

Package: screen (via r-universe)

April 2, 2025

Type Package

Title Yahoo Finance Screener

Version 0.1

Author Jason Foster

Maintainer Jason Foster <jason.j.foster@gmail.com>

Description Fast and efficient access to Yahoo Finance's screener functionality for querying and retrieval of financial data.

License GPL (>= 2)

Encoding UTF-8

URL <https://github.com/jasonjfoster/screen/r>

BugReports <https://github.com/jasonjfoster/screen/r/issues>

Imports curl, jsonlite

Suggests covr, testthat

LazyData true

Roxygen list(old_usage = TRUE)

RoxygenNote 7.3.2

Config/pak/sysreqs libssl-dev

Repository <https://jasonjfoster.r-universe.dev>

RemoteUrl <https://github.com/jasonjfoster/screen>

RemoteRef HEAD

RemoteSha b0f9d5ec3f5ed12ae94f76d3c87a279d20d4ad9c

RemoteSubdir r

Contents

create_payload	2
create_query	3
data_categoryname	4

data_errors	4
data_exchange	4
data_filters	5
data_fundfamilyname	5
data_industry	5
data_peer_group	6
data_region	6
data_sector	6
get_data	7
get_session	7

Index	9
--------------	----------

create_payload	<i>Create a Payload for the Yahoo Finance API</i>
----------------	---

Description

A function to create a payload to query the Yahoo Finance API with customizable parameters.

Usage

```
create_payload(sec_type = "equity", query = NULL, size = 25,
              offset = 0, sort_field = NULL, sort_type = NULL,
              top_operator = "and")
```

Arguments

sec_type	string. Type of security to search (i.e., "equity", "mutualfund", "etf", "index", "future").
query	list. Structured query to filter results created by the create_query function.
size	integer. Number of results to return.
offset	integer. Starting position of the results.
sort_field	string. Field to sort the results.
sort_type	string. Type of sort to apply (i.e., "asc", "desc").
top_operator	string. Logical operator for the top-level of the query (i.e., "and", "or")

Value

A list representing the payload to be sent to the Yahoo Finance API with the specified parameters.

Examples

```

filters <- list(
  list("eq", list("region", "us")),
  list("btwn", list("intradaymarketcap", 2000000000, 10000000000)),
  list("btwn", list("intradaymarketcap", 10000000000, 100000000000)),
  list("gt", list("intradaymarketcap", 100000000000)),
  list("gt", list("dayvolume", 5000000))
)

query <- create_query(filters)

payload <- create_payload("equity", query)

```

create_query

Create a Structured Query for the Yahoo Finance API

Description

A function to create a structured query with logical operations and nested conditions formatted for the Yahoo Finance API.

Usage

```

create_query(filters = list("eq", list("region", "us")),
  top_operator = "and")

```

Arguments

filters	list. Each element is a sublist that defines a filtering condition with the following structure:
	comparison string. Comparison operator (i.e., "gt", "lt", "eq", "btwn").
	field list. Field name (e.g. "region") and its associated value(s).
top_operator	string. Top-level logical operator to combine all filters (i.e., "and", "or").

Value

A nested list representing the structured query with logical operations and nested conditions formatted for the Yahoo Finance API.

Examples

```

filters <- list(
  list("eq", list("region", "us")),
  list("btwn", list("intradaymarketcap", 2000000000, 10000000000)),
  list("btwn", list("intradaymarketcap", 10000000000, 100000000000)),
  list("gt", list("intradaymarketcap", 100000000000)),
  list("gt", list("dayvolume", 5000000))
)

```

```
query <- create_query(filters)
```

data_categoryname	<i>Category Name Data for the Yahoo Finance API</i>
-------------------	---

Description

A data frame with the available category name data for the Yahoo Finance API.

Usage

```
data_categoryname
```

Format

A data frame.

data_errors	<i>Errors Data for the Yahoo Finance API</i>
-------------	--

Description

A data frame with the available errors data for the Yahoo Finance API.

Usage

```
data_errors
```

Format

A data frame.

data_exchange	<i>Exchange Data for the Yahoo Finance API</i>
---------------	--

Description

A data frame with the available exchange data for the Yahoo Finance API.

Usage

```
data_exchange
```

Format

A data frame.

data_filters	<i>Filters Data for the Yahoo Finance API</i>
--------------	---

Description

A data frame with the available filters data for the Yahoo Finance API.

Usage

```
data_filters
```

Format

A data frame.

data_fundfamilyname	<i>Fund Family Name Data for the Yahoo Finance API</i>
---------------------	--

Description

A data frame with the available fund family name data for the Yahoo Finance API.

Usage

```
data_fundfamilyname
```

Format

A data frame.

data_industry	<i>Industry Data for the Yahoo Finance API</i>
---------------	--

Description

A data frame with the available industry name for the Yahoo Finance API.

Usage

```
data_industry
```

Format

A data frame.

data_peer_group	<i>Peer Group Data for the Yahoo Finance API</i>
-----------------	--

Description

A data frame with the available peer group data for the Yahoo Finance API.

Usage

```
data_peer_group
```

Format

A data frame.

data_region	<i>Regions Data for the Yahoo Finance API</i>
-------------	---

Description

A data frame with the available regions for the Yahoo Finance API.

Usage

```
data_region
```

Format

A data frame.

data_sector	<i>Sector Data for the Yahoo Finance API</i>
-------------	--

Description

A data frame with the available sector data for the Yahoo Finance API.

Usage

```
data_sector
```

Format

A data frame.

`get_data`*Get Data from the Yahoo Finance API*

Description

A function to get data from the Yahoo Finance API using the specified payload.

Usage

```
get_data(payload = NULL)
```

Arguments

`payload` list. Payload that contains search criteria created using the [create_query](#) and [create_payload](#) functions.

Value

A data frame that contains data from the Yahoo Finance API for the specified search criteria.

Examples

```
filters <- list(
  list("eq", list("region", "us")),
  list("btwn", list("intradaymarketcap", 2000000000, 10000000000)),
  list("btwn", list("intradaymarketcap", 10000000000, 100000000000)),
  list("gt", list("intradaymarketcap", 100000000000)),
  list("gt", list("dayvolume", 5000000))
)

query <- create_query(filters)

payload <- create_payload("equity", query)

data <- get_data(payload)
```

`get_session`*Get the Crumb, Cookies, and Handle for Yahoo Finance API*

Description

A function to get the crumb, cookies, and handle required to authenticate and interact with the Yahoo Finance API.

Usage

```
get_session()
```

Value

A list containing the following elements:

handle	A curl handle object for subsequent requests.
crumb	A string representing the crumb value for authentication.
cookies	A data frame of cookies for the request.

Examples

```
session <- get_session()
```


Index

* datasets

- data_categoryname, 4
- data_errors, 4
- data_exchange, 4
- data_filters, 5
- data_fundfamilyname, 5
- data_industry, 5
- data_peer_group, 6
- data_region, 6
- data_sector, 6

- create_payload, 2, 7
- create_query, 2, 3, 7

- data_categoryname, 4
- data_errors, 4
- data_exchange, 4
- data_filters, 5
- data_fundfamilyname, 5
- data_industry, 5
- data_peer_group, 6
- data_region, 6
- data_sector, 6

- get_data, 7
- get_session, 7