentation/

Usage

```
login(
  username = getOption("brickset_username"),
  password = getOption("brickset_password"),
  key = getOption("brickset_key")
)
```

Arguments

`username`	the Brickset username.
`password`	the Brickset password.
`key`	the Brickset API key.

Value

the user hash used for other API calls.

Index