

Package ‘SNBdata’

May 19, 2024

Type Package

Title Download Data from the Swiss National Bank (SNB)

Version 0.2.1

Date 2023-12-12

Maintainer Enrico Schumann <es@enricoschumann.net>

Description Download data (tables and datasets) from the Swiss National Bank (SNB; <<https://www.snb.ch/en>>), the Swiss central bank. The package is lightweight and comes with few dependencies; suggested packages are used only if data is to be transformed into particular data structures, for instance into 'zoo' objects. Downloaded data can optionally be cached, to avoid repeated downloads of the same files.

Suggests jsonlite, zoo

License GPL-3

URL <http://enricoschumann.net/R/packages/SNBdata/> ,
<https://git.sr.ht/~enricoschumann/SNBdata>

LazyLoad yes

NeedsCompilation no

Author Enrico Schumann [aut, cre] (<<https://orcid.org/0000-0001-7601-6576>>)

Contents

SNBdata	2
Index	5

Description

Download data ('tables' and 'datasets') from the Swiss National Bank (SNB) and convert to data-frames.

Usage

```
fetch_data(id,
           type = "table", dest.dir = NULL,
           return.class = NULL, verbose = TRUE,
           language = "en",
           name.sep = " :: ",
           method,
           na.drop = TRUE,
           time.series = FALSE, ...)

fetch_last_update(id,
                  type = "table", dest.dir = NULL,
                  verbose = TRUE, language = "en", ...)

fetch_info(id,
           type = "table", dest.dir = NULL,
           verbose = TRUE,
           language = "en",
           name.sep = " :: ",
           method, ...)
```

Arguments

<code>dest.dir</code>	file path: where to store the downloaded files? See Details.
<code>id</code>	string: identifier of the table/dataset
<code>type</code>	string: 'table' or 'dataset'
<code>method</code>	see download.file
<code>verbose</code>	logical: print messages, e.g. about download progress?
<code>return.class</code>	NULL or character: not yet implemented (but in future versions, zoo will be supported)
<code>language</code>	string: en, fr or de
<code>...</code>	passed on to read.table
<code>name.sep</code>	string used when pasting description hierarchies
<code>time.series</code>	logical. If TRUE, data are transformed into time-series.
<code>na.drop</code>	logical. If TRUE, rows with no finite values at all are dropped. Only used if <code>time.series</code> is TRUE.

Details

The Swiss National Bank (SNB) provides data as either so-called tables or datasets. `fetch_table` can handle both, but the `type` argument must be specified.

`fetch_table` downloads data, which typically are in CSV format, from the SNB's website and stores them, with a date prefix, in directory `dest.dir`. If the latter is `NULL`, a temporary directory is used (through `tempdir`); but much better is to use a more-persistent storage location. If a file with today's date exists in `dest.dir`, that file is read, and nothing is downloaded.

For downloading, function `download.file` is used. See `download.file` for options; in particular, see the hints about timeout when downloading large files.

When argument `time.series` is `TRUE`, `fetch_table` will rearrange the data into time-series. This requires columns named "Date" and "Value" in the particular dataset/table. If the "Date" column has a YYYY-MM-DD pattern, it will be transformed with `as.Date`.

Value

typically a `data.frame`, potentially with attributes additional attributes:

`info` information about identifiers used by the SNB

`columns` if `time.series` is `TRUE`, the names of the columns used for grouping the rows

Should the download fail, the function returns `NULL` invisibly.

Author(s)

Enrico Schumann

References

<https://data.snb.ch/en>

for help on the API, see https://data.snb.ch/en/help#data_api

Examples

```
## (Internet connection required)

### set directory for storing the files. This is
### only an example: Much better is to use a permanent
### storage-location, such as '~/Downloads/SNBdata'
data.dir <- tempdir()

rates <- fetch_data("rendoblim",
                    type = "table",
                    dest.dir = data.dir,
                    language = "en")

### ==> have data transformed into time-series
rates <- fetch_data("rendoblim",
                    type = "table",
                    dest.dir = data.dir,
```

```
        language = "en",
        time.series = TRUE)

if (!is.null(rates)) ## check: if download failed, results
                    ## are NULL
    attr(rates, "info")

stock.markets <- fetch_data("capchstocki",
                           type = "table",
                           dest.dir = data.dir,
                           time.series = TRUE)
## e.g.: stock.markets[, "GDR"] ## total return index
```

Index

* **SNB**

SNBdata, [2](#)

as.Date, [3](#)

data.frame, [3](#)

download.file, [2](#), [3](#)

fetch_data (SNBdata), [2](#)

fetch_info (SNBdata), [2](#)

fetch_last_update (SNBdata), [2](#)

NULL, [3](#)

read.table, [2](#)

SNBdata, [2](#)

SNBdata-package (SNBdata), [2](#)

tempdir, [3](#)

zoo, [2](#)