

Package ‘eodhdR2’

September 22, 2026

Type Package

Title Official R API for Fetching Data from 'EODHD'

Version 0.8

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

Depends R (>= 4.1.0)

Imports cli, dplyr, fs, glue, httr, jsonlite, lubridate, purrr, readr, tidyr

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

Config/roxygen2/version 8.1.0

NeedsCompilation no

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

Repository CRAN

Date/Publication 2026-09-22 21:40:02 UTC

Contents

get_bulk_eod	3
get_bulk_fundamentals	4
get_cds_market_aggregates	5

get_commodities	6
get_corporate_cmdi	7
get_corporate_hqm_yields	8
get_default_spreads	10
get_demo_token	11
get_dividends	11
get_dividends_calendar	12
get_earnings	13
get_earnings_trends	14
get_economic_events	15
get_eodhd	16
get_exchanges	17
get_exchange_details	18
get_fundamentals	19
get_funding_stress_spreads	20
get_historical_market_cap	21
get_id_mapping	23
get_index_composition	24
get_index_list	25
get_insider_transactions	25
get_intraday	27
get_ipos	28
get_macro_indicator	29
get_news	30
get_news_word_weights	31
get_options_contracts	32
get_options_eod	33
get_options_underlyings	35
get_policy_rates	36
get_prices	37
get_real_estate	38
get_real_estate_countries	39
get_real_estate_detailed	40
get_real_estate_series	42
get_real_time	43
get_reference_rates	44
get_sanctions_entities	45
get_sanctions_programs	46
get_sanctions_sources	47
get_sanctions_vessels	48
get_screener	49
get_search	50
get_sentiments	51
get_sovereign_cds_spreads	52
get_sovereign_credit_ratings	53
get_sovereign_risk_premium	54
get_splits	55
get_splits_calendar	56

get_bulk_eod 3

<i>get_symbol_change_history</i>	57
<i>get_technical</i>	58
<i>get_tickers</i>	60
<i>get_ticks</i>	60
<i>get_user_info</i>	62
<i>get_ust_rates</i>	62
<i>get_us_quote_delayed</i>	63
<i>parse_financials</i>	64
<i>set_token</i>	65

Index 66

<i>get_bulk_eod</i>	<i>Retrieves end of day data for a whole exchange in a single call</i>
---------------------	--

Description

Queries the bulk endpoint <https://eodhd.com/financial-apis/bulk-api-eod-splits-dividends>, which returns the last trading day of every ticker of an exchange (or of the tickers given in argument symbols).

Usage

```
get_bulk_eod(  
  exchange = "US",  
  date = NULL,  
  type = "eod",  
  symbols = NULL,  
  cache_folder = get_default_cache(),  
  check_quota = TRUE  
)
```

Arguments

<i>exchange</i>	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"
<i>date</i>	an optional date. Default (NULL) is the last trading day.
<i>type</i>	one of "eod" (default), "splits" or "dividends"
<i>symbols</i>	an optional character vector of tickers, without the exchange (e.g. c("AAPL", "MSFT"))
<i>cache_folder</i>	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 <i>cache_folder</i> 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 one row per ticker

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_bulk <- get_bulk_eod("US", symbols = c("AAPL", "MSFT"))

## End(Not run)
```

`get_bulk_fundamentals` *Retrieves the fundamentals of a whole exchange*

Description

Queries the bulk fundamentals endpoint <https://eodhd.com/financial-apis/bulk-api-eod-splits-dividends>, which returns the fundamental data of many companies in a single call. Mind the cost: each company of the response is charged as a fundamentals call.

Usage

```
get_bulk_fundamentals(
  exchange = "US",
  symbols = NULL,
  limit = 10,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

<code>exchange</code>	the exchange code (e.g. "US")
<code>symbols</code>	an optional character vector of tickers (e.g. <code>c("AAPL", "MSFT")</code>). Default (NULL) returns the companies of the page.
<code>limit</code>	number of companies of the page (default 10, maximum 1000)
<code>offset</code>	pagination offset (default 0)

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 named list, one element per company, in the same shape as [get_fundamentals\(\)](#)

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
l_bulk <- get_bulk_fundamentals("US", symbols = c("AAPL", "MSFT"))

## End(Not run)
```

get_cds_market_aggregates

Retrieves the aggregates of the cds market

Description

Queries the credit risk endpoints of eodhd for the aggregated metrics of the credit default swap market, by dimension (sector, region, rating).

Usage

```
get_cds_market_aggregates(
  metric = NULL,
  dimension = NULL,
  value = NULL,
  region = NULL,
  first_date = NULL,
  last_date = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

metric	an optional metric to aggregate
dimension	an optional dimension of the aggregation (e.g. "sector")
value	an optional value of the dimension (e.g. "Energy")
region	an optional region
first_date	an optional first date of the window
last_date	an optional last date of the window
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the cds market aggregates

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_agg <- get_cds_market_aggregates()

## End(Not run)
```

get_commodities	<i>Retrieves the historical prices of a commodity</i>
-----------------	---

Description

Queries the commodities endpoint of eodhd, which serves the daily, weekly and monthly series of energy, metals and agricultural commodities.

Usage

```
get_commodities(
  code = "WTI",
  interval = "daily",
  first_date = NULL,
  last_date = NULL,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

code	the code of the commodity (e.g. "WTI", "BRENT", "GOLD")
interval	"daily" (default), "weekly" or "monthly"
first_date	an optional first date of the window
last_date	an optional last date of the window
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 the historical prices of the commodity

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_wti <- get_commodities("WTI")

## End(Not run)
```

get_corporate_cmdi	<i>Retrieves the corporate market distress index (cmdi)</i>
--------------------	---

Description

Queries the credit risk endpoints of eodhd for the corporate market distress index, a daily measure of the stress of the corporate bond market.

Usage

```
get_corporate_cmdi(
  first_date = NULL,
  last_date = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

first_date	an optional first date of the window
last_date	an optional last date of the window
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the cmdi series

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_cmdi <- get_corporate_cmdi()

## End(Not run)
```

```
get_corporate_hqm_yields
```

Retrieves the high quality market (hqm) corporate bond yields

Description

Queries the credit risk endpoints of eodhd for the hqm corporate yield curve, by tenor.

Usage

```
get_corporate_hqm_yields(  
  tenor = NULL,  
  type = NULL,  
  first_date = NULL,  
  last_date = NULL,  
  limit = 100,  
  offset = 0,  
  cache_folder = get_default_cache(),  
  check_quota = TRUE  
)
```

Arguments

tenor	an optional tenor in years (e.g. 10)
type	an optional curve type
first_date	an optional first date of the window
last_date	an optional last date of the window
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the hqm yields

Examples

```
## Not run:  
set_token("YOUR_VALID_TOKEN")  
df_hqm <- get_corporate_hqm_yields(tenor = 10)  
  
## End(Not run)
```

get_default_spreads	<i>Retrieves the default spreads by rating</i>
---------------------	--

Description

Queries the credit risk endpoints of eodhd for the spread over the risk free rate of each rating bucket.

Usage

```
get_default_spreads(
  rating = NULL,
  as_of = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

rating	an optional rating (e.g. "Baa2")
as_of	an optional reference date. Default (NULL) is the latest release.
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the default spreads

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_spreads <- get_default_spreads()

## End(Not run)
```

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_dividends_calendar

Retrieves the dividend calendar

Description

Queries the dividend calendar endpoint of eodhd. The api requires either a symbol or a single date, so one of arguments `symbol` and `date` must be given.

Usage

```
get_dividends_calendar(
  symbol = NULL,
  date = NULL,
  first_date = NULL,
  last_date = NULL,
  limit = 100,
```

```

    offset = 0,
    cache_folder = get_default_cache(),
    check_quota = TRUE
  )

```

Arguments

symbol	a ticker with exchange (e.g. "AAPL.US")
date	a single date. Used when symbol is NULL.
first_date	an optional first date, to narrow the output of a symbol query
last_date	an optional last date, to narrow the output of a symbol query
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the dividend calendar

Examples

```

## Not run:
set_token("YOUR_VALID_TOKEN")
df_div <- get_dividends_calendar(symbol = "AAPL.US")

## End(Not run)

```

get_earnings	<i>Retrieves the earnings calendar</i>
--------------	--

Description

Queries <https://eodhd.com/financial-apis/calendar-upcoming-earnings-ipos-and-splits> and returns reported and upcoming earnings, with estimate, actual value and surprise.

Usage

```
get_earnings(
  first_date = Sys.Date(),
  last_date = Sys.Date() + 30,
  symbols = NULL,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

<code>first_date</code>	the first date of the window. Default is today.
<code>last_date</code>	the last date of the window. Default is in one month.
<code>symbols</code>	an optional character vector of tickers with exchange (e.g. <code>c("AAPL.US", "MSFT.US")</code>). Default (NULL) returns every company.
<code>cache_folder</code>	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.
<code>check_quota</code>	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 the earnings calendar

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_earnings <- get_earnings(symbols = "AAPL.US")

## End(Not run)
```

<code>get_earnings_trends</code>	<i>Retrieves analyst estimates and their revisions</i>
----------------------------------	--

Description

Queries the earning trends endpoint <https://eodhd.com/financial-apis/calendar-upcoming-earnings-ipos-and-s> which returns, for each period, the average analyst estimate for earnings and revenue, together with the revisions of the last months.

Usage

```
get_earnings_trends(
  symbols = "AAPL.US",
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

symbols	a character vector of tickers with exchange (e.g. c("AAPL.US", "MSFT.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 the analyst estimates

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_trends <- get_earnings_trends("AAPL.US")

## End(Not run)
```

get_economic_events	<i>Retrieves the economic calendar</i>
---------------------	--

Description

Queries <https://eodhd.com/financial-apis/macro-economics-data-and-macro-indicators-api> for the scheduled and released macroeconomic events, with actual, estimate and previous values.

Usage

```
get_economic_events(
  first_date = Sys.Date(),
  last_date = Sys.Date() + 30,
  country = NULL,
  comparison = NULL,
```

```

    limit = 50,
    offset = 0,
    cache_folder = get_default_cache(),
    check_quota = TRUE
  )

```

Arguments

first_date	the first date of the window. Default is today.
last_date	the last date of the window. Default is in one month.
country	an optional country code in ISO 3166-1 alpha 2 (e.g. "US")
comparison	an optional comparison period ("mom", "qoq" or "yoy")
limit	maximum number of events (default 50, maximum 1000)
offset	pagination offset (default 0)
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 the economic events

Examples

```

## Not run:
set_token("YOUR_VALID_TOKEN")
df_events <- get_economic_events(country = "US")

## End(Not run)

```

get_eodhd

Retrieves data from any eodhd endpoint

Description

A low level escape hatch for the endpoints that do not (yet) have a dedicated function in the package. See <https://eodhd.com/financial-apis/> for the full list of endpoints and their parameters. The api token is added for you.

Usage

```
get_eodhd(
  endpoint,
  params = list(),
  as_dataframe = TRUE,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

endpoint	the endpoint path, without the base url and without the token (e.g. "eod/AAPL.US", "real-estate/US", "sanctions/entities")
params	a named list of query parameters (e.g. list(from = "2026-01-01"))
as_dataframe	if TRUE (default) the response is parsed into a dataframe. Set it to FALSE for endpoints that return a nested object (the raw list is returned instead).
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 (or a list, when as_dataframe = FALSE)

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_rates <- get_eodhd("ust/yield-rates", list(`filter[year]` = 2026))

## End(Not run)
```

get_exchanges

Retrieves the list of available exchanges

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 `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 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_exchange_details	<i>Retrieves the details of an exchange</i>
----------------------	---

Description

Queries <https://eodhd.com/financial-apis/exchanges-api-list-of-tickers-and-trading-hours> and returns trading hours, holidays and the number of active tickers of an exchange.

Usage

```
get_exchange_details(  
  exchange = "US",  
  first_date = NULL,  
  last_date = NULL,  
  cache_folder = get_default_cache(),  
  check_quota = TRUE  
)
```

Arguments

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"
first_date	optional first date of the holiday window
last_date	optional last date of the holiday window
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 list with the exchange details (including dataframes of trading hours and holidays)

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
l_details <- get_exchange_details("US")

## 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 <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 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)
```

get_funding_stress_spreads

Retrieves the funding stress spreads

Description

Queries the funding stress endpoint of eodhd, with the spreads used to monitor the stress of the money market.

Usage

```
get_funding_stress_spreads(
  code = NULL,
  first_date = NULL,
  last_date = NULL,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

code	an optional code of the spread
first_date	an optional first date of the window
last_date	an optional last date of the window
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 the funding stress spreads

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_spreads <- get_funding_stress_spreads()

## End(Not run)
```

get_historical_market_cap

Retrieves the historical market capitalisation of a company

Description

Queries <https://eodhd.com/financial-apis/stock-etfs-fundamental-data-feeds> (historical market capitalisation), which returns the daily market cap of a company. History is limited to the last year for most subscriptions.

Usage

```
get_historical_market_cap(  
  ticker = "AAPL",  
  exchange = "US",  
  first_date = Sys.Date() - 365,  
  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 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 of the window. Default is one year ago.
last_date	the last date of the window. 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 columns date and value

Examples

```
## Not run:  
set_token("YOUR_VALID_TOKEN")  
df_cap <- get_historical_market_cap("AAPL", "US")  
  
## End(Not run)
```

get_id_mapping	<i>Maps a security between identifiers (ticker, isin, figi, lei, cusip, cik)</i>
----------------	--

Description

Queries the id mapping endpoint of eodhd. Give it one identifier and it returns all the others.

Usage

```
get_id_mapping(  
  symbol = NULL,  
  exchange = NULL,  
  isin = NULL,  
  figi = NULL,  
  lei = NULL,  
  cusip = NULL,  
  cik = NULL,  
  limit = 50,  
  offset = 0,  
  cache_folder = get_default_cache(),  
  check_quota = TRUE  
)
```

Arguments

symbol	an optional ticker (e.g. "AAPL")
exchange	an optional exchange code, used together with symbol
isin, figi, lei, cusip, cik	optional identifiers, one is enough
limit	maximum number of rows (default 50)
offset	pagination offset (default 0)
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 the matching identifiers

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_ids <- get_id_mapping(symbol = "AAPL")

## End(Not run)
```

get_index_composition *Get Index Composition*

Description

Retrieves the current and historical components of a financial index from the EODHD API. Uses a caching system to avoid redundant API calls.

Usage

```
get_index_composition(index, cache_folder = get_default_cache())
```

Arguments

index	index to find composition (e.g. "GSPC.INDX")
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 list with three elements: info, current_components, historical_components

Examples

```
## Not run:
l_out <- get_index_composition("GSPC.INDX")
str(l_out)

## End(Not run)
```

get_index_list	<i>Retrieves the list of indices with composition data</i>
----------------	--

Description

Queries the S&P Global data of the eodhd marketplace and returns the indices that have components available for `get_index_composition()`.

Usage

```
get_index_list(cache_folder = get_default_cache(), check_quota = TRUE)
```

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 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 the available indices

Examples

```
## Not run:  
set_token("YOUR_VALID_TOKEN")  
df_indices <- get_index_list()  
  
## End(Not run)
```

get_insider_transactions	<i>Retrieves insider transactions</i>
--------------------------	---------------------------------------

Description

Queries <https://eodhd.com/financial-apis/insider-transactions-api>, with the purchases and sales reported by the insiders of US companies (SEC form 4).

Usage

```
get_insider_transactions(
  ticker = NULL,
  exchange = "US",
  first_date = Sys.Date() - 90,
  last_date = Sys.Date(),
  limit = 100,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

ticker	an optional company ticker (e.g. "AAPL"). Default (NULL) returns the transactions of every company in the window.
exchange	the exchange of the ticker (default "US")
first_date	the first date of the window. Default is 90 days ago.
last_date	the last date of the window. Default is today.
limit	maximum number of rows (default 100, maximum 1000)
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 the insider transactions

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_insider <- get_insider_transactions("AAPL")

## End(Not run)
```

`get_intraday`*Retrieves intraday data for a given ticker and exchange*

Description

This function will query the intraday endpoint of eodhd <https://eodhd.com/financial-apis/intraday-historical-data-api> and return all bars available within a user supplied time period. Since the api limits how far back a single query can reach, the function walks backwards in time, querying consecutive windows until the first date is covered.

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 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"
frequency	The frequency of the intraday data. Available options: "1m" = 1 minute, "5m" = 5 minutes, "1h" = 1 hour
first_date	the first date of the query window. The function will keep querying the api until this date is reached. Default is previous week.
last_date	the last date of the query window. 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 intraday prices (datetime, open, high, low, close, volume)

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_macro_indicator	<i>Retrieves a macroeconomic indicator of a country</i>
---------------------	---

Description

Queries <https://eodhd.com/financial-apis/macro-economics-data-and-macro-indicators-api>, which serves the World Bank series (gdp, inflation, unemployment, population and 40 others) by country.

Usage

```
get_macro_indicator(
  country = "USA",
  indicator = "gdp_current_usd",
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

country	a country code in ISO 3166-1 alpha 3 (e.g. "USA", "DEU", "BRA")
indicator	the code of the indicator (default "gdp_current_usd"). The full list is at the endpoint documentation.
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 the historical values of the indicator

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_gdp <- get_macro_indicator("USA", "gdp_current_usd")

## 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_news_word_weights` *Retrieves the most relevant words of the news of a ticker*

Description

Queries the news word weights endpoint of eodhd, which returns the keywords of the news of a ticker with their weight.

Usage

```
get_news_word_weights(
  ticker = "AAPL",
  exchange = "US",
  first_date = NULL,
  last_date = NULL,
  limit = 50,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

<code>ticker</code>	A company ticker (e.g. AAPL). You can find all tickers for a particular exchange with get_tickers() .
<code>exchange</code>	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"
<code>first_date</code>	an optional first date of the window
<code>last_date</code>	an optional last date of the window
<code>limit</code>	maximum number of words (default 50)
<code>cache_folder</code>	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 columns word and weight

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_words <- get_news_word_weights("AAPL", "US")

## End(Not run)
```

`get_options_contracts` *Retrieves option contracts*

Description

Queries the contracts of the Unicorn Bay options data <https://eodhd.com/marketplace/unicornbay/options>. Each row is one contract, with its strike, expiration, greeks and implied volatility.

Usage

```
get_options_contracts(
  underlying_symbol = "AAPL",
  contract = NULL,
  type = NULL,
  exp_date = NULL,
  first_exp_date = NULL,
  last_exp_date = NULL,
  first_strike = NULL,
  last_strike = NULL,
  sort = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

`underlying_symbol`
the underlying symbol, without the exchange (e.g. "AAPL"). This endpoint only covers US options, so a symbol given with its exchange ("AAPL.US") is trimmed for you.

contract	an optional single contract code (e.g. "AAPL270115C00200000"). When given, the underlying symbol is ignored by the api.
type	an optional type of contract ("call" or "put")
exp_date	a single expiration date, or NULL
first_exp_date	an optional first expiration date of the window
last_exp_date	an optional last expiration date of the window
first_strike	an optional lower bound of the strike
last_strike	an optional upper bound of the strike
sort	an optional sort expression (e.g. "-exp_date")
limit	maximum number of rows (default 100, maximum 1000)
offset	pagination offset (default 0)
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 the option contracts

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_contracts <- get_options_contracts("AAPL", type = "call")

## End(Not run)
```

get_options_eod	<i>Retrieves the end of day prices of option contracts</i>
-----------------	--

Description

Queries the eod endpoint of the Unicorn Bay options data <https://eodhd.com/marketplace/unicornbay/options>, with one row per contract and trading day.

Usage

```
get_options_eod(
  underlying_symbol = "AAPL",
  contract = NULL,
  type = NULL,
  date = NULL,
  first_date = NULL,
  last_date = NULL,
  exp_date = NULL,
  first_exp_date = NULL,
  last_exp_date = NULL,
  first_strike = NULL,
  last_strike = NULL,
  sort = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

<code>underlying_symbol</code>	the underlying symbol, without the exchange (e.g. "AAPL"). This endpoint only covers US options, so a symbol given with its exchange ("AAPL.US") is trimmed for you.
<code>contract</code>	an optional single contract code (e.g. "AAPL270115C00200000"). When given, the underlying symbol is ignored by the api.
<code>type</code>	an optional type of contract ("call" or "put")
<code>date</code>	a single trading date, or NULL
<code>first_date</code>	an optional first trading date of the window
<code>last_date</code>	an optional last trading date of the window
<code>exp_date</code>	a single expiration date, or NULL
<code>first_exp_date</code>	an optional first expiration date of the window
<code>last_exp_date</code>	an optional last expiration date of the window
<code>first_strike</code>	an optional lower bound of the strike
<code>last_strike</code>	an optional upper bound of the strike
<code>sort</code>	an optional sort expression (e.g. "-exp_date")
<code>limit</code>	maximum number of rows (default 100, maximum 1000)
<code>offset</code>	pagination offset (default 0)
<code>cache_folder</code>	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 the end of day prices of the contracts

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_eod <- get_options_eod("AAPL")

## End(Not run)
```

```
get_options_underlyings
```

Retrieves the symbols with options data

Description

Queries the Unicorn Bay options data of the eodhd marketplace <https://eodhd.com/marketplace/unicornbay/options>, and returns the underlying symbols that have contracts available.

Usage

```
get_options_underlyings(
  limit = 1000,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

limit maximum number of rows (default 1000)

offset pagination offset (default 0)

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 one column, underlying_symbol

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_underlyings <- get_options_underlyings()

## End(Not run)
```

get_policy_rates	<i>Retrieves the policy rates of the central banks</i>
------------------	--

Description

Queries the policy rates endpoint of eodhd, with the historical decisions of the central banks (fed funds target, ecb deposit rate, and so on).

Usage

```
get_policy_rates(
  code = NULL,
  country = NULL,
  central_bank = NULL,
  first_date = NULL,
  last_date = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

code	an optional code of the rate (e.g. "FED_FUNDS_TARGET_UPPER")
country	an optional country code (e.g. "US")
central_bank	an optional central bank code (e.g. "FED", "ECB")
first_date	an optional first date of the window
last_date	an optional last date of the window
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)

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 the policy rates

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_policy <- get_policy_rates(central_bank = "FED")

## End(Not run)
```

get_prices	<i>Retrieves adjusted and unadjusted stock prices</i>
------------	---

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_real_estate	<i>Retrieves the real estate series of a country</i>
-----------------	--

Description

Queries the real estate endpoints of eodhd for the headline indices of a country (house prices, rents and their real counterparts).

Usage

```
get_real_estate(
  code = "US",
  type = NULL,
  metric = NULL,
  first_date = NULL,
  last_date = NULL,
  sort = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

code	the country code (e.g. "US"). See <code>get_real_estate_countries()</code> .
type	an optional type of series
metric	an optional metric
first_date	an optional first date of the window
last_date	an optional last date of the window
sort	an optional sort expression (e.g. "code")
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the real estate series

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_re <- get_real_estate("US")

## End(Not run)
```

```
get_real_estate_countries
```

Retrieves the countries covered by the real estate data

Description

Queries the real estate endpoints of eodhd, which serve house price indices and rent series by country and region.

Usage

```
get_real_estate_countries(
  sort = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

sort	an optional sort expression (e.g. "code")
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the available countries

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_countries <- get_real_estate_countries()

## End(Not run)
```

```
get_real_estate_detailed
```

Retrieves the detailed real estate series of a country

Description

Queries the detailed real estate endpoint of eodhd, which breaks the series of a country by area, property type and vintage.

Usage

```
get_real_estate_detailed(
  code = "US",
  area = NULL,
  property_type = NULL,
  vintage = NULL,
  freq = NULL,
  first_date = NULL,
  last_date = NULL,
  sort = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

code	the country code (e.g. "US"). See get_real_estate_countries() .
area	an optional area (e.g. a metropolitan area)
property_type	an optional property type
vintage	an optional vintage of the series
freq	an optional frequency of the series
first_date	an optional first date of the window
last_date	an optional last date of the window
sort	an optional sort expression (e.g. "code")
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the detailed real estate series

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_re <- get_real_estate_detailed("US")

## End(Not run)
```

```
get_real_estate_series
```

Retrieves the catalog of detailed real estate series of a country

Description

Lists the series available at [get_real_estate_detailed\(\)](#) for a country, with their area, property type and vintage.

Usage

```
get_real_estate_series(
  code = "US",
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

code	the country code (e.g. "US"). See get_real_estate_countries() .
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 the available series

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_series <- get_real_estate_series("US")

## End(Not run)
```

get_real_time	<i>Retrieves delayed real time (live) quotes</i>
---------------	--

Description

Queries the live/delayed endpoint <https://eodhd.com/financial-apis/live-ohlcv-stocks-api> for one or more tickers. The delay depends on the exchange and on your subscription. Live data is never cached.

Usage

```
get_real_time(  
  ticker = "AAPL",  
  exchange = "US",  
  extra_tickers = NULL,  
  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"
extra_tickers	an optional character vector of additional tickers, with exchange included (e.g. c("TSLA.US", "VTI.US")). The api allows up to 15 extra tickers per call.
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 one row per ticker

Examples

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

get_reference_rates *Retrieves the overnight reference rates*

Description

Queries the reference rates endpoint of eodhd, with the benchmark overnight rates (SOFR, ESTR, SONIA and others).

Usage

```
get_reference_rates(
  code = NULL,
  currency = NULL,
  first_date = NULL,
  last_date = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

code	an optional code of the rate (e.g. "SOFR")
currency	an optional currency of the rate ("USD", "GBP" or "EUR")
first_date	an optional first date of the window
last_date	an optional last date of the window
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the reference rates

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_sofr <- get_reference_rates(code = "SOFR")

## End(Not run)
```

```
get_sanctions_entities
```

Retrieves sanctioned entities

Description

Queries the sanctions endpoints of eodhd, which consolidate the lists of the main sanctioning bodies (OFAC, EU, UK and others).

Usage

```
get_sanctions_entities(
  source = NULL,
  type = NULL,
  program = NULL,
  country = NULL,
  query = NULL,
  active = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

source	an optional source of the list (see get_sanctions_sources())
type	an optional type of entity (e.g. "individual", "entity")
program	an optional sanction program (see get_sanctions_programs())
country	an optional country code
query	an optional free text search
active	an optional flag (TRUE/FALSE) to keep only the active records
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the sanctioned entities

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_entities <- get_sanctions_entities(country = "RU", limit = 10)

## End(Not run)
```

`get_sanctions_programs`

Retrieves the sanction programs

Description

Lists the programs that can be used to filter `get_sanctions_entities()`.

Usage

```
get_sanctions_programs(cache_folder = get_default_cache(), check_quota = TRUE)
```

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 `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 the sanction programs

Examples

```
## Not run:  
set_token("YOUR_VALID_TOKEN")  
df_programs <- get_sanctions_programs()  
  
## End(Not run)
```

get_sanctions_sources *Retrieves the sources of the sanction lists*

Description

Lists the sanctioning bodies behind the consolidated data, with the date of their last update.

Usage

```
get_sanctions_sources(cache_folder = get_default_cache(), check_quota = TRUE)
```

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 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 the sources

Examples

```
## Not run:  
set_token("YOUR_VALID_TOKEN")  
df_sources <- get_sanctions_sources()  
  
## End(Not run)
```

get_sanctions_vessels *Retrieves sanctioned vessels*

Description

Queries the sanctions endpoints of eodhd for the vessels of the consolidated lists, searchable by IMO number and flag.

Usage

```
get_sanctions_vessels(
  source = NULL,
  imo = NULL,
  flag = NULL,
  vessel_type = NULL,
  query = NULL,
  program = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

source	an optional source of the list (see get_sanctions_sources())
imo	an optional IMO number of the vessel
flag	an optional flag (country) of the vessel
vessel_type	an optional type of vessel
query	an optional free text search
program	an optional sanction program (see get_sanctions_programs())
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the sanctioned vessels

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_vessels <- get_sanctions_vessels(limit = 10)

## End(Not run)
```

get_screener

*Screens stocks by fundamentals, price and technical signals***Description**

Queries the screener endpoint <https://eodhd.com/financial-apis/stock-market-screener-api>. Filters are given as a list of conditions, each one a list of field, operation and value – the function converts them to the json string the api expects.

Usage

```
get_screener(
  filters = NULL,
  signals = NULL,
  sort = NULL,
  limit = 50,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

filters	a list of conditions, e.g. <code>list(list("market_capitalization", ">", 1e12), list("exchange", "=", "us"))</code> . A character string with the json expected by the api is also accepted.
signals	an optional character vector of signals (e.g. "bookvalue_neg", "200d_new_lo")
sort	an optional sorting rule (e.g. "market_capitalization.desc")
limit	maximum number of rows (default 50, maximum 100)
offset	pagination offset (default 0)
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 the screened companies

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_screen <- get_screener(
  filters = list(list("market_capitalization", ">", 1e12))
)

## End(Not run)
```

get_search	<i>Searches for tickers, companies and isins</i>
------------	--

Description

Queries the search endpoint <https://eodhd.com/financial-apis/search-api-for-stocks-etfs-mutual-funds>, which accepts a company name, a ticker or an isin and returns the matching instruments.

Usage

```
get_search(
  query,
  limit = 15,
  type = NULL,
  exchange = NULL,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

query	the text to search for (e.g. "apple", "AAPL", "US0378331005")
limit	maximum number of results (default 15, maximum 500)
type	an optional instrument type ("stock", "etf", "fund", "bonds", "index", "commodity", "crypto")
exchange	an optional exchange code, to restrict the search

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 the matching instruments

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_search <- get_search("apple")

## End(Not run)
```

get_sentiments	<i>Retrieves the daily news sentiment of one or more tickers</i>
----------------	--

Description

Queries the sentiment endpoint <https://eodhd.com/financial-apis/stock-market-financial-news-api>, which aggregates, by day, the number of news articles of a ticker and their normalised sentiment (from -1 to 1).

Usage

```
get_sentiments(
  symbols = "AAPL.US",
  first_date = Sys.Date() - 30,
  last_date = Sys.Date(),
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

symbols	a character vector of tickers with exchange (e.g. c("AAPL.US", "BTC-USD.CC"))
first_date	the first date of the window. Default is 30 days ago.
last_date	the last date of the window. 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 columns symbol, date, count and normalized

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_sentiment <- get_sentiments("AAPL.US")

## End(Not run)
```

```
get_sovereign_cds_spreads
```

Retrieves the sovereign cds spreads

Description

Queries the credit risk endpoints of eodhd for the credit default swap spreads of the sovereign issuers.

Usage

```
get_sovereign_cds_spreads(
  country = NULL,
  as_of = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

country	an optional country code (e.g. "BR")
as_of	an optional reference date. Default (NULL) is the latest release.
limit	maximum number of rows (default 100)

offset	pagination offset (default 0)
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 the sovereign cds spreads

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_cds <- get_sovereign_cds_spreads()

## End(Not run)
```

```
get_sovereign_credit_ratings
```

Retrieves the sovereign credit ratings

Description

Queries the credit risk endpoints of eodhd for the ratings of the three major agencies by country.

Usage

```
get_sovereign_credit_ratings(
  country = NULL,
  as_of = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

country	an optional country code (e.g. "BR")
as_of	an optional reference date. Default (NULL) is the latest release.
limit	maximum number of rows (default 100)

offset	pagination offset (default 0)
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 the sovereign credit ratings

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_ratings <- get_sovereign_credit_ratings()

## End(Not run)
```

```
get_sovereign_risk_premium
```

Retrieves the sovereign risk premium

Description

Queries the credit risk endpoints of eodhd, which serve the country risk premium built from the sovereign rating and the default spread.

Usage

```
get_sovereign_risk_premium(
  country = NULL,
  region = NULL,
  as_of = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

country	an optional country code (e.g. "BR")
region	an optional region name (e.g. "Latin America")
as_of	an optional reference date. Default (NULL) is the latest release.
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the sovereign risk premium

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_premium <- get_sovereign_risk_premium(country = "BR")

## 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 <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 split information

Examples

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

## End(Not run)
```

get_splits_calendar	<i>Retrieves the splits calendar</i>
---------------------	--------------------------------------

Description

Queries <https://eodhd.com/financial-apis/calendar-upcoming-earnings-ipos-and-splits> and returns the stock splits of a period, including the upcoming ones.

Usage

```
get_splits_calendar(
  first_date = Sys.Date(),
  last_date = Sys.Date() + 30,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

first_date	the first date of the window. Default is today.
last_date	the last date of the window. Default is in one month.
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 the splits calendar

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_splits <- get_splits_calendar()

## End(Not run)
```

get_symbol_change_history

Retrieves the history of ticker changes

Description

Queries <https://eodhd.com/financial-apis/exchanges-api-list-of-tickers-and-trading-hours> (symbol change history) and returns the tickers that were renamed within a period. Useful for keeping a database of symbols in sync.

Usage

```
get_symbol_change_history(
  first_date = Sys.Date() - 30,
  last_date = Sys.Date(),
  exchange = NULL,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

first_date	the first date of the window. Default is 30 days ago.
last_date	the last date of the window. Default is today.
exchange	an optional exchange code, to restrict the output
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 old symbol, new symbol and effective date

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_changes <- get_symbol_change_history()

## End(Not run)
```

get_technical	<i>Retrieves technical indicators</i>
---------------	---------------------------------------

Description

Queries the technical endpoint <https://eodhd.com/financial-apis/technical-indicators-api>, which computes an indicator (moving averages, rsi, macd, bollinger bands, beta, ..) on the server side. See the api documentation for the full list of indicators and of the parameters each one accepts.

Usage

```
get_technical(
  ticker = "AAPL",
  exchange = "US",
  indicator = "sma",
  period = 50,
  first_date = Sys.Date() - 365,
  last_date = Sys.Date(),
  order = "a",
  ...,
```

```

    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"
indicator	the indicator to compute (api parameter "function"), e.g. "sma", "ema", "rsi", "macd", "bbands", "volatility", "stochastic"
period	the number of periods used by the indicator. Default is 50.
first_date	the first date of the output. Default is one year ago. Note that the api needs history before this date to compute the indicator, so ask for a window comfortably longer than the period.
last_date	the last date of the output. Default is today.
order	"d" for descending dates or "a" (default) for ascending
...	any other parameter accepted by the indicator (e.g. fast_period = 12)
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 the indicator values

Examples

```

## Not run:
set_token("YOUR_VALID_TOKEN")
df_sma <- get_technical("AAPL", "US", indicator = "sma", period = 50)

## 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 <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 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)
```

get_ticks	<i>Retrieves historical tick data</i>
-----------	---------------------------------------

Description

Queries the tick endpoint <https://eodhd.com/financial-apis/us-stock-market-tick-data-api>, which returns every trade of a symbol within a time window. The window is given in unix timestamps (seconds), as required by the api.

Usage

```
get_ticks(
  ticker = "AAPL",
  exchange = "US",
  first_time = Sys.time() - 3600,
  last_time = Sys.time(),
  limit = 1000,
  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_time	the start of the window, as a POSIXct or as a unix timestamp
last_time	the end of the window, as a POSIXct or as a unix timestamp
limit	maximum number of ticks (api default is 1, maximum 10000)
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 tick data

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_ticks <- get_ticks("AAPL", "US",
  first_time = Sys.time() - 3600,
  last_time = Sys.time())

## End(Not run)
```

get_user_info

Retrieves the details of the account behind the current token

Description

Returns the information of the user api (<https://eodhd.com/financial-apis/user-api>): name, subscription, number of api calls used today and the daily limit.

Usage

```
get_user_info()
```

Value

a dataframe with one row

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
get_user_info()

## End(Not run)
```

get_ust_rates

Retrieves the rates of the US Treasury

Description

Queries the US Treasury endpoints of eodhd: the daily par yield curve, the bill rates, the long term rates and the real yield curve.

Usage

```
get_ust_rates(
  type = "yield",
  first_date = NULL,
  last_date = NULL,
  year = NULL,
  limit = 100,
  offset = 0,
  cache_folder = get_default_cache(),
  check_quota = TRUE
)
```

Arguments

type	one of "yield" (default, the par yield curve), "bill", "long-term" or "real-yield"
first_date	an optional first date of the window
last_date	an optional last date of the window
year	an optional single year (e.g. 2026), as an alternative to the dates
limit	maximum number of rows (default 100)
offset	pagination offset (default 0)
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 the treasury rates

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_ust <- get_ust_rates("yield", first_date = "2026-01-01")

## End(Not run)
```

get_us_quote_delayed *Retrieves delayed quotes for US tickers*

Description

Queries the delayed us quote endpoint (<https://eodhd.com/financial-apis/live-v2-for-us-stocks-extended-quotes>) which returns a richer quote (bid/ask, exchange, company name) than `get_real_time()`, for US tickers only. Live data is never cached.

Usage

```
get_us_quote_delayed(tickers = "AAPL", fields = NULL, check_quota = TRUE)
```

Arguments

tickers	a character vector of US tickers (e.g. c("AAPL", "MSFT"))
fields	an optional character vector restricting the returned fields
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 one row per ticker

Examples

```
## Not run:
set_token("YOUR_VALID_TOKEN")
df_quotes <- get_us_quote_delayed(c("AAPL", "MSFT"))

## End(Not run)
```

parse_financials

Parses financial data from the API

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 get_demo_token() for using a demo token.
-------	---

Value

Nothing

Examples

```
## Not run:
set_token()

## End(Not run)
```

Index

`get_bulk_eod`, 3
`get_bulk_fundamentals`, 4
`get_cds_market_aggregates`, 5
`get_commodities`, 6
`get_corporate_cmdi`, 7
`get_corporate_hqm_yields`, 8
`get_default_spreads`, 10
`get_demo_token`, 11
`get_demo_token()`, 65
`get_dividends`, 11
`get_dividends_calendar`, 12
`get_earnings`, 13
`get_earnings_trends`, 14
`get_economic_events`, 15
`get_eodhd`, 16
`get_exchange_details`, 18
`get_exchanges`, 17
`get_fundamentals`, 19
`get_fundamentals()`, 5, 64
`get_funding_stress_spreads`, 20
`get_historical_market_cap`, 21
`get_id_mapping`, 23
`get_index_composition`, 24
`get_index_composition()`, 25
`get_index_list`, 25
`get_insider_transactions`, 25
`get_intraday`, 27
`get_ipos`, 28
`get_macro_indicator`, 29
`get_news`, 30
`get_news_word_weights`, 31
`get_options_contracts`, 32
`get_options_eod`, 33
`get_options_underlyings`, 35
`get_policy_rates`, 36
`get_prices`, 37
`get_real_estate`, 38
`get_real_estate_countries`, 39
`get_real_estate_countries()`, 39, 41, 42
`get_real_estate_detailed`, 40
`get_real_estate_detailed()`, 42
`get_real_estate_series`, 42
`get_real_time`, 43
`get_real_time()`, 63
`get_reference_rates`, 44
`get_sanctions_entities`, 45
`get_sanctions_entities()`, 46
`get_sanctions_programs`, 46
`get_sanctions_programs()`, 45, 48
`get_sanctions_sources`, 47
`get_sanctions_sources()`, 45, 48
`get_sanctions_vessels`, 48
`get_screener`, 49
`get_search`, 50
`get_sentiments`, 51
`get_sovereign_cds_spreads`, 52
`get_sovereign_credit_ratings`, 53
`get_sovereign_risk_premium`, 54
`get_splits`, 55
`get_splits_calendar`, 56
`get_symbol_change_history`, 57
`get_technical`, 58
`get_tickers`, 60
`get_tickers()`, 3, 12, 19, 20, 22, 27, 30, 31, 37, 43, 56, 59–61
`get_ticks`, 60
`get_us_quote_delayed`, 63
`get_user_info`, 62
`get_ust_rates`, 62

`parse_financials`, 64
`parse_financials()`, 19

`set_token`, 65