

Package ‘eodhdR2’

July 29, 2025

Type Package

Title Official R API for Fetching Data from 'EODHD'

Version 0.5.2

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

Description Second and backward-incompatible version of R package 'eodhd' <<https://eodhd.com/>>, extended with a cache and quota system, also offering functions for cleaning and aggregating the financial data.

License MIT + file LICENSE

URL <https://github.com/EodHistoricalData/R-Library-for-financial-data-2024>

BugReports

<https://github.com/EodHistoricalData/R-Library-for-financial-data-2024/issues>

Encoding UTF-8

RoxygenNote 7.3.2

Suggests knitr, rmarkdown, cli, dplyr, fs, glue, httr, jsonlite, lubridate, purrr, readr, tidyr, testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Marcelo S. Perlin [aut, cre, ctr],
Unicorn Data Services [cph]

Depends R (>= 4.1.0)

Repository CRAN

Date/Publication 2025-07-29 13:10:02 UTC

Contents

get_demo_token	2
get_dividends	2
get_exchanges	3
get_fundamentals	4
get_intraday	5

get_ipos	6
get_news	7
get_prices	9
get_splits	10
get_tickers	11
parse_financials	11
set_token	12

Index	14
--------------	-----------

get_demo_token	Returns token for demonstration
----------------	---------------------------------

Description

Returns token for demonstration

Usage

get_demo_token()

Value

A string with token

Examples

get_demo_token()

get_dividends	Retrieves dividend data from the api
---------------	--------------------------------------

Description

This function will query the dividend end point <https://eodhd.com/financial-apis/api-splits-dividends> and return:

- dates (declaration, record, payment)
- value of dividend (adjusted and unadjusted)
- currency of dividend

Usage

```
get_dividends(  
    ticker = "AAPL",  
    exchange = "US",  
    cache_folder = get_default_cache(),  
    check_quota = TRUE  
)
```

Arguments

ticker	A company ticker (e.g. AAPL). You can find all tickers for a particular exchange with <code>get_tickers()</code> .
exchange	A exchange symbol (e.g. US). You can find all tickers for a particular exchange with <code>get_tickers()</code> . Be aware that, for US companies, the exchange symbols is simply "US"
cache_folder	A local directory to store cache files. By default, all functions use a temporary path, meaning that the caching system is session persistent (it will remove all files when you exit your R session). If you want a persistent caching system, simply point argument <code>cache_folder</code> to a local directory in your filesystem. Be aware, however, that a persistent cache will not refresh your data for new api queries.
check_quota	A flag (TRUE/FALSE) for whether to check the current quota status from the api. This option implies a small cost of execution time. If you need speed, just set it to FALSE.

Value

A dataframe with dividend information

Examples

```
## Not run:
set_token(get_demo_token())
df_div <- get_dividends(ticker = "AAPL", exchange = "US")
df_div

## End(Not run)
```

get_exchanges	<i>Retrieves the list of available exchanges</i>
---------------	--

Description

Retrieves the list of available exchanges

Usage

```
get_exchanges(cache_folder = get_default_cache())
```

Arguments

cache_folder	A local directory to store cache files. By default, all functions use a temporary path, meaning that the caching system is session persistent (it will remove all files when you exit your R session). If you want a persistent caching system, simply point argument <code>cache_folder</code> to a local directory in your filesystem. Be aware, however, that a persistent cache will not refresh your data for new api queries.
--------------	---

Value

a dataframe with information about available exchanges

Examples

```
# you need a valid token (not test) for this to work
## Not run:
set_token("YOUR_VALID_TOKEN")
df_exc <- get_exchanges()

## End(Not run)
```

get_fundamentals	<i>Retrieves and parses fundamental and financial data from eodhd api</i>
------------------	---

Description

This function will download raw data from the fundamental end point of eodhd <https://eodhd.com/financial-apis/stock-etfs-fundamental-data-feeds> and return a list. The raw data includes:

- General information for the company (code, ISIN, currency, ..)
- Financial highlights
- Valuation
- Raw financial data (see [parse_financials\(\)](#) for parsing this data)
- and many more (see example for more details regarding the output)

Usage

```
get_fundamentals(
  ticker = "AAPL",
  exchange = "US",
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

ticker	A company ticker (e.g. AAPL). You can find all tickers for a particular exchange with get_tickers() .
exchange	A exchange symbol (e.g. US). You can find all tickers for a particular exchange with get_tickers() . Be aware that, for US companies, the exchange symbols is simply "US"

- `cache_folder` A local directory to store cache files. By default, all functions use a temporary path, meaning that the caching system is session persistent (it will remove all files when you exit your R session). If you want a persistent caching system, simply point argument `cache_folder` to a local directory in your filesystem. Be aware, however, that a persistent cache will not refresh your data for new api queries.
- `check_quota` A flag (TRUE/FALSE) for whether to check the current quota status from the api. This option implies a small cost of execution time. If you need speed, just set it to FALSE.

Value

a list with several fundamental information

Examples

```
## Not run:
set_token(get_demo_token())
l_out <- get_fundamentals(ticker = "AAPL", exchange = "US")
names(l_out)

## End(Not run)
```

<code>get_intraday</code>	<i>Retrieves intraday data for a given ticker and exchange</i>
---------------------------	--

Description

This function will query the intraday endpoint of eodhd and return all data for a user supplied time period.

Usage

```
get_intraday(
  ticker = "AAPL",
  exchange = "US",
  frequency = "5m",
  first_date = Sys.Date() - 7,
  last_date = Sys.Date(),
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

ticker	A company ticker (e.g. AAPL). You can find all tickers for a particular exchange with <code>get_tickers()</code> .
exchange	A exchange symbol (e.g. US). You can find all tickers for a particular exchange with <code>get_tickers()</code> . Be aware that, for US companies, the exchange symbols is simply "US"
frequency	The frequency of the intraday data. Available options: "1m" = 1 minute, "5m" = 5 minutes, "1h" = 1 hour
first_date	the first date to fetch news. The function will keep querying the api until this date is reached. Default is previous week.
last_date	the last date to fetch news. Default is today.
cache_folder	A local directory to store cache files. By default, all functions use a temporary path, meaning that the caching system is session persistent (it will remove all files when you exit your R session). If you want a persistent caching system, simply point argument cache_folder to a local directory in your filesystem. Be aware, however, that a persistent cache will not refresh your data for new api queries.
check_quota	A flag (TRUE/FALSE) for whether to check the current quota status from the api. This option implies a small cost of execution time. If you need speed, just set it to FALSE.

Value

A dataframe with news events and sentiments

Examples

```
## Not run:
set_token(get_demo_token())
df_intraday <- get_intraday(ticker = "AAPL", exchange = "US")

## End(Not run)
```

get_ipos

Retrieves IPO (Initial Public Offering) data for a given time period

Description

This function will query the IPO end point of eodhd and return all ipos for a user supplied time period.

Usage

```
get_ipos(
  first_date = Sys.Date() - 3 * 365,
  last_date = Sys.Date(),
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

first_date	the first date to fetch ipos information. Default is previous three years
last_date	the last date to fetch news. Default is today.
cache_folder	A local directory to store cache files. By default, all functions use a temporary path, meaning that the caching system is session persistent (it will remove all files when you exit your R session). If you want a persistent caching system, simply point argument cache_folder to a local directory in your filesystem. Be aware, however, that a persistent cache will not refresh your data for new api queries.
check_quota	A flag (TRUE/FALSE) for whether to check the current quota status from the api. This option implies a small cost of execution time. If you need speed, just set it to FALSE.

Value

A dataframe with news events and sentiments

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_news <- get_ipos()

## End(Not run)
```

get_news

Retrieves news for a given ticker and exchange

Description

This function will query the news point of eodhd and return all news for a user supplied time period.

Usage

```
get_news(
  ticker = "AAPL",
  exchange = "US",
  first_date = Sys.Date() - 3 * 30,
  last_date = Sys.Date(),
  offset_delta = 500,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

ticker	A company ticker (e.g. AAPL). You can find all tickers for a particular exchange with get_tickers() .
exchange	A exchange symbol (e.g. US). You can find all tickers for a particular exchange with get_tickers() . Be aware that, for US companies, the exchange symbols is simply "US"
first_date	the first date to fetch news. The function will keep querying the api until this date is reached. Default is previous three months.
last_date	the last date to fetch news. Default is today.
offset_delta	how much to change offset in each iterations (higher values will result in more query time, but less queries). Default is 500.
cache_folder	A local directory to store cache files. By default, all functions use a temporary path, meaning that the caching system is session persistent (it will remove all files when you exit your R session). If you want a persistent caching system, simply point argument cache_folder to a local directory in your filesystem. Be aware, however, that a persistent cache will not refresh your data for new api queries.
check_quota	A flag (TRUE/FALSE) for whether to check the current quota status from the api. This option implies a small cost of execution time. If you need speed, just set it to FALSE.

Value

A dataframe with news events and sentiments

Examples

```
## Not run:
set_token(get_demo_token())
df_news <- get_news(ticker = "AAPL", exchange = "US")

## End(Not run)
```

`get_prices`*Retrieves adjusted and unadjusted stock prices*

Description

This function will query the price end point of eodhd and return daily stock price from a set of ticker and exchange. It also includes the daily stock return (percentage variation).

Usage

```
get_prices(  
  ticker = "AAPL",  
  exchange = "US",  
  cache_folder = get_default_cache(),  
  check_quota = TRUE  
)
```

Arguments

ticker	A company ticker (e.g. AAPL). You can find all tickers for a particular exchange with get_tickers() .
exchange	A exchange symbol (e.g. US). You can find all tickers for a particular exchange with get_tickers() . Be aware that, for US companies, the exchange symbols is simply "US"
cache_folder	A local directory to store cache files. By default, all functions use a temporary path, meaning that the caching system is session persistent (it will remove all files when you exit your R session). If you want a persistent caching system, simply point argument cache_folder to a local directory in your filesystem. Be aware, however, that a persistent cache will not refresh your data for new api queries.
check_quota	A flag (TRUE/FALSE) for whether to check the current quota status from the api. This option implies a small cost of execution time. If you need speed, just set it to FALSE.

Value

A dataframe with prices

Examples

```
## Not run:  
set_token(get_demo_token())  
df_prices <- get_prices(ticker = "AAPL", exchange = "US")  
  
## End(Not run)
```

get_splits

Retrieves splits data from eodhd

Description

This function will query the splits end point of eodhd and return all split information for a given stock/exchange.

Usage

```
get_splits(
  ticker = "AAPL",
  exchange = "US",
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

ticker	A company ticker (e.g. AAPL). You can find all tickers for a particular exchange with get_tickers() .
exchange	A exchange symbol (e.g. US). You can find all tickers for a particular exchange with get_tickers() . Be aware that, for US companies, the exchange symbols is simply "US"
cache_folder	A local directory to store cache files. By default, all functions use a temporary path, meaning that the caching system is session persistent (it will remove all files when you exit your R session). If you want a persistent caching system, simply point argument cache_folder to a local directory in your filesystem. Be aware, however, that a persistent cache will not refresh your data for new api queries.
check_quota	A flag (TRUE/FALSE) for whether to check the current quota status from the api. This option implies a small cost of execution time. If you need speed, just set it to FALSE.

Value

A dataframe with split information

Examples

```
## Not run:
# requires a subscription (paid) token
df_split <- get_splits(ticker = "AAPL", exchange = "US")

## End(Not run)
```

get_tickers	<i>Retrieves a list of tickers for a particular exchange</i>
-------------	--

Description

Retrieves a list of tickers for a particular exchange

Usage

```
get_tickers(exchange = "US", cache_folder = get_default_cache())
```

Arguments

exchange	A exchange symbol (e.g. US). You can find all tickers for a particular exchange with get_tickers() . Be aware that, for US companies, the exchange symbols is simply "US"
cache_folder	A local directory to store cache files. By default, all functions use a temporary path, meaning that the caching system is session persistent (it will remove all files when you exit your R session). If you want a persistent caching system, simply point argument cache_folder to a local directory in your filesystem. Be aware, however, that a persistent cache will not refresh your data for new api queries.

Value

A dataframe with a list of tickers

Examples

```
## Not run: # requires a subscription (paid) token
df_tickers <- get_tickers("US")

## End(Not run)
```

parse_financials	<i>Parses financial data from the API</i>
------------------	---

Description

This function will organize the raw financial data from [get_fundamentals\(\)](#), aggregating all information into a single dataframe, including quarterly and yearly data from the Balance_sheet, Cash-flow statement and Income statement. Whenever no financial data is found in l_out, the function returns an empty dataframe.

Usage

```
parse_financials(l_out, type_table = "long")
```

Arguments

l_out	A list with raw data (output from <code>get_fundamentals()</code>)
type_table	Format of table in output ("wide" or "long"). A "wide" table is a typical Excel column-oriented table where each columns is a data/year. A long type of table row-oriented, where each each point of new information is a row of the table. The data is the same, it just changes the orientation of rows/columns. The default value is a "long" table.

Value

A dataframe with organized financial data in the wide or long format

Examples

```
## Not run:
set_token(get_demo_token())
l_out <- get_fundamentals(ticker = "AAPL", exchange = "US")

df_fin <- parse_financials(l_out, "long")
df_fin

## End(Not run)
```

set_token	<i>Sets up authentication for eodhd</i>
-----------	---

Description

Uses the token from <https://eodhd.com/cp/dashboard> to authenticate your R session. You can find your own eodhd token from the website. Alternatively, a demo token is also available for testing purposes, with a limited supply of data.

Usage

```
set_token(token = get_demo_token())
```

Arguments

token	the token from eodhd. The default value is a demo token "demo", which allows for partial access to the data. See <code>get_demo_token()</code> for using a demo token.
-------	--

Value

Nothing

Examples

```
## Not run:  
set_token()  
  
## End(Not run)
```

Index

`get_demo_token`, [2](#)
`get_demo_token()`, [12](#)
`get_dividends`, [2](#)
`get_exchanges`, [3](#)
`get_fundamentals`, [4](#)
`get_fundamentals()`, [11](#), [12](#)
`get_intraday`, [5](#)
`get_ipos`, [6](#)
`get_news`, [7](#)
`get_prices`, [9](#)
`get_splits`, [10](#)
`get_tickers`, [11](#)
`get_tickers()`, [3](#), [4](#), [6](#), [8–11](#)

`parse_financials`, [11](#)
`parse_financials()`, [4](#)

`set_token`, [12](#)