Package 'GetQuandlData'

May 7, 2026

Type Package

Title Fast and Cached Import of Data from 'Quandl' Using the 'json API'

Version 1.0.0

Author Marcelo S. Perlin

Maintainer Marcelo S. Perlin <marceloperlin@gmail.com>

Description Imports time series data from the 'Quandl' database <https://data.nasdaq.com/>. The package uses the 'json api' at <https://data.nasdaq.com/search>, local caching ('memoise' package) and the tidy format by default. Also allows queries of databases, allowing the user to see which time series are available for each database id. In short, it is an alternative to package 'Quandl', with faster data importation in the tidy/long format.

Imports jsonlite, memoise, dplyr, purrr, utils, readr, fs

Depends R (>= 4.0.0)

License GPL-2

BugReports https://github.com/msperlin/GetQuandlData/issues

URL https://github.com/msperlin/GetQuandlData/

Encoding UTF-8

RoxygenNote 7.2.3

Suggests knitr, rmarkdown, testthat (>= 3.0.0), ggplot2, tibble

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Repository CRAN

Date/Publication 2023-02-15 12:40:04 UTC

Contents

get_cache_folder 2
get_database_info 2
get_Quandl_series 3
get_single_Quandl 4
json_to_tibble 5

Index **7**

`get_cache_folder` *Returns the default cache folder*

Description

Returns the default cache folder

Usage

```
get_cache_folder()
```

Value

a path (temporary)

Examples

```
get_cache_folder()
```

`get_database_info` *Get inform about quandl database*

Description

Uses metadata link to download information about available series and dates for a given database id.

Usage

```
get_database_info(db_in, api_key)
```

Arguments

`db_in`	Database id (e.g. "RATEINF")
`api_key`	YOUR api key

Value

A dataframe

Examples

```
db_in <- 'RATEINF'
api_key <- 'YOUR_API_HERE'

## Not run:
df_db <- get_database_info(db_in, api_key)

## End(Not run)
```

`get_Quandl_series` *Import data from Quandl API*

Description

Uses the json api from Quandl (<https://www.quandl.com/tools/api>) to import data into an R session. The great benefit from the original Quandl::Quandl is the use of package memoise to cache results, organization of the output dataframe in the tidy/long format and passing different multiple parameters to manipulate series.

Usage

```
get_Quandl_series(
  id_in,
  api_key = NULL,
  first_date = Sys.Date() - 365,
  last_date = Sys.Date(),
  do_cache = TRUE,
  order = "asc",
  collapse = "none",
  transform = "none",
  cache_folder = get_cache_folder()
)
```

Arguments

`id_in`	Character vector of ids to grab data. When using a named vector, the name is used to register the time series. Example: id_in <- c('US GDP' = 'FRED/GDP')
`api_key`	YOUR api key (get your own at <https://www.quandl.com/sign-up-modal?defaultModal=showSignUp>)
`first_date`	First date of all requested series as YYYY-MM-DD (default = Sys.date() - 365)
`last_date`	Last date of all requested series as YYYY-MM-DD (default = Sys.date() - 365)
`do_cache`	Do cache? TRUE (default) or FALSE. Sets the use of package memoise to cache results from the api

order	How to order the time series data: 'desc' (descending dates, default) or 'asc' (ascending)
collapse	Frequency of time series: 'none' (default), 'daily', 'weekly', 'monthly', 'quarterly', 'annual'
transform	Quandl transformation: 'none', 'diff', 'rdiff', 'rdiff_from', 'cumul', 'normalize'. Details at <https://docs.quandl.com/docs/parameters-2>
cache_folder	Folder where to save memoise cache files (temporary folder as default)

Details

ATTENTION: You'll need a api key in order to use this function. Get one at <https://www.quandl.com/sign-up-modal?defaultModal=showSignUp>.

Value

A dataframe in the long format

Examples

```
api_key <- 'YOUR_API_KEY_HERE'
id_in <- c('Inflation Canada' = 'RATEINF/INFLATION_CAN')
## Not run:
 df <- get_Quandl_series(id_in = id_in, api_key = api_key)

## End(Not run)
```

---

get_single_Quandl *Fetches a single time series from Quandl*

Description

Fetches a single time series from Quandl

Usage

```
get_single_Quandl(
  id_in,
  name_in,
  api_key,
  first_date,
  last_date,
  do_cache = TRUE,
  order = "asc",
  collapse = "none",
  transform = "none"
)
```

Arguments

id_in	Character vector of ids to grab data. When using a named vector, the name is used to register the time series. Example: id_in <- c('US GDP' = 'FRED/GDP')
name_in	Name of series to fetch
api_key	YOUR api key (get your own at <https://www.quandl.com/sign-up-modal?defaultModal=showSignUp>)
first_date	First date of all requested series as YYYY-MM-DD (default = Sys.date() - 365)
last_date	Last date of all requested series as YYYY-MM-DD (default = Sys.date() - 365)
do_cache	Do cache? TRUE (default) or FALSE. Sets the use of package memoise to cache results from the api
order	How to order the time series data: 'desc' (descending dates, default) or 'asc' (ascending)
collapse	Frequency of time series: 'none' (default), 'daily', 'weekly', 'monthly', 'quarterly', 'annual'
transform	Quandl transformation: 'none', 'diff', 'rdiff', 'rdiff_from', 'cumul', 'normalize'. Details at <https://docs.quandl.com/docs/parameters-2>

Value

A single dataframe

Examples

```
api_key <- 'YOUR_API_KEY_HERE'
id_in <- c('Inflation argentina' = 'RATEINF/INFLATION_ARG')
## Not run:
 df <- get_single_Quandl(id_in = id_in, name_in = '',
                         api_key = api_key,
                         first_date = '2010-01-01',
                         last_date = Sys.Date())

## End(Not run)
```

---

json_to_tibble *Transforms and organize json output to a tibble*

Description

Transforms and organize json output to a tibble

Usage

```
json_to_tibble(l_in, id_in, name_in)
```

Arguments

l_in	Output of get_single_Quandl
id_in	Value of id
name_in	Name of id

Value

A beautiful dataframe

Examples

```
## Not run:
my_id <- 'BCB/7832'
my_api <- 'YOURAPIHERE'
json_link <- sprintf(
                paste0('https://www.quandl.com/api/v3/datasets/%s',
                       '.json?start_date=2010-01-01?end_date=2019-09-30?',
                       'order=asc?collapse=none?transform=none?api_key=%s'),
                    my_id, my_api)
l_out <- jsonlite::fromJSON(json_link)
df <- json_to_tibble(l_out, id_in = my_id, name_in = 'Ibov change')

## End(Not run)
```

Index

get_cache_folder, 2
get_database_info, 2
get_Quandl_series, 3
get_single_Quandl, 4

json_to_tibble, 5