Package 'bbk'

October 21, 2026

Title Client for Central Bank APIs

Version 0.14.0

Description A client for retrieving data and metadata from central bank APIs including 'Banco Central do Brasil' (BCB), 'Banco de España' (BdE), 'Banco de México' (Banxico), 'Banco de Portugal' (BdP), 'Bank for International Settlements' (BIS), 'Bank of Canada' (BoC), 'Bank of England' (BoE), 'Bank of Israel' (BoI), 'Bank of Japan' (BoJ), 'Banque de France' (BdF), 'Czech National Bank' (CNB), 'Deutsche Bundesbank' (BBk), 'European Central Bank' (ECB), 'National Bank of Belgium' (NBB), 'National Bank of Poland' (NBP), 'Norges Bank' (NoB), 'Oesterreichische Nationalbank' (OeNB), 'Sveriges Riksbank' (SRb), and 'Swiss National Bank' (SNB).

License MIT + file LICENSE

URL https://m-muecke.github.io/bbk/, https://github.com/m-muecke/bbk

BugReports https://github.com/m-muecke/bbk/issues

Depends R (>= 4.1.0)

Imports checkmate (>= 1.9.0), curl, data.table (>= 1.17.0), httr2 (>= 1.0.6), jsonlite, stats, tools, utils, xml2

Suggests ggplot2, knitr, scales, testthat (>= 3.3.0), withr

Config/Needs/website knitr, rmarkdown

Config/roxygen2/markdown TRUE

Config/roxygen2/version 8.1.1

Config/testthat/edition 3

Config/testthat/parallel true

Encoding UTF-8

NeedsCompilation no

Author Maximilian Mücke [aut, cre] (ORCID: <https://orcid.org/0009-0000-9432-9795>)

Maintainer Maximilian Mücke <muecke.maximilian@gmail.com>

Repository CRAN

Date/Publication 2026-10-21 13:20:29 UTC

Contents

banxico_data . . . 3
banxico_metadata . . . 5
bbk_data . . . 6
bbk_dimension . . . 8
bbk_metadata . . . 9
bbk_series . . . 10
bcb_currencies . . . 11
bcb_data . . . 12
bcb_expectations . . . 13
bcb_fx_rates . . . 14
bcb_inflation . . . 15
bcb_selic . . . 17
bcb_top5 . . . 18
bde_data . . . 19
bde_latest . . . 20
bdf_codelist . . . 21
bdf_data . . . 22
bdf_dataset . . . 23
bdf_dimension . . . 24
bdp_data . . . 25
bdp_dataset . . . 27
bdp_dimension . . . 28
bdp_domain . . . 29
bdp_series . . . 30
bis_data . . . 31
bis_dimension . . . 33
bis_metadata . . . 34
boc_catalog . . . 35
boc_data . . . 35
boc_fx_rates . . . 37
boc_metadata . . . 38
boe_data . . . 39
boi_data . . . 40
boi_dimension . . . 41
boi_metadata . . . 42
boj_data . . . 43
boj_metadata . . . 44
cache . . . 45
cnb_czeonia . . . 46
cnb_data . . . 47
cnb_dimension . . . 49
cnb_fx_other_rates . . . 50
cnb_fx_rates . . . 51
cnb_indicators . . . 52
cnb_pribor . . . 54
cnb_snapshots . . . 55

cnb_tree 56
ecb_data 57
ecb_dimension 59
ecb_fx_rates 60
ecb_metadata 61
nbb_data 62
nbb_dimension 63
nbb_metadata 64
nbp_fx_rates 65
nbp_gold 66
nob_data 67
nob_dimension 69
nob_metadata 70
onb_data 71
onb_dimension 72
onb_frequency 73
onb_hierarchy 74
onb_metadata 75
onb_toc 76
snb_data 77
snb_dimension 78
snb_metadata 79
snb_toc 80
srb_calendar 81
srb_cross_rates 82
srb_data 83
srb_series 84

Index **86**

`banxico_data` *Fetch Banco de México (Banxico) data*

Description

Retrieve time series data from the Banco de México Sistema de Información Económica (SIE) API.

Usage

```
banxico_data(
  series,
  start_date = NULL,
  end_date = NULL,
  api_key = banxico_key()
)
```

Arguments

series	(character()) Up to 20 SIE series codes to query (e.g., "SF43718" for the FIX peso/US dollar exchange rate). Series codes can be found on the SIE website.
start_date	(NULL | Date(1) | character(1)) Start date of the data (e.g., "2024-01-01"). Must be paired with end_date. If both are NULL, the full history is returned. Default NULL.
end_date	(NULL | Date(1) | character(1)) End date of the data, in the same format as start_date. Must be paired with start_date. If both are NULL, the full history is returned. Default NULL.
api_key	(character(1)) The SIE API token. Defaults to the BANXICO_KEY environment variable.

Details

The SIE API requires a free access token. Request one at https://www.banxico.org.mx/SieAPIRest/service/v1/token and supply it via the api_key argument or the BANXICO_KEY environment variable.

Supply both start_date and end_date to restrict the period, or neither to return the complete history of each series. Supplying only one is an error.

Value

A data.table::data.table() with columns date, key, and value.

Source

https://www.banxico.org.mx/SieAPIRest/service/v1/

See Also

Other data: bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# fetch the FIX peso/US dollar exchange rate
banxico_data("SF43718", start_date = "2024-01-01", end_date = "2024-01-31")

# fetch multiple series at once
banxico_data(c("SF43718", "SF60653"), start_date = "2024-01-01", end_date = "2024-01-31")
```

banxico_metadata *Fetch Banco de México (Banxico) metadata*

Description

Retrieve series metadata from the Banco de México Sistema de Información Económica (SIE) API.

Usage

```
banxico_metadata(series, api_key = banxico_key())
```

Arguments

series	(character()) Up to 20 SIE series codes to query (e.g., "SF43718" for the FIX peso/US dollar exchange rate). Series codes can be found on the SIE website.
api_key	(character(1)) The SIE API token. Defaults to the BANXICO_KEY environment variable.

Value

A data.table::data.table() with columns id, title, freq, unit, start, and end.

Source

https://www.banxico.org.mx/SieAPIRest/service/v1/

See Also

Other metadata: bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
banxico_metadata("SF43718")
```

bbk_data *Fetch Deutsche Bundesbank (BBk) data*

Description

Retrieve time series data from the Bundesbank SDMX Web Service.

Usage

```
bbk_data(
  flow,
  key = NULL,
  start_period = NULL,
  end_period = NULL,
  first_n = NULL,
  last_n = NULL,
  updated_after = NULL
)
```

Arguments

flow	(character(1)) The flow to query, 5-8 characters. See bbk_metadata() for available dataflows.
key	(NULL | character()) The series keys to query.
start_period	(NULL | character(1) | integer(1)) The start date of the data. Supported formats: • YYYY for annual data (e.g., 2019) • YYYY-S[1-2] for semi-annual data (e.g., "2019-S1") • YYYY-Q[1-4] for quarterly data (e.g., "2019-Q1") • YYYY-MM for monthly data (e.g., "2019-01") • YYYY-W[01-53] for weekly data (e.g., "2019-W01") • YYYY-MM-DD for daily and business data (e.g., "2019-01-01") If NULL, no start date restriction is applied (data retrieved from the earliest available date). Default NULL.
end_period	(NULL | character(1) | integer(1)) The end date of the data, in the same format as start_period. If NULL, no end date restriction is applied (data retrieved up to the most recent available date). Default NULL.
first_n	(NULL | numeric(1)) Number of observations to retrieve from the start of the series. If NULL, no restriction is applied. Default NULL.
last_n	(NULL | numeric(1)) Number of observations to retrieve from the end of the series. If NULL, no restriction is applied. Default NULL.

updated_after (NULL | character(1) | Date(1) | POSIXct(1))
Retrieve only observations updated after the given timestamp (e.g., "2024-06-01T00:00:00"). Useful for incremental retrieval. If NULL, no restriction is applied. Default NULL.

Value

A data.table::data.table() with the requested data.

Source

https://api.statistiken.bundesbank.de/

See Also

Other data: banxico_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# fetch all data for a given flow and key
data = bbk_data("BBSIS", "D.I.ZAR.ZI.EUR.S1311.B.A604.R10XX.R.A.A._Z._Z.A")
head(data)

# fetch data for multiple keys
data = bbk_data("BBEX3", c("M.ISK.EUR", "USD.CA.AC.A01"))
head(data)

# specified period (start date-end date) for daily data
data = bbk_data(
  "BBSIS", "D.I.ZAR.ZI.EUR.S1311.B.A604.R10XX.R.A.A._Z._Z.A",
  start_period = "2020-01-01",
  end_period = "2020-08-01"
)
head(data)

# or only specify the start date
data = bbk_data(
  "BBSIS", "D.I.ZAR.ZI.EUR.S1311.B.A604.R10XX.R.A.A._Z._Z.A",
  start_period = "2024-04-01"
)
head(data)
```

bbk_dimension *Fetch Deutsche Bundesbank (BBk) dimensions*

Description

Retrieve the dimension structure for a given dataflow from the Bundesbank SDMX Web Service.

Usage

```
bbk_dimension(id)
```

Arguments

id	(character(1)) The id of the data structure definition to query (e.g., "BBK_BBSIS").

Value

A data.table::data.table() with columns:

id	The dimension id (e.g., "BBK_STD_FREQ", "BBK_STD_AREA")
position	The position of the dimension in the series key
codelist	The id of the associated codelist

Source

https://api.statistiken.bundesbank.de/

See Also

Other metadata: banxico_metadata(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
bbk_dimension("BBK_ERX")
```

bbk_metadata *Fetch Deutsche Bundesbank (BBk) metadata*

Description

Retrieve metadata from the Bundesbank time series database via the SDMX Web Service.

Usage

```
bbk_metadata(type, id = NULL, lang = "en")
```

Arguments

type	(character(1)) The type of metadata to query. One of: "datastructure", "dataflow", "codelist", or "concept".
id	(NULL | character(1)) The id to query. Default NULL.
lang	(character(1)) Language to query, either "en" or "de". Default "en".

Value

A data.table::data.table() with the requested metadata. The columns are:

id	The id of the metadata
name	The name of the metadata

Source

https://api.statistiken.bundesbank.de/

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
bbk_metadata("datastructure")
bbk_metadata("dataflow", "BBSIS")
bbk_metadata("codelist", "CL_BBK_ACIP_ASSET_LIABILITY")
bbk_metadata("concept", "CS_BBK_BSPL")
```

`bbk_series` *Fetch the Deutsche Bundesbank (BBk) series*

Description

Retrieve a single series by its key via the Bundesbank SDMX Web Service.

Usage

```
bbk_series(key)
```

Arguments

`key`	(`NULL` | `character()`) The series keys to query.

Value

A `data.table::data.table()` with the requested data.

Source

https://api.statistiken.bundesbank.de/

See Also

`bbk_data()` for an endpoint with more options.

Other data: `banxico_data()`, `bbk_data()`, `bcb_data()`, `bcb_expectations()`, `bcb_fx_rates()`, `bcb_inflation()`, `bcb_selic()`, `bcb_top5()`, `bde_data()`, `bde_latest()`, `bdf_codelist()`, `bdf_data()`, `bdf_dataset()`, `bdp_data()`, `bis_data()`, `boc_data()`, `boe_data()`, `boi_data()`, `boj_data()`, `cnb_czeonia()`, `cnb_data()`, `cnb_fx_other_rates()`, `cnb_fx_rates()`, `cnb_pribor()`, `ecb_data()`, `nbb_data()`, `nbp_fx_rates()`, `nbp_gold()`, `nob_data()`, `onb_data()`, `snb_data()`, `srb_cross_rates()`, `srb_data()`

Examples

```
bbk_series("BBEX3.M.DKK.EUR.BB.AC.A01")
bbk_series("BBAF3.Q.F41.S121.DE.S1.W0.LE.N._X.B")
bbk_series("BBBK11.D.TTA000")
```

---

bcb_currencies *Fetch Banco Central do Brasil (BCB) currencies*

Description

Retrieve the list of currencies available from the Banco Central do Brasil PTAX API.

Usage

```
bcb_currencies()
```

Value

A `data.table::data.table()` with columns `code`, `name`, and `type`.

Source

https://dadosabertos.bcb.gov.br/

See Also

Other metadata: `banxico_metadata()`, `bbk_dimension()`, `bbk_metadata()`, `bdf_dimension()`, `bdp_dataset()`, `bdp_dimension()`, `bdp_domain()`, `bdp_series()`, `bis_dimension()`, `bis_metadata()`, `boi_dimension()`, `boi_metadata()`, `boj_metadata()`, `cnb_dimension()`, `cnb_indicators()`, `cnb_snapshots()`, `cnb_tree()`, `ecb_dimension()`, `ecb_metadata()`, `nbb_dimension()`, `nbb_metadata()`, `nob_dimension()`, `nob_metadata()`, `onb_dimension()`, `onb_frequency()`, `onb_hierarchy()`, `onb_metadata()`, `onb_toc()`, `snb_dimension()`, `snb_metadata()`, `snb_toc()`, `srb_calendar()`, `srb_series()`

Examples

```
bcb_currencies()
```

bcb_data *Fetch Banco Central do Brasil (BCB) data*

Description

Retrieve time series data from the Banco Central do Brasil SGS (Sistema Gerenciador de Séries Temporais) API.

Usage

```
bcb_data(series, start_date = NULL, end_date = NULL)
```

Arguments

series	(integer(1)) The SGS series code to query (e.g., 1 for the USD/BRL exchange rate). Series codes can be found on the SGS website.
start_date	(NULL | Date(1) | character(1)) Start date of the data (e.g., "2024-01-01"). If NULL, all available data is returned. Default NULL.
end_date	(NULL | Date(1) | character(1)) End date of the data, in the same format as start_date. If NULL, data up to the latest available date is returned. Default NULL.

Details

Daily series require a date range and the API limits the query window to at most 10 years; supply start_date (and optionally end_date) when querying such series.

Value

A data.table::data.table() with the requested data.

Source

https://dadosabertos.bcb.gov.br/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# fetch USD/BRL exchange rate
bcb_data(1, start_date = "2024-01-01", end_date = "2024-01-31")

# fetch the Selic target rate
bcb_data(432, start_date = "2024-01-01", end_date = "2024-01-31")
```

---

bcb_expectations *Fetch Banco Central do Brasil (BCB) market expectations*

Description

Retrieve market expectations from the Banco Central do Brasil Focus survey (Relatório Focus) via the Olinda API. Each row is the summary of survey responses collected on a given date for a future reference period.

Usage

```
bcb_expectations(
  type = "annual",
  indicator = NULL,
  start_date = NULL,
  end_date = NULL
)
```

Arguments

type	(character(1)) The forecast horizon. One of "annual", "monthly", or "quarterly". Default "annual".
indicator	(NULL | character(1)) The economic indicator to filter on (e.g. "IPCA", "Selic", "Câmbio"). If NULL, all indicators are returned. Default NULL.
start_date	(NULL | Date(1) | character(1)) Start of the survey date range (e.g., "2024-01-01"). If NULL, no lower bound is applied. Default NULL.
end_date	(NULL | Date(1) | character(1)) End of the survey date range, in the same format as start_date. If NULL, no upper bound is applied. Default NULL.

Details

Querying without a filter returns the full history, which is large; supplying indicator and/or a date range is recommended.

Value

A data.table::data.table() with columns date, indicator, detail, reference, mean, median, sd, min, max, respondents, and base.

Source

https://dadosabertos.bcb.gov.br/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# annual IPCA inflation expectations
bcb_expectations("annual", "IPCA", start_date = "2024-01-01", end_date = "2024-01-31")
```

---

bcb_fx_rates *Fetch Banco Central do Brasil (BCB) exchange rates*

Description

Retrieve PTAX foreign exchange reference rates from the Banco Central do Brasil Olinda API. The rates are the closing (Fechamento) bid and ask quotations expressed in Brazilian real (BRL) per unit of the foreign currency.

Usage

```
bcb_fx_rates(currency, start_date, end_date = NULL)
```

Arguments

currency	(character()) One or more ISO 4217 currency codes (e.g. "USD", c("USD", "EUR")). See bcb_currencies() for the available currencies.
start_date	(Date(1) | character(1)) Start date of the data (e.g., "2024-01-01").
end_date	(NULL | Date(1) | character(1)) End date of the data, in the same format as start_date. If NULL, the start_date is used. Default NULL.

Details

Rates are published only on business days, so weekends and holidays return no rows.

Value

A data.table::data.table() with columns date, currency, bid, and ask.

Source

https://dadosabertos.bcb.gov.br/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# fetch USD/BRL closing rates
bcb_fx_rates("USD", start_date = "2024-01-01", end_date = "2024-01-31")

# fetch multiple currencies
bcb_fx_rates(c("USD", "EUR"), start_date = "2024-01-01", end_date = "2024-01-31")
```

---

bcb_inflation *Fetch Banco Central do Brasil (BCB) inflation expectations*

Description

Retrieve the rolling 12- or 24-month inflation expectations from the Banco Central do Brasil Focus survey (Relatório Focus) via the Olinda API.

Usage

```
bcb_inflation(
  horizon = "12m",
  indicator = NULL,
  start_date = NULL,
  end_date = NULL
)
```

Arguments

horizon	(character(1)) The forecast horizon, either "12m" (next 12 months) or "24m" (next 24 months). Default "12m".
indicator	(NULL | character(1)) The economic indicator to filter on (e.g. "IPCA", "Selic", "Câmbio"). If NULL, all indicators are returned. Default NULL.
start_date	(NULL | Date(1) | character(1)) Start of the survey date range (e.g., "2024-01-01"). If NULL, no lower bound is applied. Default NULL.
end_date	(NULL | Date(1) | character(1)) End of the survey date range, in the same format as start_date. If NULL, no upper bound is applied. Default NULL.

Value

A data.table::data.table() with columns date, indicator, smoothed, mean, median, sd, min, max, respondents, and base. The smoothed column indicates whether the forecast is the smoothed (suavizada) series.

Source

https://dadosabertos.bcb.gov.br/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# next-12-months IPCA inflation expectations
bcb_inflation("12m", "IPCA", start_date = "2024-01-01", end_date = "2024-01-31")
```

bcb_selic *Fetch Banco Central do Brasil (BCB) Selic expectations*

Description

Retrieve market expectations for the Selic target rate from the Banco Central do Brasil Focus survey (Relatório Focus) via the Olinda API. Each row summarises the forecasts collected on a given survey date for a future COPOM meeting.

Usage

```
bcb_selic(start_date = NULL, end_date = NULL)
```

Arguments

start_date	(NULL | Date(1) | character(1)) Start of the survey date range (e.g., "2024-01-01"). If NULL, no lower bound is applied. Default NULL.
end_date	(NULL | Date(1) | character(1)) End of the survey date range, in the same format as start_date. If NULL, no upper bound is applied. Default NULL.

Value

A data.table::data.table() with columns date, meeting, mean, median, sd, min, max, respondents, and base. The meeting column identifies the COPOM meeting the forecast refers to (e.g. "R3/2028").

Source

https://dadosabertos.bcb.gov.br/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
bcb_selic(start_date = "2024-01-01", end_date = "2024-01-31")
```

bcb_top5 *Fetch Banco Central do Brasil (BCB) Top-5 market expectations*

Description

Retrieve the Top-5 market expectations from the Banco Central do Brasil Focus survey (Relatório Focus) via the Olinda API. The Top-5 ranking summarises the forecasts of the institutions with the most accurate projections for a given indicator.

Usage

```
bcb_top5(type = "annual", indicator = NULL, start_date = NULL, end_date = NULL)
```

Arguments

type	(character(1)) The forecast horizon. One of "annual", "monthly", "quarterly", or "selic". Default "annual".
indicator	(NULL | character(1)) The economic indicator to filter on (e.g. "IPCA", "Selic", "Câmbio"). If NULL, all indicators are returned. Default NULL.
start_date	(NULL | Date(1) | character(1)) Start of the survey date range (e.g., "2024-01-01"). If NULL, no lower bound is applied. Default NULL.
end_date	(NULL | Date(1) | character(1)) End of the survey date range, in the same format as start_date. If NULL, no upper bound is applied. Default NULL.

Value

A data.table::data.table() with columns date, indicator, type_calc, reference, mean, median, sd, min, and max. The type_calc column holds the ranking horizon ("C" short, "M" medium, or "L" long term). For type = "selic" the reference column holds the COPOM meeting (e.g. "R4/2026").

Source

https://dadosabertos.bcb.gov.br/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# annual Top-5 IPCA inflation expectations
bcb_top5("annual", "IPCA", start_date = "2024-01-01", end_date = "2024-01-31")

# Top-5 Selic target rate expectations
bcb_top5("selic", start_date = "2024-01-01", end_date = "2024-01-31")
```

`bde_data` *Fetch Banco de España (BdE) data*

Description

Retrieve time series data from the BdE statistics API.

Usage

```
bde_data(key, time_range = NULL, lang = "en")
```

Arguments

`key`	(`character()`) The series keys to query.
`time_range`	(`NULL` | `character(1)` | `integer(1)`) The time range for the data. Can be an annual range (e.g., `2024`) or a frequency-based code: • Daily frequency (D): `"3M"` (last 3 months), `"12M"`, `"36M"` • Monthly frequency (M): `"30M"`, `"60M"`, `"MAX"` (entire series) • Quarterly frequency (Q): `"30M"`, `"60M"`, `"MAX"` • Annual frequency (A): `"60M"`, `"MAX"` If `NULL` (default), returns the smallest range for the series frequency (e.g., `"30M"` for monthly series).
`lang`	(`character(1)`) Language to query, either `"en"` or `"es"`.

Details

You can search for the series codes in the BIEST application or in the tables published by the Banco de España.

Value

A `data.table::data.table()` with the requested data.

Source

https://www.bde.es/webbe/en/estadisticas/recursos/api-estadisticas-bde.html

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
bde_data("D_1NBAF472", time_range = "30M")
bde_data(c("DTNPDE2010_P0000P_PS_APU", "DTNSEC2010_S0000P_APU_SUMAMOVIL"), time_range = "MAX")
bde_data("DEEQ.N.ES.W1.S1.S1.T.B.G._Z._Z._Z.EUR._T._X.N.ALL", time_range = 2024)
```

---

bde_latest *Fetch latest Banco de España (BdE) data*

Description

Retrieve the most recently published value for one or more series from the BdE statistics API.

Usage

```
bde_latest(key, lang = "en")
```

Arguments

key	(character()) The series keys to query.
lang	(character(1)) Language to query, either "en" or "es".

Value

A data.table::data.table() with the latest observation per series.

Source

https://www.bde.es/webbe/en/estadisticas/recursos/api-estadisticas-bde.html

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
bde_latest("D_1NBAF472")
bde_latest(c("D_1NBAF472", "DTNPDE2010_P0000P_PS_APU"))
```

---

bdf_codelist *Fetch Banque de France (BdF) codelists*

Description

Fetch Banque de France (BdF) codelists

Usage

```
bdf_codelist(..., lang = "en")
```

Arguments

...	(any) Extra arguments appended to the API request. Combined with the default arguments with modifyList().
lang	(character(1)) Language to query. Default "en".

Value

A data.table::data.table() with the requested data.

Source

https://webstat.banque-france.fr/en/pages/guide-migration-api/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
## Not run:
bdf_codelist()

# filter for a specific codelist
bdf_codelist(where = "codelist_id = 'CL_FREQ'")

## End(Not run)
```

---

bdf_data *Fetch Banque de France (BdF) data*

Description

Retrieve time series data from the BdF Webstat API.

Usage

```
bdf_data(
  ...,
  key = NULL,
  start_date = NULL,
  end_date = NULL,
  lang = "en",
  api_key = bdf_key()
)
```

Arguments

...	(any) Extra arguments appended to the API request. Combined with the default arguments with modifyList().
key	(NULL | character(1)) The series key to query. Default NULL.
start_date	(NULL | character(1) | Date(1)) Start date of the data. Default NULL.
end_date	(NULL | character(1) | Date(1)) End date of the data. Default NULL.

lang (character(1))
Language to query. Default "en".

api_key (character(1))
API key to use for the request. Defaults to the value returned by bdf_key(), which reads from the BANQUEDEFRANCE_KEY environment variable.

Value

A data.table::data.table() with the requested data.

Source

https://webstat.banque-france.fr/en/pages/guide-migration-api/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
## Not run:
bdf_data(key = "CONJ2.M.R24.T.SM.0RG24.EFTPM100.10")

# inflation rate
bdf_data(key = "ICP.M.FR.N.000000.4.ANR")

# or with a date filter
bdf_data(key = "ICP.M.FR.N.000000.4.ANR", start_date = "2025-01-01", end_date = "2025-06-30")

# advanced filter with where clause
bdf_data(key = "ICP.M.FR.N.000000.4.ANR", where = "time_period_start >= date'2025-01-01'")

## End(Not run)
```

---

bdf_dataset *Fetch Banque de France (BdF) datasets*

Description

Fetch Banque de France (BdF) datasets

Usage

```
bdf_dataset(..., lang = "en")
```

Arguments

`...`	(any) Extra arguments appended to the API request. Combined with the default arguments with `modifyList()`.
`lang`	(`character(1)`) Language to query. Default `"en"`.

Value

A `data.table::data.table()` with the requested data.

Source

https://webstat.banque-france.fr/en/pages/guide-migration-api/

See Also

Other data: `banxico_data()`, `bbk_data()`, `bbk_series()`, `bcb_data()`, `bcb_expectations()`, `bcb_fx_rates()`, `bcb_inflation()`, `bcb_selic()`, `bcb_top5()`, `bde_data()`, `bde_latest()`, `bdf_codelist()`, `bdf_data()`, `bdp_data()`, `bis_data()`, `boc_data()`, `boe_data()`, `boi_data()`, `boj_data()`, `cnb_czeonia()`, `cnb_data()`, `cnb_fx_other_rates()`, `cnb_fx_rates()`, `cnb_pribor()`, `ecb_data()`, `nbb_data()`, `nbp_fx_rates()`, `nbp_gold()`, `nob_data()`, `onb_data()`, `snb_data()`, `srb_cross_rates()`, `srb_data()`

Examples

```
## Not run:
bdf_dataset()

# structure of a dataset
bdf_dataset(where = "dataset_id = 'CONJ2'")

## End(Not run)
```

`bdf_dimension` *Fetch Banque de France (BdF) dimensions*

Description

Retrieve the dimension structure for a given dataset from the BdF Webstat API.

Usage

```
bdf_dimension(dataset_id, lang = "en", api_key = bdf_key())
```

Arguments

dataset_id	(character(1)) The id of the dataset to query (e.g., "CONJ2"). See bdf_dataset() for available datasets.
lang	(character(1)) Language to query. Default "en".
api_key	(character(1)) API key to use for the request. Defaults to the value returned by bdf_key(), which reads from the BANQUEDEFRANCE_KEY environment variable.

Value

A data.table::data.table() with columns:

id	The dimension id (e.g., "FREQ", "REF_AREA")
position	The position of the dimension in the series key
codelist	The id of the associated codelist

Source

https://webstat.banque-france.fr/en/pages/guide-migration-api/

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
## Not run:
bdf_dimension("CONJ2")

## End(Not run)
```

---

bdp_data *Fetch Banco de Portugal (BdP) data*

Description

Retrieve time series data from the BPstat API.

Usage

```
bdp_data(
  domain_id,
  dataset_id,
  series_ids = NULL,
  start_date = NULL,
  end_date = NULL,
  last_n = NULL,
  updated_after = NULL,
  lang = "en"
)
```

Arguments

domain_id	(integer(1)) The domain ID. Use bdp_domain() to list available domains.
dataset_id	(character(1)) The dataset ID within the domain.
series_ids	(NULL | integer()) Optional series IDs to filter the dataset. If NULL, every series in the dataset is returned.
start_date	(NULL | character(1) | Date(1)) Start date of the data.
end_date	(NULL | character(1) | Date(1)) End date of the data.
last_n	(NULL | integer(1)) Return only the last n observations per series.
updated_after	(NULL | character(1) | Date(1) | POSIXct(1)) Retrieve only observations published after the given timestamp (e.g., "2024-06-01T00:00:00"). Useful for incremental retrieval. If NULL, no restriction is applied. Default NULL.
lang	(character(1)) Language for labels, either "en" or "pt".

Details

The BPstat API uses a two-step workflow: first look up the series metadata with bdp_series() to find the domain_id and dataset_id, then use those to fetch the actual observations.

You can browse available series at the BPstat portal.

Value

A data.table::data.table() with the requested data.

Source

https://bpstat.bportugal.pt/data/docs

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# Portuguese GDP (annual, current prices)
bdp_data(54L, "ce3e4e50cda325537eff729ef64037cd", series_ids = 12518356L)

# several series at once
bdp_data(19L, "0da378eb4c39011fb7fb371c6623af8f", series_ids = c(12558817L, 12558819L))
```

---

bdp_dataset *Fetch Banco de Portugal (BdP) datasets*

Description

Retrieve the list of datasets for a given domain from the BPstat API.

Usage

```
bdp_dataset(domain_id, lang = "en")
```

Arguments

domain_id	(integer(1)) The domain ID. Use bdp_domain() to list available domains.
lang	(character(1)) Language for labels, either "en" or "pt".

Value

A data.table::data.table() with available datasets.

Source

https://bpstat.bportugal.pt/data/docs

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
bdp_dataset(54L)
```

---

bdp_dimension *Fetch Banco de Portugal (BdP) dimensions*

Description

Retrieve the list of dimensions for a given domain, or the categories within a single dimension.

Usage

```
bdp_dimension(domain_id, dimension_id = NULL, lang = "en")
```

Arguments

domain_id	(integer(1)) The domain ID. Use bdp_domain() to list available domains.
dimension_id	(NULL | integer(1)) Optional dimension ID. If NULL, all dimensions for the domain are returned. If specified, the categories within that dimension are returned.
lang	(character(1)) Language for labels, either "en" or "pt".

Value

A data.table::data.table() with dimensions or categories.

Source

https://bpstat.bportugal.pt/data/docs

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
bdp_dimension(54L)
```

---

bdp_domain *Fetch Banco de Portugal (BdP) domains*

Description

Retrieve the list of available statistical domains from the BPstat API, or details for a single domain.

Usage

```
bdp_domain(domain_id = NULL, lang = "en")
```

Arguments

domain_id	(NULL | integer(1)) Optional domain ID. If NULL, all domains are returned.
lang	(character(1)) Language for labels, either "en" or "pt".

Value

A data.table::data.table() with available domains.

Source

https://bpstat.bportugal.pt/data/docs

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
bdp_domain()
```

`bdp_series` *Fetch Banco de Portugal (BdP) series metadata*

Description

Retrieve metadata for one or more series from the BPstat API. This is useful to discover the `domain_id` and `dataset_id` needed for `bdp_data()`.

Usage

```
bdp_series(series_ids, lang = "en")
```

Arguments

`series_ids`	(`integer()`) One or more series IDs to look up.
`lang`	(`character(1)`) Language for labels, either `"en"` or `"pt"`.

Value

A `data.table::data.table()` with series metadata including `domain_id` and `dataset_id`.

Source

https://bpstat.bportugal.pt/data/docs

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
bdp_series(12518356L)
```

---

bis_data *Fetch Bank for International Settlements (BIS) data*

Description

Retrieve time series data from the BIS SDMX Web Service.

Usage

```
bis_data(
  flow,
  key = NULL,
  start_period = NULL,
  end_period = NULL,
  first_n = NULL,
  last_n = NULL,
  updated_after = NULL
)
```

Arguments

flow	(character(1)) The dataflow to query. See bis_metadata() for available dataflows.
key	(NULL | character()) The series keys to query using dot-separated dimension values (e.g., "M.CH"). Use + for multiple values in one dimension (e.g., "M.CH+US"). If NULL, all data for the flow is returned. Default NULL.
start_period	(NULL | character(1) | integer(1)) Start date of the data. Supported formats: • YYYY for annual data (e.g., 2019)

- YYYY-S[1-2] for semi-annual data (e.g., "2019-S1")
- YYYY-Q[1-4] for quarterly data (e.g., "2019-Q1")
- YYYY-MM for monthly data (e.g., "2019-01")
- YYYY-MM-DD for daily data (e.g., "2019-01-01")

If NULL, no start date restriction is applied (data retrieved from the earliest available date). Default NULL.

end_period (NULL | character(1) | integer(1))
End date of the data, in the same format as start_period. If NULL, no end date restriction is applied (data retrieved up to the most recent available date). Default NULL.

first_n (NULL | numeric(1))
Number of observations to retrieve from the start of the series. If NULL, no restriction is applied. Default NULL.

last_n (NULL | numeric(1))
Number of observations to retrieve from the end of the series. If NULL, no restriction is applied. Default NULL.

updated_after (NULL | character(1) | Date(1) | POSIXct(1))
Retrieve only observations updated after the given timestamp (e.g., "2024-06-01T00:00:00"). Useful for incremental retrieval. If NULL, no restriction is applied. Default NULL.

Value

A data.table::data.table() with the requested data.

Source

https://stats.bis.org/api-doc/v1/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# fetch Swiss central bank policy rate
bis_data("WS_CBPOL", "M.CH", last_n = 5L)

# fetch effective exchange rates
bis_data("WS_EER", "M.N.B.CH", start_period = "2020-01")
```

`bis_dimension` *Fetch Bank for International Settlements (BIS) dimensions*

Description

Retrieve the dimension structure for a given dataflow from the BIS SDMX Web Service.

Usage

```
bis_dimension(id)
```

Arguments

`id`	(`character(1)`) The id of the data structure definition to query (e.g., `"BIS_CBPOL"`).

Value

A `data.table::data.table()` with columns:

`id`	The dimension id (e.g., `"FREQ"`, `"REF_AREA"`)
`position`	The position of the dimension in the series key
`codelist`	The id of the associated codelist (e.g., `"CL_FREQ"`)

Source

https://stats.bis.org/api-doc/v1/

See Also

Other metadata: `banxico_metadata()`, `bbk_dimension()`, `bbk_metadata()`, `bcb_currencies()`, `bdf_dimension()`, `bdp_dataset()`, `bdp_dimension()`, `bdp_domain()`, `bdp_series()`, `bis_metadata()`, `boi_dimension()`, `boi_metadata()`, `boj_metadata()`, `cnb_dimension()`, `cnb_indicators()`, `cnb_snapshots()`, `cnb_tree()`, `ecb_dimension()`, `ecb_metadata()`, `nbb_dimension()`, `nbb_metadata()`, `nob_dimension()`, `nob_metadata()`, `onb_dimension()`, `onb_frequency()`, `onb_hierarchy()`, `onb_metadata()`, `onb_toc()`, `snb_dimension()`, `snb_metadata()`, `snb_toc()`, `srb_calendar()`, `srb_series()`

Examples

```
bis_dimension("BIS_CBPOL")
```

`bis_metadata` *Fetch Bank for International Settlements (BIS) metadata*

Description

Retrieve metadata from the BIS SDMX Web Service.

Usage

```
bis_metadata(type, id = NULL)
```

Arguments

`type`	(`character(1)`) The type of metadata to query. One of: `"datastructure"`, `"dataflow"`, `"codelist"`, or `"concept"`.
`id`	(`NULL` | `character(1)`) The id to query. Default `NULL`.

Value

A `data.table::data.table()` with the requested metadata.

Source

https://stats.bis.org/api-doc/v1/

See Also

Other metadata: `banxico_metadata()`, `bbk_dimension()`, `bbk_metadata()`, `bcb_currencies()`, `bdf_dimension()`, `bdp_dataset()`, `bdp_dimension()`, `bdp_domain()`, `bdp_series()`, `bis_dimension()`, `boi_dimension()`, `boi_metadata()`, `boj_metadata()`, `cnb_dimension()`, `cnb_indicators()`, `cnb_snapshots()`, `cnb_tree()`, `ecb_dimension()`, `ecb_metadata()`, `nbb_dimension()`, `nbb_metadata()`, `nob_dimension()`, `nob_metadata()`, `onb_dimension()`, `onb_frequency()`, `onb_hierarchy()`, `onb_metadata()`, `onb_toc()`, `snb_dimension()`, `snb_metadata()`, `snb_toc()`, `srb_calendar()`, `srb_series()`

Examples

```
bis_metadata("dataflow")
bis_metadata("datastructure", "BIS_CBPOL")
bis_metadata("codelist", "CL_FREQ")
```

boc_catalog *Fetch Bank of Canada (BoC) available series or group*

Description

Access all available series or groups from the Bank of Canada Valet API.

Usage

```
boc_catalog(type = "groups")
```

Arguments

type	(character(1)) Set of data to return. One of "groups" or "series". Default "groups".

Value

A data.table::data.table() with the requested data.

Source

https://www.bankofcanada.ca/valet/docs

Examples

```
## Not run:
catalog = boc_catalog()
head(catalog)

# filter for effective exchange rate series
dt = catalog[grepl("CEER", label)]
head(dt)

## End(Not run)
```

boc_data *Fetch Bank of Canada (BoC) data*

Description

Retrieve time series data from the Bank of Canada Valet API.

Usage

```
boc_data(
  group_name = NULL,
  series_name = NULL,
  start_date = NULL,
  end_date = NULL
)
```

Arguments

group_name	(NULL | character(1)) Name of the group. Only one of group_name or series_name can be used.
series_name	(NULL | character()) Name of the series.
start_date	(NULL | Date(1) | character(1)) Start date of the data. Default NULL.
end_date	(NULL | Date(1) | character(1)) End date of the data. Default NULL.

Value

A data.table::data.table() with the requested data.

Source

https://www.bankofcanada.ca/valet/docs

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
## Not run:
# fetch all data for a single group
dt = boc_data(group_name = "FX_RATES_DAILY")
head(dt)

# or for multiple series ids
dt = boc_data(
  series_name = c("FXUSDCAD", "FXEURCAD"),
  start_date = "2023-01-23",
  end_date = "2023-07-19"
)
```

```
head(dt)

## End(Not run)
```

`boc_fx_rates` *Fetch Bank of Canada foreign exchange rates*

Description

Fetch the latest or historical foreign exchange reference rates from the Bank of Canada (BoC).

Usage

```
boc_fx_rates(start_date = NULL, end_date = NULL, limit = NULL, skip = NULL)
```

Arguments

`start_date`	(`NULL` | `Date(1)` | `character(1)`) Start date of the data. Default `NULL`.
`end_date`	(`NULL` | `Date(1)` | `character(1)`) End date of the data. Default `NULL`.
`limit`	(`NULL` | `integer(1)`) Maximum number of records to return. Default `NULL` (all records).
`skip`	(`NULL` | `integer(1)`) Number of records to skip. Default `NULL` (do not skip any records).

Details

The recorded rates indicate the number of Canadian dollars required to buy a single unit of the foreign currency. New rates are released by the Bank of Canada (BoC) daily at 4:30 pm. The Canada Border Services (CBSA) retrieves these updates between 4:30 pm and 5 pm ET.

BoC provides 23 foreign exchange rates. All other rates are maintained by the CBSA.

Exchange rates from the BoC are updated daily in the system while other exchange rates are updated by the CBSA at set intervals. The updated rates are available for retrieval between 7 pm and 11:59 pm ET.

As BoC publishes exchange rates every business day, it is recommended that exchange rate data be retrieved on a daily basis. This retrieval should occur after 7 pm ET to ensure retrieval of the latest updates.

Value

A `data.table::data.table()` with the exchange rates.

Source

https://www.cbsa-asfc.gc.ca/eservices/api/er-tc-api-eng.html

Examples

```
# fetch latest exchange rates
boc_fx_rates()

# fetch historical exchange rates
boc_fx_rates(start_date = "2021-10-22", end_date = "2021-10-23", limit = 10, skip = 2)
```

`boc_metadata` *Fetch Bank of Canada (BoC) metadata (details)*

Description

Fetch Bank of Canada (BoC) metadata (details)

Usage

```
boc_metadata(group_name = NULL, series_name = NULL)
```

Arguments

`group_name`	(`NULL` | `character(1)`) Name of the group. Only one of `group_name` or `series_name` can be used.
`series_name`	(`NULL` | `character(1)`) Name of the series. Only one of `group_name` or `series_name` can be used.

Value

A `data.table::data.table()` with the requested data.

Source

https://www.bankofcanada.ca/valet/docs

Examples

```
## Not run:
boc_metadata(group_name = "FX_RATES_DAILY")
boc_metadata(series_name = "FXUSDCAD")

## End(Not run)
```

boe_data *Fetch Bank of England (BoE) data*

Description

Retrieve time series data from the BoE database.

Usage

```
boe_data(key, start_date, end_date = Sys.Date())
```

Arguments

key	(character()) The series keys to query.
start_date	(character(1) | Date(1)) Start date of the data.
end_date	(character(1) | Date(1)) End date of the data. Default is today's date.

Value

A data.table::data.table() with the requested data.

Source

https://www.bankofengland.co.uk/boeapps/database

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# Bank Rate
boe_data("IUDBEDR", "2015-01-01")

# SONIA daily rate
boe_data("IUDSOIA", "2015-01-01")

# 10-year nominal par yield
boe_data("IUDMNPY", "2015-01-01")
```

```
# multiple series
boe_data(c("IUMABEDR", "IUALBEDR"), "2015-01-01")
```

`boi_data` *Fetch Bank of Israel (BoI) data*

Description

Retrieve time series data from the Bank of Israel SDMX Web Service.

Usage

```
boi_data(
  flow,
  key = NULL,
  start_period = NULL,
  end_period = NULL,
  first_n = NULL,
  last_n = NULL
)
```

Arguments

`flow`	(`character(1)`) The dataflow to query. See `boi_metadata()` for available dataflows.
`key`	(`NULL` | `character(1)`) The series key to query using dot-separated dimension values (e.g., `"RER_GBP_ILS"`). Use `+` for multiple values in one dimension. If `NULL`, all data for the flow is returned. Default `NULL`.
`start_period`	(`NULL` | `character(1)` | `integer(1)`) Start date of the data (e.g., `"2024-01-01"` or `2024`). If `NULL`, no start date restriction is applied. Default `NULL`.
`end_period`	(`NULL` | `character(1)` | `integer(1)`) End date of the data, in the same format as start_period. If `NULL`, no end date restriction is applied. Default `NULL`.
`first_n`	(`NULL` | `numeric(1)`) Number of observations to retrieve from the start of the series. If `NULL`, no restriction is applied. Default `NULL`.
`last_n`	(`NULL` | `numeric(1)`) Number of observations to retrieve from the end of the series. If `NULL`, no restriction is applied. Default `NULL`.

Value

A data.table::data.table() with the requested data.

Source

https://www.boi.org.il/en/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# fetch GBP/ILS exchange rate
boi_data("EXR", "RER_GBP_ILS", last_n = 5L)

# fetch a range
boi_data("EXR", "RER_GBP_ILS", start_period = "2024-01-01", end_period = "2024-01-31")
```

---

boi_dimension *Fetch Bank of Israel (BoI) dimensions*

Description

Retrieve the dimension structure for a given dataflow from the Bank of Israel SDMX Web Service.

Usage

```
boi_dimension(id)
```

Arguments

id	(character(1)) The id of the data structure definition to query (e.g., "EXR").

Value

A data.table::data.table() with columns:

id	The dimension id (e.g., "FREQ", "BASE_CURRENCY")
position	The position of the dimension in the series key
codelist	The id of the associated codelist (e.g., "CL_FREQ")

Source

https://www.boi.org.il/en/

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
boi_dimension("EXR")
```

---

boi_metadata *Fetch Bank of Israel (BoI) metadata*

Description

Retrieve metadata from the Bank of Israel SDMX Web Service.

Usage

```
boi_metadata(type, id = NULL, lang = "en")
```

Arguments

type	(character(1)) The type of metadata to query. One of: "datastructure", "dataflow", "codelist", or "concept".
id	(NULL | character(1)) The id to query. Default NULL.
lang	(character(1)) Language for names, either "en" or "he". Default "en".

Value

A data.table::data.table() with the requested metadata.

Source

https://www.boi.org.il/en/

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
boi_metadata("dataflow")
boi_metadata("datastructure")
boi_metadata("codelist", "CL_FREQ")
```

---

boj_data *Fetch Bank of Japan (BoJ) data*

Description

Retrieve time series data from the Bank of Japan Statistics API.

Usage

```
boj_data(db, code, start_date = NULL, end_date = NULL, lang = "en")
```

Arguments

db	(character(1)) The database code to query (e.g., "FM08" for foreign exchange rates). See the API manual for available databases.
code	(character()) One or more series codes to query (e.g., "FXERD01" for USD/JPY spot rate). Maximum 250 codes per request. All codes must have the same frequency. Use boj_metadata() to find available codes.
start_date	(NULL | character(1) | integer(1)) Start date of the data. Format depends on frequency: "YYYYMM" for daily, weekly, and monthly data, "YYYYQQ" for quarterly (where QQ is 01-04), and "YYYY" for annual. If NULL, all available data is returned. Default NULL.
end_date	(NULL | character(1) | integer(1)) End date of the data, in the same format as start_date. If NULL, data up to the latest available date is returned. Default NULL.
lang	(character(1)) Language for series names, either "en" or "jp". Default "en".

Value

A data.table::data.table() with the requested data.

Source

https://www.stat-search.boj.or.jp/index_en.html

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# fetch USD/JPY exchange rate
boj_data("FM08", "FXERD01", start_date = "202401")

# fetch multiple exchange rates
boj_data("FM08", c("FXERD01", "FXERD02"), start_date = "202401")
```

---

boj_metadata *Fetch Bank of Japan (BoJ) metadata*

Description

Retrieve series metadata from the Bank of Japan Statistics API.

Usage

```
boj_metadata(db, lang = "en")
```

Arguments

db	(character(1)) The database code to query (e.g., "FM08" for foreign exchange rates).
lang	(character(1)) Language for names, either "en" or "jp". Default "en".

Value

A data.table::data.table() with the requested metadata.

Source

https://www.stat-search.boj.or.jp/index_en.html

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
boj_metadata("FM08")
```

---

cache *Get or manage the bbk API cache*

Description

`bbk_cache_dir()` returns the path where cached API responses are stored. `bbk_cache_clear()` clears all cached responses.

Usage

```
bbk_cache_dir()

bbk_cache_clear()
```

Details

The cache is only used when enabled with `options(bbk.cache = TRUE)`.

Each API's caching headers decide whether and for how long a cached response is reused, so the cache helps more with some APIs than others. APIs that allow reuse do so for minutes rather than hours. Others check a cached response with the API on every call, which only saves downloading it again, and many don't allow caching at all.

Cached responses older than 1 day are deleted. Change this with `options(bbk.cache_max_age = seconds)`. A higher value doesn't make responses be reused for longer than the API allows.

Value

`bbk_cache_dir()` returns a string with the path to the cache directory.

`bbk_cache_clear()` is called for its side effect of clearing the cached responses and returns `NULL` invisibly.

Examples

```
## Not run:
# enable caching
options(bbk.cache = TRUE)

# view cache location
bbk_cache_dir()

# clear the cache
bbk_cache_clear()

## End(Not run)
```

`cnb_czeonia` *Fetch Czech National Bank (CNB) CZEONIA rates*

Description

Retrieve the Czech Overnight Index Average (CZEONIA) reference rate from the CNB API.

Usage

```
cnb_czeonia(date = NULL, year = NULL)
```

Arguments

`date`	(`NULL` | `character(1)` | `Date(1)`) The date to query. If `NULL`, the latest available rates are returned. Mutually exclusive with `year`. Default `NULL`.
`year`	(`NULL` | `integer(1)`) A calendar year, returning rates for every working day of that year. Mutually exclusive with `date`. Default `NULL`.

Value

A `data.table::data.table()` with the requested rates. The `czeonia` column holds the rate in percent and `volume` the trading volume in millions of Czech koruna.

Source

https://api.cnb.cz/cnbapi/swagger-ui.html

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# latest rate
cnb_czeonia()

# all rates for a given year
cnb_czeonia(year = 2024L)
```

`cnb_data` *Fetch Czech National Bank (CNB) ARAD time series data*

Description

Retrieve time series observations from the CNB ARAD database. ARAD is the CNB's full statistical database, covering monetary, financial-market, balance-of-payments, and government finance statistics. Access requires an API key, which can be generated from a free account at https://www.cnb.cz/arad/.

Usage

```
cnb_data(
  indicator_id = NULL,
  set_id = NULL,
  base_id = NULL,
  selection_id = NULL,
  start_period = NULL,
  end_period = NULL,
  snapshot_id = NULL,
  api_key = cnb_arad_key()
)
```

Arguments

`indicator_id`	(`NULL` | `character()`) One or more indicator identifiers to retrieve (e.g. `"SMV5M603"`). Exactly one of `indicator_id`, `set_id`, `base_id`, or `selection_id` must be provided.

set_id (NULL | character(1))
A set ("sestava") identifier, returning all of its indicators.

base_id (NULL | character(1))
A base identifier, returning all of its indicators.

selection_id (NULL | character(1))
The identifier of a named selection ("My selections") created in your ARAD account.

start_period (NULL | character(1) | Date(1))
Start of the period to retrieve. If NULL, no start restriction is applied. Default NULL.

end_period (NULL | character(1) | Date(1))
End of the period to retrieve, in the same format as start_period. Default NULL.

snapshot_id (NULL | character())
One or more snapshot ids to retrieve historical vintages, or "ALL" for every snapshot. If NULL, the current (non-snapshot) data is returned. See cnb_snapshots(). Default NULL.

api_key (character(1))
API key to use for the request. Defaults to the value returned by cnb_arad_key(), which reads from the CNB_ARAD_KEY environment variable.

Value

A data.table::data.table() with columns:

date The observation period

indicator_id The indicator identifier

snapshot_id The snapshot identifier, or NA for non-snapshot data

value The observation value

Source

https://www.cnb.cz/docs/arad20/dokumentace/arad_rest_api_cs.pdf

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
## Not run:
# a single indicator over 24 months
cnb_data("SMV5M603", start_period = "2023-01-01")

# every indicator in a set
cnb_data(set_id = "1058")

## End(Not run)
```

`cnb_dimension` *Fetch Czech National Bank (CNB) ARAD dimensions*

Description

Retrieve the dimension structure of indicators from the CNB ARAD database, returning one row per indicator dimension.

Usage

```
cnb_dimension(
  indicator_id = NULL,
  set_id = NULL,
  base_id = NULL,
  selection_id = NULL,
  lang = "en",
  api_key = cnb_arad_key()
)
```

Arguments

`indicator_id`	(`NULL` | `character()`) One or more indicator identifiers to retrieve (e.g. `"SMV5M603"`). Exactly one of `indicator_id`, `set_id`, `base_id`, or `selection_id` must be provided.
`set_id`	(`NULL` | `character(1)`) A set ("sestava") identifier, returning all of its indicators.
`base_id`	(`NULL` | `character(1)`) A base identifier, returning all of its indicators.
`selection_id`	(`NULL` | `character(1)`) The identifier of a named selection ("My selections") created in your ARAD account.
`lang`	(`character(1)`) Language for the textual attributes, either `"en"` or `"cs"`. Default `"en"`.
`api_key`	(`character(1)`) API key to use for the request. Defaults to the value returned by `cnb_arad_key()`, which reads from the `CNB_ARAD_KEY` environment variable.

Value

A data.table::data.table() with columns including indicator_id, base_code, base_name, dim_code, dim_name, dim_value_code, dim_value_name, and dim_rank.

Source

https://www.cnb.cz/docs/arad20/dokumentace/arad_rest_api_cs.pdf

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
## Not run:
cnb_dimension(indicator_id = "MIRFMDF12ERATPECD")

## End(Not run)
```

---

cnb_fx_other_rates *Fetch Czech National Bank (CNB) exchange rates of other currencies*

Description

Retrieve the monthly exchange rate fixing for less commonly traded ("other") currencies from the CNB API. These currencies are not part of the daily fixing returned by cnb_fx_rates().

Usage

```
cnb_fx_other_rates(year_month = NULL, year = NULL, lang = "EN")
```

Arguments

year_month	(NULL | character(1)) The month to query in "YYYY-MM" format, returning rates for all currencies in that month. The fixing for a month is published at its end. Mutually exclusive with year. Default NULL.
year	(NULL | integer(1)) A calendar year, returning rates for all currencies in every month of that year. Mutually exclusive with year_month. Default NULL.
lang	(character(1)) Language for the country and currency names, either "EN" or "CZ". Default "EN".

Value

A data.table::data.table() with the requested exchange rates. The rate is the amount of Czech koruna per amount units of the foreign currency.

Source

https://api.cnb.cz/cnbapi/swagger-ui.html

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# a specific month
cnb_fx_other_rates(year_month = "2024-01")

# all months of a given year
cnb_fx_other_rates(year = 2024L)
```

---

cnb_fx_rates *Fetch Czech National Bank (CNB) exchange rates*

Description

Retrieve the central bank exchange rate fixing (Czech koruna against foreign currencies) from the CNB API.

Usage

```
cnb_fx_rates(date = NULL, year = NULL, lang = "EN")
```

Arguments

date	(NULL | character(1) | Date(1)) The date to query, returning rates for all currencies on that day. If NULL, the latest available fixing is returned. Mutually exclusive with year. Default NULL.
year	(NULL | integer(1)) A calendar year, returning rates for all currencies on every working day of that year. Mutually exclusive with date. Default NULL.

lang (character(1))
Language for the country and currency names, either "EN" or "CZ". With year, the names are taken from the first and last fixings of that year. Default "EN".

Value

A data.table::data.table() with the requested exchange rates. The rate is the amount of Czech koruna per amount units of the foreign currency.

Source

https://api.cnb.cz/cnbapi/swagger-ui.html

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# latest fixing for all currencies
cnb_fx_rates()

# all fixings for a given year
cnb_fx_rates(year = 2024L)
```

---

cnb_indicators *Fetch Czech National Bank (CNB) ARAD indicators*

Description

Retrieve the available indicators and their attributes from the CNB ARAD database. Use this to discover indicator identifiers for cnb_data().

Usage

```
cnb_indicators(
  indicator_id = NULL,
  set_id = NULL,
  base_id = NULL,
  selection_id = NULL,
```

```
  lang = "en",
  api_key = cnb_arad_key()
)
```

Arguments

indicator_id	(NULL | character()) One or more indicator identifiers to retrieve (e.g. "SMV5M603"). Exactly one of indicator_id, set_id, base_id, or selection_id must be provided.
set_id	(NULL | character(1)) A set ("sestava") identifier, returning all of its indicators.
base_id	(NULL | character(1)) A base identifier, returning all of its indicators.
selection_id	(NULL | character(1)) The identifier of a named selection ("My selections") created in your ARAD account.
lang	(character(1)) Language for the textual attributes, either "en" or "cs". Default "en".
api_key	(character(1)) API key to use for the request. Defaults to the value returned by cnb_arad_key(), which reads from the CNB_ARAD_KEY environment variable.

Value

A data.table::data.table() with one row per indicator and columns including indicator_id, indicator_name, frequency_code, frequency_name, unit_mult_code, unit_mult_name, and unit.

Source

https://www.cnb.cz/docs/arad20/dokumentace/arad_rest_api_cs.pdf

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
## Not run:
cnb_indicators(set_id = "1058")

## End(Not run)
```

cnb_pribor *Fetch Czech National Bank (CNB) PRIBOR rates*

Description

Retrieve the Prague Interbank Offered Rate (PRIBOR) reference rates for all maturities from the CNB API.

Usage

```
cnb_pribor(date = NULL, year = NULL)
```

Arguments

date	(NULL | character(1) | Date(1)) The date to query. If NULL, the latest available rates are returned. Mutually exclusive with year. Default NULL.
year	(NULL | integer(1)) A calendar year, returning rates for every working day of that year. Mutually exclusive with date. Default NULL.

Value

A data.table::data.table() with the requested rates. The period column holds the maturity (e.g. "ONE_DAY", "THREE_MONTH") and pribor the rate in percent.

Source

https://api.cnb.cz/cnbapi/swagger-ui.html

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# latest rates for all maturities
cnb_pribor()

# all rates for a given year
cnb_pribor(year = 2024L)
```

cnb_snapshots *Fetch Czech National Bank (CNB) ARAD snapshots*

Description

Retrieve the list of available snapshots (data vintages) from the CNB ARAD database. Snapshot ids can be passed to the `snapshot_id` argument of `cnb_data()`.

Usage

```
cnb_snapshots(lang = "en", api_key = cnb_arad_key())
```

Arguments

`lang`	(`character(1)`) Language for the textual attributes, either `"en"` or `"cs"`. Default `"en"`.
`api_key`	(`character(1)`) API key to use for the request. Defaults to the value returned by `cnb_arad_key()`, which reads from the `CNB_ARAD_KEY` environment variable.

Value

A `data.table::data.table()` with columns `snapshot_id` and `snapshot_name`.

Source

https://www.cnb.cz/docs/arad20/dokumentace/arad_rest_api_cs.pdf

See Also

Other metadata: `banxico_metadata()`, `bbk_dimension()`, `bbk_metadata()`, `bcb_currencies()`, `bdf_dimension()`, `bdp_dataset()`, `bdp_dimension()`, `bdp_domain()`, `bdp_series()`, `bis_dimension()`, `bis_metadata()`, `boi_dimension()`, `boi_metadata()`, `boj_metadata()`, `cnb_dimension()`, `cnb_indicators()`, `cnb_tree()`, `ecb_dimension()`, `ecb_metadata()`, `nbb_dimension()`, `nbb_metadata()`, `nob_dimension()`, `nob_metadata()`, `onb_dimension()`, `onb_frequency()`, `onb_hierarchy()`, `onb_metadata()`, `onb_toc()`, `snb_dimension()`, `snb_metadata()`, `snb_toc()`, `srb_calendar()`, `srb_series()`

Examples

```
## Not run:
cnb_snapshots()

## End(Not run)
```

cnb_tree *Fetch Czech National Bank (CNB) ARAD indicator tree*

Description

Retrieve the placement of indicators within the ARAD topic tree.

Usage

```
cnb_tree(
  indicator_id = NULL,
  set_id = NULL,
  base_id = NULL,
  selection_id = NULL,
  lang = "en",
  api_key = cnb_arad_key()
)
```

Arguments

indicator_id	(NULL | character()) One or more indicator identifiers to retrieve (e.g. "SMV5M603"). Exactly one of indicator_id, set_id, base_id, or selection_id must be provided.
set_id	(NULL | character(1)) A set ("sestava") identifier, returning all of its indicators.
base_id	(NULL | character(1)) A base identifier, returning all of its indicators.
selection_id	(NULL | character(1)) The identifier of a named selection ("My selections") created in your ARAD account.
lang	(character(1)) Language for the textual attributes, either "en" or "cs". Default "en".
api_key	(character(1)) API key to use for the request. Defaults to the value returned by cnb_arad_key(), which reads from the CNB_ARAD_KEY environment variable.

Value

A data.table::data.table() with columns indicator_id and path, where path is the slash-separated location of the indicator in the ARAD tree.

Source

https://www.cnb.cz/docs/arad20/dokumentace/arad_rest_api_cs.pdf

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
## Not run:
cnb_tree(indicator_id = "MIRFMDF12ERATPECD")

## End(Not run)
```

---

ecb_data *Fetch European Central Bank (ECB) data*

Description

Retrieve time series data from the ECB SDMX Web Service.

Usage

```
ecb_data(
  flow,
  key = NULL,
  start_period = NULL,
  end_period = NULL,
  first_n = NULL,
  last_n = NULL,
  updated_after = NULL
)
```

Arguments

flow	(character(1)) Flow to query.
key	(NULL | character()) The series keys to query.
start_period	(NULL | character(1) | integer(1)) Start date of the data. Supported formats: • YYYY for annual data (e.g., 2019) • YYYY-S[1-2] for semi-annual data (e.g., "2019-S1") • YYYY-Q[1-4] for quarterly data (e.g., "2019-Q1")

- YYYY-MM for monthly data (e.g., "2019-01")
- YYYY-W[01-53] for weekly data (e.g., "2019-W01")
- YYYY-MM-DD for daily and business data (e.g., "2019-01-01")

If NULL, no start date restriction is applied (data retrieved from the earliest available date). Default NULL.

end_period (NULL | character(1) | integer(1))
End date of the data, in the same format as start_period. If NULL, no end date restriction is applied (data retrieved up to the most recent available date). Default NULL.

first_n (NULL | numeric(1))
Number of observations to retrieve from the start of the series. If NULL, no restriction is applied. Default NULL.

last_n (NULL | numeric(1))
Number of observations to retrieve from the end of the series. If NULL, no restriction is applied. Default NULL.

updated_after (NULL | character(1) | Date(1) | POSIXct(1))
Retrieve only observations updated after the given timestamp (e.g., "2024-06-01T00:00:00"). Useful for incremental retrieval. If NULL, no restriction is applied. Default NULL.

Value

A data.table::data.table() with the requested data.

Source

https://data.ecb.europa.eu/help/api/data

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# fetch US dollar/Euro exchange rate
ecb_data("EXR", "D.USD.EUR.SP00.A")

# fetch data for multiple keys
ecb_data("EXR", c("D.USD", "JPY.EUR.SP00.A"))
```

ecb_dimension *Fetch European Central Bank (ECB) dimensions*

Description

Retrieve the dimension structure for a given dataflow from the ECB SDMX Web Service.

Usage

```
ecb_dimension(id)
```

Arguments

id	(character(1)) The id of the data structure definition to query (e.g., "ECB_EXR1").

Value

A data.table::data.table() with columns:

id	The dimension id (e.g., "FREQ", "CURRENCY")
position	The position of the dimension in the series key
codelist	The id of the associated codelist (e.g., "CL_FREQ")

Source

https://data.ecb.europa.eu/help/api/metadata

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
ecb_dimension("ECB_EXR1")
```

`ecb_fx_rates` *Fetch Euro foreign exchange reference rates*

Description

Fetch the latest or historical Euro foreign exchange reference rates from the European Central Bank (ECB).

Usage

```
ecb_fx_rates(x = "latest")

ecb_euro_rates(x = "latest")
```

Arguments

x	(character(1)) One of "latest" or "history". Default "latest".

Details

Note you can achieve the same by calling the `ecb_data()` function with the right parameters for each currency.

The reference rates are usually updated at around 16:00 CET every working day, except on TARGET closing days.

They are based on the daily concertation procedure between central banks across Europe, which normally takes place around 14:10 CET. The reference rates are published for information purposes only. Using the rates for transaction purposes is strongly discouraged.

Value

A `data.table::data.table()` with the exchange rates.

Source

https://www.ecb.europa.eu/stats/policy_and_exchange_rates/euro_reference_exchange_rates/html/index.en.html

Examples

```
ecb_fx_rates()
```

ecb_metadata *Fetch European Central Bank (ECB) metadata*

Description

Retrieve metadata from the ECB time series database via the SDMX Web Service.

Usage

```
ecb_metadata(type, agency = NULL, id = NULL)
```

Arguments

type	(character(1)) The type of metadata to query. One of: "datastructure", "dataflow", "codelist", or "concept".
agency	(NULL | character(1)) The id of the agency to query. Default NULL.
id	(NULL | character(1)) The id of the resource to query. Default NULL.

Value

A data.table::data.table() with the requested metadata. The columns are:

agency	The agency of the metadata
id	The id of the metadata
name	The name of the metadata

Source

https://data.ecb.europa.eu/help/api/metadata

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
ecb_metadata("datastructure")
ecb_metadata("datastructure", "ECB")
ecb_metadata("datastructure", "ECB", "ECB_EXR1")
ecb_metadata("datastructure", id = "ECB_EXR1")
```

`nbb_data` *Fetch National Bank of Belgium (NBB) data*

Description

Retrieve time series data from the National Bank of Belgium SDMX Web Service.

Usage

```
nbb_data(
  flow,
  key = NULL,
  start_period = NULL,
  end_period = NULL,
  first_n = NULL,
  last_n = NULL
)
```

Arguments

`flow`	(`character(1)`) The dataflow to query (e.g., `"DF_EXR"`). See `nbb_metadata()` for available dataflows.
`key`	(`NULL` | `character(1)`) The series key to query using dot-separated dimension values (e.g., `"D.USD"`). Use `+` for multiple values in one dimension (e.g., `"M.USD+GBP"`). If `NULL`, all data for the flow is returned. Default `NULL`.
`start_period`	(`NULL` | `character(1)` | `integer(1)`) Start date of the data (e.g., `"2024-01"` or `2024`). If `NULL`, no start date restriction is applied. Default `NULL`.
`end_period`	(`NULL` | `character(1)` | `integer(1)`) End date of the data, in the same format as start_period. If `NULL`, no end date restriction is applied. Default `NULL`.
`first_n`	(`NULL` | `numeric(1)`) Number of observations to retrieve from the start of the series. If `NULL`, no restriction is applied. Default `NULL`.
`last_n`	(`NULL` | `numeric(1)`) Number of observations to retrieve from the end of the series. If `NULL`, no restriction is applied. Default `NULL`.

Value

A data.table::data.table() with the requested data.

Source

https://dataexplorer.nbb.be/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# fetch daily EUR/USD reference rate
nbb_data("DF_EXR", "D.USD", last_n = 5L)

# fetch multiple monthly average exchange rates
nbb_data("DF_EXR", "M.USD+GBP", start_period = "2024-01")

# fetch Belgian HICP inflation
nbb_data("DF_HICP_2025", "M.BE.000000.2025.HCP.GROWTH_RATE", last_n = 5L)
```

---

nbb_dimension *Fetch National Bank of Belgium (NBB) dimensions*

Description

Retrieve the dimension structure for a given data structure from the National Bank of Belgium SDMX Web Service.

Usage

```
nbb_dimension(id)
```

Arguments

id	(character(1)) The id of the data structure definition to query (e.g., "DSD_EXR").

Value

A data.table::data.table() with columns:

id	The dimension id (e.g., "FREQ", "EXR_CURRENCY")
position	The position of the dimension in the series key
codelist	The id of the associated codelist (e.g., "CL_FREQ")

Source

https://dataexplorer.nbb.be/

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
nbb_dimension("DSD_EXR")
```

---

nbb_metadata *Fetch National Bank of Belgium (NBB) metadata*

Description

Retrieve metadata from the National Bank of Belgium SDMX Web Service.

Usage

```
nbb_metadata(type, id = NULL, lang = "en")
```

Arguments

type	(character(1)) The type of metadata to query. One of: "datastructure", "dataflow", "codelist", or "concept".
id	(NULL | character(1)) The id to query. Default NULL.
lang	(character(1)) Language for names, one of "en", "fr", or "nl". Default "en".

Value

A data.table::data.table() with the requested metadata.

Source

https://dataexplorer.nbb.be/

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
nbb_metadata("dataflow")
nbb_metadata("datastructure", "DSD_EXR")
nbb_metadata("codelist", "CL_EXR_CURRENCY")
nbb_metadata("dataflow", "DF_EXR", lang = "fr")
```

---

nbp_fx_rates *Fetch National Bank of Poland (NBP) exchange rates*

Description

Retrieve foreign currency exchange rates from the NBP Web API.

Usage

```
nbp_fx_rates(
  table,
  code = NULL,
  start_date = NULL,
  end_date = NULL,
  last_n = NULL
)
```

Arguments

table	(character(1)) Table type: "a" (mid rates, major currencies), "b" (mid rates, less common currencies), or "c" (bid/ask rates).
code	(NULL | character(1)) ISO 4217 currency code (e.g. "usd", "eur"). If NULL, returns all currencies.
start_date	(NULL | character(1) | Date(1)) Start date of the data. If end_date is omitted, only start_date is returned. The API limits the range to 367 days.
end_date	(NULL | character(1) | Date(1)) End date of the data. Requires start_date. If NULL, defaults to start_date.
last_n	(NULL | integer(1)) Return only the last n quotations.

Value

A data.table::data.table() with exchange rates.

Source

https://api.nbp.pl/en.html

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
nbp_fx_rates("a", "eur")
```

---

nbp_gold *Fetch National Bank of Poland (NBP) gold prices*

Description

Retrieve the price of gold from the NBP Web API.

Usage

```
nbp_gold(start_date = NULL, end_date = NULL, last_n = NULL)
```

Arguments

start_date	(NULL | character(1) | Date(1)) Start date of the data. If end_date is omitted, only start_date is returned. The API limits the range to 367 days.
end_date	(NULL | character(1) | Date(1)) End date of the data. Requires start_date. If NULL, defaults to start_date.
last_n	(NULL | integer(1)) Return only the last n quotations.

Value

A data.table::data.table() with gold prices.

Source

https://api.nbp.pl/en.html

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nob_data(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
nbp_gold(last_n = 10L)
```

---

nob_data *Fetch Norges Bank (NoB) data*

Description

Retrieve time series data from the Norges Bank SDMX Web Service.

Usage

```
nob_data(
  flow,
  key = NULL,
  start_period = NULL,
  end_period = NULL,
  first_n = NULL,
  last_n = NULL
)
```

Arguments

flow	(character(1)) The dataflow to query. See nob_metadata() for available dataflows.
key	(NULL | character(1)) The series key to query using dot-separated dimension values (e.g., "B.USD.NOK.SP"). Use + for multiple values in one dimension (e.g., "B.USD+EUR.NOK.SP"). If NULL, all data for the flow is returned. Default NULL.
start_period	(NULL | character(1) | integer(1)) Start date of the data (e.g., "2024-01-01" or 2024). If NULL, no start date restriction is applied. Default NULL.
end_period	(NULL | character(1) | integer(1)) End date of the data, in the same format as start_period. If NULL, no end date restriction is applied. Default NULL.
first_n	(NULL | numeric(1)) Number of observations to retrieve from the start of the series. If NULL, no restriction is applied. Default NULL.
last_n	(NULL | numeric(1)) Number of observations to retrieve from the end of the series. If NULL, no restriction is applied. Default NULL.

Value

A data.table::data.table() with the requested data.

Source

https://www.norges-bank.no/en/topics/Statistics/open-data/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), onb_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
# fetch USD/NOK exchange rate
nob_data("EXR", "B.USD.NOK.SP", last_n = 5L)

# fetch multiple exchange rates
nob_data("EXR", "B.USD+EUR+GBP.NOK.SP", start_period = "2024-01-01")

# fetch policy rate
nob_data("IR", last_n = 5L)
```

---

nob_dimension *Fetch Norges Bank (NoB) dimensions*

Description

Retrieve the dimension structure for a given dataflow from the Norges Bank SDMX Web Service.

Usage

```
nob_dimension(id)
```

Arguments

id	(character(1)) The id of the data structure definition to query (e.g., "DSD_EXR").

Value

A data.table::data.table() with columns:

id	The dimension id (e.g., "FREQ", "BASE_CUR")
position	The position of the dimension in the series key
codelist	The id of the associated codelist (e.g., "CL_FREQ")

Source

https://www.norges-bank.no/en/topics/Statistics/open-data/

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
nob_dimension("DSD_EXR")
```

`nob_metadata` *Fetch Norges Bank (NoB) metadata*

Description

Retrieve metadata from the Norges Bank SDMX Web Service.

Usage

```
nob_metadata(type, id = NULL, lang = "en")
```

Arguments

`type`	(`character(1)`) The type of metadata to query. One of: `"datastructure"`, `"dataflow"`, `"codelist"`, or `"concept"`.
`id`	(`NULL` | `character(1)`) The id to query. Default `NULL`.
`lang`	(`character(1)`) Language for names, either `"en"` or `"no"`. Default `"en"`.

Value

A `data.table::data.table()` with the requested metadata.

Source

https://www.norges-bank.no/en/topics/Statistics/open-data/

See Also

Other metadata: `banxico_metadata()`, `bbk_dimension()`, `bbk_metadata()`, `bcb_currencies()`, `bdf_dimension()`, `bdp_dataset()`, `bdp_dimension()`, `bdp_domain()`, `bdp_series()`, `bis_dimension()`, `bis_metadata()`, `boi_dimension()`, `boi_metadata()`, `boj_metadata()`, `cnb_dimension()`, `cnb_indicators()`, `cnb_snapshots()`, `cnb_tree()`, `ecb_dimension()`, `ecb_metadata()`, `nbb_dimension()`, `nbb_metadata()`, `nob_dimension()`, `onb_dimension()`, `onb_frequency()`, `onb_hierarchy()`, `onb_metadata()`, `onb_toc()`, `snb_dimension()`, `snb_metadata()`, `snb_toc()`, `srb_calendar()`, `srb_series()`

Examples

```
nob_metadata("dataflow")
nob_metadata("datastructure")
nob_metadata("codelist", "CL_CURRENCY")
```

---

onb_data *Fetch Österreichische Nationalbank (OeNB) data*

Description

Retrieve time series data from the OeNB Web Service.

Usage

```
onb_data(
  hier_id,
  key,
  ...,
  start_period = NULL,
  end_period = NULL,
  freq = NULL,
  lang = "en"
)
```

Arguments

hier_id	(integer(1)) Hierarchy id to query.
key	(character()) The series keys to query.
...	(any) Additional parameters to pass to the API.
start_period	(NULL | character(1) | integer(1)) Start date of the data.
end_period	(NULL | character(1) | integer(1)) End date of the data.
freq	(NULL | character(1)) Frequency of the data.
lang	(character(1)) Language to query. Default "en".

Details

A key can cover several series, such as one per partner country, which share the same key and differ only in their attr columns. Group by key and the attr columns to separate them.

Value

A data.table::data.table() with the requested data.

Source

https://www.oenb.at/en/daten-und-forschung/individuelle-datenabfrage/webservice.html

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), snb_data(), srb_cross_rates(), srb_data()

Examples

```
onb_data(hier_id = 11, key = "VDBFKBSC217000")

# Loans to euro area residents, since 2000:
onb_data(hier_id = 11, key = "VDBFKBSC217000", start_period = "2000-01-01")

# Austrian imports and exports of goods from/to Germany, 2002-2012, annual frequency:
onb_data(hier_id = 901, key = "VDBQZA1000", start_period = 2002, end_period = 2012, freq = "A")

# Number of Austrian banks' subsidiaries abroad and in the EU, from 2005, semiannual:
onb_data(
  hier_id = 321,
  key = c("VDBKISDANZTAU", "VDBKISDANZTEU"),
  start_period = 200501,
  freq = "H"
)
```

---

onb_dimension *Fetch Österreichische Nationalbank (OeNB) dimension*

Description

Fetch Österreichische Nationalbank (OeNB) dimension

Usage

```
onb_dimension(hier_id, key, lang = "en")
```

Arguments

hier_id	(integer(1)) Hierarchy id to query.
key	(character()) The series keys to query.
lang	(character(1)) Language to query. Default "en".

Value

A data.table::data.table() with the requested data.

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
onb_dimension(hier_id = 11, key = "VDBFKBSC217000")
```

---

onb_frequency *Fetch Österreichische Nationalbank (OeNB) data frequency*

Description

Fetch Österreichische Nationalbank (OeNB) data frequency

Usage

```
onb_frequency(hier_id, key, ..., lang = "en")
```

Arguments

hier_id	(integer(1)) Hierarchy id to query.
key	(character()) The series keys to query.
...	(any) Additional parameters to pass to the API.
lang	(character(1)) Language to query. Default "en".

Value

A data.table::data.table() with the requested data.

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
onb_frequency(hier_id = 74, key = "VDBOSBHAGBSTIN")
onb_frequency(hier_id = 11, key = "VDBFKBSC217000")
```

---

onb_hierarchy *Fetch Österreichische Nationalbank (OeNB) hierarchy*

Description

Fetch Österreichische Nationalbank (OeNB) hierarchy

Usage

```
onb_hierarchy(hier_id, lang = "en")
```

Arguments

hier_id	(integer(1)) Hierarchy id to query.
lang	(character(1)) Language to query. Default "en".

Value

A data.table::data.table() with the requested data.

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
onb_hierarchy(hier_id = 11)
```

---

onb_metadata *Fetch Österreichische Nationalbank (OeNB) metadata*

Description

Fetch Österreichische Nationalbank (OeNB) metadata

Usage

```
onb_metadata(hier_id, key, ..., lang = "en")
```

Arguments

hier_id	(integer(1)) Hierarchy id to query.
key	(character()) The series keys to query.
...	(any) Additional parameters to pass to the API.
lang	(character(1)) Language to query. Default "en".

Value

A data.table::data.table() with the requested data.

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
onb_metadata(hier_id = 11, key = "VDBFKBSC217000")
```

---

onb_toc *Fetch Österreichische Nationalbank (OeNB) table of contents*

Description

Fetch Österreichische Nationalbank (OeNB) table of contents

Usage

```
onb_toc(lang = "en")
```

Arguments

lang	(character(1)) Language to query. Default "en".

Value

A data.table::data.table() with the requested data.

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar(), srb_series()

Examples

```
onb_toc()
```

---

snb_data *Fetch Swiss National Bank (SNB) data*

Description

Retrieve time series data from the SNB data portal.

Usage

```
snb_data(key, start_date = NULL, end_date = NULL, lang = "en")
```

Arguments

key	(character(1)) The series key to query.
start_date	(NULL | character(1) | Date(1)) Start date of the data.
end_date	(NULL | character(1) | Date(1)) End date of the data.
lang	(character(1)) Language to query, either "en" or "de". Default "en".

Value

A data.table::data.table() with the requested data.

Source

https://data.snb.ch/en

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), srb_cross_rates(), srb_data()

Examples

```
snb_data("rendopar")

# or filter for date range
snb_data("rendopar", "2020-01-01", "2020-12-31")
```

`snb_dimension` *Fetch Swiss National Bank (SNB) dimensions*

Description

Retrieve the dimension structure for a given cube from the SNB data portal.

Usage

```
snb_dimension(key, lang = "en")
```

Arguments

`key`	(`character(1)`) The series key to query.
`lang`	(`character(1)`) Language to query, either `"en"` or `"de"`. Default `"en"`.

Value

A `data.table::data.table()` with the dimension structure.

Source

https://data.snb.ch/en

See Also

Other metadata: `banxico_metadata()`, `bbk_dimension()`, `bbk_metadata()`, `bcb_currencies()`, `bdf_dimension()`, `bdp_dataset()`, `bdp_dimension()`, `bdp_domain()`, `bdp_series()`, `bis_dimension()`, `bis_metadata()`, `boi_dimension()`, `boi_metadata()`, `boj_metadata()`, `cnb_dimension()`, `cnb_indicators()`, `cnb_snapshots()`, `cnb_tree()`, `ecb_dimension()`, `ecb_metadata()`, `nbb_dimension()`, `nbb_metadata()`, `nob_dimension()`, `nob_metadata()`, `onb_dimension()`, `onb_frequency()`, `onb_hierarchy()`, `onb_metadata()`, `onb_toc()`, `snb_metadata()`, `snb_toc()`, `srb_calendar()`, `srb_series()`

Examples

```
snb_dimension("rendopar")
```

`snb_metadata` *Fetch Swiss National Bank (SNB) cube metadata*

Description

Retrieve cube-level metadata (title, frequency, source, publication date) from the SNB data portal.

Usage

```
snb_metadata(key, lang = "en")
```

Arguments

`key`	(`character(1)`) The series key to query.
`lang`	(`character(1)`) Language to query, either `"en"` or `"de"`. Default `"en"`.

Value

A single-row `data.table::data.table()` with columns:

`key`	The cube key
`title`	The cube title
`sub_title`	The cube subtitle
`publishing_title`	The publishing section title
`public_since_date`	The first publication date
`frequency`	The frequency specification (e.g., `"Day"`, `"Month"`)
`source`	The data source
`has_multiple_sources`	Whether the cube has multiple sources

Source

https://data.snb.ch/en

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_toc(), srb_calendar(), srb_series()

Examples

```
snb_metadata("rendopar")
```

---

snb_toc *Fetch Swiss National Bank (SNB) table of contents*

Description

Retrieve the publication topic tree from the SNB data portal, listing the cubes and charts available under each topic.

Usage

```
snb_toc(lang = "en")
```

Arguments

lang	(character(1)) Language to query, either "en" or "de". Default "en".

Value

A data.table::data.table() with one row per topic node and columns:

topic_id	The top-level topic id
topic	The top-level topic title
sub_topic	The sub-topic title
cube_id	The associated cube id (if any)
chart_id	The associated chart id (if any)
doc_id	The associated documentation id (if any)

Source

https://data.snb.ch/en

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), srb_calendar(), srb_series()

Examples

```
snb_toc()
```

---

srb_calendar *Fetch Sveriges Riksbank (SRb) calendar days*

Description

Retrieve Swedish banking calendar information from the Sveriges Riksbank SWEA API.

Usage

```
srb_calendar(start_date, end_date = NULL)
```

Arguments

start_date	(Date(1) | character(1)) Start date of the range (e.g., "2024-01-01").
end_date	(NULL | Date(1) | character(1)) End date of the range. If NULL, data up to the latest available date is returned. Default NULL.

Value

A data.table::data.table() with the calendar day information.

Source

https://developer.api.riksbank.se/

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_series()

Examples

```
srb_calendar("2024-01-01", "2024-01-31")
```

---

srb_cross_rates *Fetch Sveriges Riksbank (SRb) cross rates*

Description

Compute cross exchange rates between two currency series from the Sveriges Riksbank SWEA API.

Usage

```
srb_cross_rates(series1, series2, start_date, end_date = NULL)
```

Arguments

series1	(character(1)) The first series ID (e.g., "SEKUSDPMI").
series2	(character(1)) The second series ID (e.g., "SEKEURPMI").
start_date	(NULL | Date(1) | character(1)) Start date of the data (e.g., "2024-01-01"). If NULL, all available data is returned. Default NULL.
end_date	(NULL | Date(1) | character(1)) End date of the data, in the same format as start_date. If NULL, data up to the latest available date is returned. Default NULL.

Value

A data.table::data.table() with the cross rate data.

Source

https://developer.api.riksbank.se/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_data()

Examples

```
# USD/EUR cross rate
srb_cross_rates("SEKUSDPMI", "SEKEURPMI", start_date = "2024-01-01", end_date = "2024-01-31")
```

`srb_data` *Fetch Sveriges Riksbank (SRb) data*

Description

Retrieve time series data from the Sveriges Riksbank SWEA API.

Usage

```
srb_data(series, start_date = NULL, end_date = NULL)
```

Arguments

series	(character(1)) The series ID to query. See srb_series() for available series.
start_date	(NULL | Date(1) | character(1)) Start date of the data (e.g., "2024-01-01"). If NULL, all available data is returned. Default NULL.
end_date	(NULL | Date(1) | character(1)) End date of the data, in the same format as start_date. If NULL, data up to the latest available date is returned. Default NULL.

Value

A data.table::data.table() with the requested data.

Source

https://developer.api.riksbank.se/

See Also

Other data: banxico_data(), bbk_data(), bbk_series(), bcb_data(), bcb_expectations(), bcb_fx_rates(), bcb_inflation(), bcb_selic(), bcb_top5(), bde_data(), bde_latest(), bdf_codelist(), bdf_data(), bdf_dataset(), bdp_data(), bis_data(), boc_data(), boe_data(), boi_data(), boj_data(), cnb_czeonia(), cnb_data(), cnb_fx_other_rates(), cnb_fx_rates(), cnb_pribor(), ecb_data(), nbb_data(), nbp_fx_rates(), nbp_gold(), nob_data(), onb_data(), snb_data(), srb_cross_rates()

Examples

```
# fetch USD/SEK exchange rate
srb_data("SEKUSDPMI", start_date = "2024-01-01")

# fetch EUR/SEK exchange rate
srb_data("SEKEURPMI", start_date = "2024-01-01")
```

---

srb_series *Fetch Sveriges Riksbank (SRb) series metadata*

Description

Retrieve available series or group metadata from the Sveriges Riksbank SWEA API.

Usage

```
srb_series(type = "series")
```

Arguments

type	(character(1)) The type of metadata to query. One of "series" or "groups". Default "series".

Value

A data.table::data.table() with the requested metadata.

Source

https://developer.api.riksbank.se/

See Also

Other metadata: banxico_metadata(), bbk_dimension(), bbk_metadata(), bcb_currencies(), bdf_dimension(), bdp_dataset(), bdp_dimension(), bdp_domain(), bdp_series(), bis_dimension(), bis_metadata(), boi_dimension(), boi_metadata(), boj_metadata(), cnb_dimension(), cnb_indicators(), cnb_snapshots(), cnb_tree(), ecb_dimension(), ecb_metadata(), nbb_dimension(), nbb_metadata(), nob_dimension(), nob_metadata(), onb_dimension(), onb_frequency(), onb_hierarchy(), onb_metadata(), onb_toc(), snb_dimension(), snb_metadata(), snb_toc(), srb_calendar()

Examples

```
srb_series()
```

Index

∗ **data**
banxico_data, 3
bbk_data, 6
bbk_series, 10
bcb_data, 12
bcb_expectations, 13
bcb_fx_rates, 14
bcb_inflation, 15
bcb_selic, 17
bcb_top5, 18
bde_data, 19
bde_latest, 20
bdf_codelist, 21
bdf_data, 22
bdf_dataset, 23
bdp_data, 25
bis_data, 31
boc_data, 35
boe_data, 39
boi_data, 40
boj_data, 43
cnb_czeonia, 46
cnb_data, 47
cnb_fx_other_rates, 50
cnb_fx_rates, 51
cnb_pribor, 54
ecb_data, 57
nbb_data, 62
nbp_fx_rates, 65
nbp_gold, 66
nob_data, 67
onb_data, 71
snb_data, 77
srb_cross_rates, 82
srb_data, 83
∗ **metadata**
banxico_metadata, 5
bbk_dimension, 8
bbk_metadata, 9
bcb_currencies, 11
bdf_dimension, 24
bdp_dataset, 27
bdp_dimension, 28
bdp_domain, 29
bdp_series, 30
bis_dimension, 33
bis_metadata, 34
boi_dimension, 41
boi_metadata, 42
boj_metadata, 44
cnb_dimension, 49
cnb_indicators, 52
cnb_snapshots, 55
cnb_tree, 56
ecb_dimension, 59
ecb_metadata, 61
nbb_dimension, 63
nbb_metadata, 64
nob_dimension, 69
nob_metadata, 70
onb_dimension, 72
onb_frequency, 73
onb_hierarchy, 74
onb_metadata, 75
onb_toc, 76
snb_dimension, 78
snb_metadata, 79
snb_toc, 80
srb_calendar, 81
srb_series, 84

banxico_data, 3
banxico_data(), *7*, *10*, *12*, *14–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
banxico_metadata, 5
banxico_metadata(), *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80–82*, *85*

bbk_cache_clear *(*cache*)*, 45
bbk_cache_dir *(*cache*)*, 45
bbk_data, 6
bbk_data(), *4*, *10*, *12*, *14–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bbk_dimension, 8
bbk_dimension(), *5*, *9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80–82*, *85*
bbk_metadata, 9
bbk_metadata(), *5, 6*, *8*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80–82*, *85*
bbk_series, 10
bbk_series(), *4*, *7*, *12*, *14–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bcb_currencies, 11
bcb_currencies(), *5*, *8, 9*, *14*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80–82*, *85*
bcb_data, 12
bcb_data(), *4*, *7*, *10*, *14–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bcb_expectations, 13
bcb_expectations(), *4*, *7*, *10*, *12*, *15–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bcb_fx_rates, 14
bcb_fx_rates(), *4*, *7*, *10*, *12*, *14*, *16–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bcb_inflation, 15
bcb_inflation(), *4*, *7*, *10*, *12*, *14, 15*, *17, 18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bcb_selic, 17
bcb_selic(), *4*, *7*, *10*, *12*, *14–16*, *18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bcb_top5, 18
bcb_top5(), *4*, *7*, *10*, *12*, *14–17*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bde_data, 19
bde_data(), *4*, *7*, *10*, *12*, *14–18*, *21–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bde_latest, 20
bde_latest(), *4*, *7*, *10*, *12*, *14–18*, *20*, *22–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bdf_codelist, 21
bdf_codelist(), *4*, *7*, *10*, *12*, *14–18*, *20, 21*, *23, 24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bdf_data, 22
bdf_data(), *4*, *7*, *10*, *12*, *14–18*, *20–22*, *24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bdf_dataset, 23
bdf_dataset(), *4*, *7*, *10*, *12*, *14–18*, *20–23*, *25*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bdf_dimension, 24
bdf_dimension(), *5*, *8, 9*, *11*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80–82*, *85*
bdp_data, 25
bdp_data(), *4*, *7*, *10*, *12*, *14–18*, *20–24*, *30*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83, 84*
bdp_dataset, 27
bdp_dataset(), *5*, *8, 9*, *11*, *25*, *29–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80–82*, *85*
bdp_dimension, 28
bdp_dimension(), *5*, *8, 9*, *11*, *25*, *28*, *30, 31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80–82*, *85*
bdp_domain, 29
bdp_domain(), *5*, *8, 9*, *11*, *25–29*, *31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80–82*, *85*
bdp_series, 30
bdp_series(), *5*, *8, 9*, *11*, *25, 26*, *28–30*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80–82*, *85*
bis_data, 31

bis_data(), 4, 7, 10, 12, 14–18, 20–24, 27, 36, 39, 41, 44, 47, 48, 51, 52, 54, 58, 63, 66–68, 72, 77, 83, 84
bis_dimension, 33
bis_dimension(), 5, 8, 9, 11, 25, 28–31, 34, 42, 43, 45, 50, 53, 55, 57, 59, 61, 64, 65, 69, 70, 73–76, 78, 80–82, 85
bis_metadata, 34
bis_metadata(), 5, 8, 9, 11, 25, 28–31, 33, 42, 43, 45, 50, 53, 55, 57, 59, 61, 64, 65, 69, 70, 73–76, 78, 80–82, 85
boc_catalog, 35
boc_data, 35
boc_data(), 4, 7, 10, 12, 14–18, 20–24, 27, 32, 39, 41, 44, 47, 48, 51, 52, 54, 58, 63, 66–68, 72, 77, 83, 84
boc_fx_rates, 37
boc_metadata, 38
boe_data, 39
boe_data(), 4, 7, 10, 12, 14–18, 20–24, 27, 32, 36, 41, 44, 47, 48, 51, 52, 54, 58, 63, 66–68, 72, 77, 83, 84
boi_data, 40
boi_data(), 4, 7, 10, 12, 14–18, 20–24, 27, 32, 36, 39, 44, 47, 48, 51, 52, 54, 58, 63, 66–68, 72, 77, 83, 84
boi_dimension, 41
boi_dimension(), 5, 8, 9, 11, 25, 28–31, 33, 34, 43, 45, 50, 53, 55, 57, 59, 61, 64, 65, 69, 70, 73–76, 78, 80–82, 85
boi_metadata, 42
boi_metadata(), 5, 8, 9, 11, 25, 28–31, 33, 34, 40, 42, 45, 50, 53, 55, 57, 59, 61, 64, 65, 69, 70, 73–76, 78, 80–82, 85
boj_data, 43
boj_data(), 4, 7, 10, 12, 14–18, 20–24, 27, 32, 36, 39, 41, 47, 48, 51, 52, 54, 58, 63, 66–68, 72, 77, 83, 84
boj_metadata, 44
boj_metadata(), 5, 8, 9, 11, 25, 28–31, 33, 34, 42, 43, 50, 53, 55, 57, 59, 61, 64, 65, 69, 70, 73–76, 78, 80–82, 85

cache, 45
cnb_czeonia, 46
cnb_czeonia(), 4, 7, 10, 12, 14–18, 20–24, 27, 32, 36, 39, 41, 44, 48, 51, 52, 54, 58, 63, 66–68, 72, 77, 83, 84
cnb_data, 47
cnb_data(), 4, 7, 10, 12, 14–18, 20–24, 27, 32, 36, 39, 41, 44, 47, 51, 52, 54, 55, 58, 63, 66–68, 72, 77, 83, 84
cnb_dimension, 49
cnb_dimension(), 5, 8, 9, 11, 25, 28–31, 33, 34, 42, 43, 45, 53, 55, 57, 59, 61, 64, 65, 69, 70, 73–76, 78, 80–82, 85
cnb_fx_other_rates, 50
cnb_fx_other_rates(), 4, 7, 10, 12, 14–18, 20–24, 27, 32, 36, 39, 41, 44, 47, 48, 52, 54, 58, 63, 66–68, 72, 77, 83, 84
cnb_fx_rates, 51
cnb_fx_rates(), 4, 7, 10, 12, 14–18, 20–24, 27, 32, 36, 39, 41, 44, 47, 48, 50, 51, 54, 58, 63, 66–68, 72, 77, 83, 84
cnb_indicators, 52
cnb_indicators(), 5, 8, 9, 11, 25, 28–31, 33, 34, 42, 43, 45, 50, 55, 57, 59, 61, 64, 65, 69, 70, 73–76, 78, 80–82, 85
cnb_pribor, 54
cnb_pribor(), 4, 7, 10, 12, 14–18, 20–24, 27, 32, 36, 39, 41, 44, 47, 48, 51, 52, 58, 63, 66–68, 72, 77, 83, 84
cnb_snapshots, 55
cnb_snapshots(), 5, 8, 9, 11, 25, 28–31, 33, 34, 42, 43, 45, 48, 50, 53, 57, 59, 61, 64, 65, 69, 70, 73–76, 78, 80–82, 85
cnb_tree, 56
cnb_tree(), 5, 8, 9, 11, 25, 28–31, 33, 34, 42, 43, 45, 50, 53, 55, 59, 61, 64, 65, 69, 70, 73–76, 78, 80–82, 85

data.table::data.table(), 4, 5, 7–12, 14–21, 23–30, 32–39, 41, 42, 44, 46, 48, 50–56, 58–61, 63–70, 72–84

ecb_data, 57
ecb_data(), 4, 7, 10, 12, 14–18, 20–24, 27, 32, 36, 39, 41, 44, 47, 48, 51, 52, 54, 60, 63, 66–68, 72, 77, 83, 84
ecb_dimension, 59
ecb_dimension(), 5, 8, 9, 11, 25, 28–31, 33, 34, 42, 43, 45, 50, 53, 55, 57, 61, 64, 65, 69, 70, 73–76, 78, 80–82, 85
ecb_euro_rates (ecb_fx_rates), 60
ecb_fx_rates, 60
ecb_metadata, 61
ecb_metadata(), 5, 8, 9, 11, 25, 28–31, 33, 34, 42, 43, 45, 50, 53, 55, 57, 59, 64,

65, *69, 70*, *73–76*, *78*, *80–82*, *85*

modifyList(), *21, 22*, *24*

nbb_data, 62
nbb_data(), *4*, *7*, *10*, *12*, *14–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *66–68*, *72*, *77*, *83, 84*
nbb_dimension, 63
nbb_dimension(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *65*, *69, 70*, *73–76*, *78*, *80–82*, *85*
nbb_metadata, 64
nbb_metadata(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61, 62*, *64*, *69, 70*, *73–76*, *78*, *80–82*, *85*
nbp_fx_rates, 65
nbp_fx_rates(), *4*, *7*, *10*, *12*, *14–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *67, 68*, *72*, *77*, *83, 84*
nbp_gold, 66
nbp_gold(), *4*, *7*, *10*, *12*, *14–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66*, *68*, *72*, *77*, *83, 84*
nob_data, 67
nob_data(), *4*, *7*, *10*, *12*, *14–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66, 67*, *72*, *77*, *83, 84*
nob_dimension, 69
nob_dimension(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *70*, *73–76*, *78*, *80–82*, *85*
nob_metadata, 70
nob_metadata(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *68, 69*, *73–76*, *78*, *80–82*, *85*

onb_data, 71
onb_data(), *4*, *7*, *10*, *12*, *14–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *77*, *83, 84*
onb_dimension, 72
onb_dimension(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *74–76*, *78*, *80–82*, *85*
onb_frequency, 73
onb_frequency(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73*, *75, 76*, *78*, *80–82*, *85*
onb_hierarchy, 74
onb_hierarchy(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73, 74*, *76*, *78*, *80–82*, *85*
onb_metadata, 75
onb_metadata(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80–82*, *85*
onb_toc, 76
onb_toc(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80–82*, *85*

snb_data, 77
snb_data(), *4*, *7*, *10*, *12*, *14–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *83, 84*
snb_dimension, 78
snb_dimension(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *80–82*, *85*
snb_metadata, 79
snb_metadata(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *81, 82*, *85*
snb_toc, 80
snb_toc(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80*, *82*, *85*
srb_calendar, 81
srb_calendar(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80, 81*, *85*
srb_cross_rates, 82
srb_cross_rates(), *4*, *7*, *10*, *12*, *14–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *84*
srb_data, 83
srb_data(), *4*, *7*, *10*, *12*, *14–18*, *20–24*, *27*, *32*, *36*, *39*, *41*, *44*, *47, 48*, *51, 52*, *54*, *58*, *63*, *66–68*, *72*, *77*, *83*
srb_series, 84
srb_series(), *5*, *8, 9*, *11*, *25*, *28–31*, *33, 34*, *42, 43*, *45*, *50*, *53*, *55*, *57*, *59*, *61*, *64, 65*, *69, 70*, *73–76*, *78*, *80–83*