Package: rbcb (via r-universe)

August 20, 2024

Type Package

Title R Interface to Brazilian Central Bank Web Services

Version 0.1.14

Author Wilson Freitas <wilson.freitas@gmail.com>

Maintainer Wilson Freitas <wilson.freitas@gmail.com>

Description The Brazilian Central Bank API delivers many datasets which regard economic activity, regional economy, international economy, public finances, credit indicators and many more. For more information please see <http://dadosabertos.bcb.gov.br/>. These datasets can be accessed through 'rbcb' functions and can be obtained in different data structures common to R ('tibble', 'data.frame', 'xts', ...).

URL https://github.com/wilsonfreitas/rbcb, https://wilsonfreitas.github.io/rbcb/

BugReports https://github.com/wilsonfreitas/rbcb/issues

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports httr, jsonlite, tibble, xts, xml2, utils, stats, methods, purrr, digest

RoxygenNote 7.2.3

Suggests testthat (>= 3.0.0), miniUI, shiny, dplyr, covr, withr

Collate 'rbcb-package.R' 'rbcb_get.R' 'sgs.R' 'datasets.R' 'addins.R' 'rbcb_get_series.R' 'currency_helpers.R' 'utils.R' 'rbcb_get_currencies.R' 'odata-expectativas.R' 'odata_ptax.R'

Config/testthat/edition 3

Repository https://wilsonfreitas.r-universe.dev

RemoteUrl https://github.com/wilsonfreitas/rbcb

RemoteRef HEAD

RemoteSha 52e1602d11f45145db64db657c81e0b263f67fd4

Contents

`get_all_currencies` *All currency values*

Description

Gets all currency values

Usage

```
get_all_currencies(date)
```

Arguments

`date`	reference date

Value

A `data.frame` with all currency values from the given date. The currency rates come quoted in BRL.

Examples

```
## Not run:
get_all_currencies("2017-03-10")

## End(Not run)
```

`get_annual_market_expectations`

Get annual market expectations of economic indicators

Description

Statistics for the annual expectations of economic indicators. All statistics are computed based on annual expectations provided by many financial institutions in Brazil: banks, funds, risk managers, so on and so forth. These expections and its statistics are used to build the FOCUS Report weekly released by the Brazilian Central Bank.

Usage

```
get_annual_market_expectations(
  indic = NULL,
  start_date = NULL,
  end_date = NULL,
  ...
)
```

Arguments

`indic`	a character vector with economic indicators names. They are case sensitive and don't forget the accents.
`start_date`	series initial date. Accepts ISO character formated date and `Date`.
`end_date`	series final date. Accepts ISO character formated date and `Date`.
`...`	additional parameters to be passed to the API `indic` argument must be one of indicators listed in Details. Respecting the case, blank spaces and accents. The `...` is to be used with API's parameters. `$top` to specify the maximum number of rows to be returned, this returns the `$top` rows, in chronological order. There is also `$skip` to ignore the first rows.

Details

There are annual expectations available for the following indicators:

- Balança Comercial
- Câmbio

- Conta corrente
- Dívida bruta do governo geral
- Dívida líquida do setor público
- IGP-DI
- IGP-M
- INPC
- Investimento direto no país
- IPA-DI
- IPA-M
- IPCA
- IPCA Administrados
- IPCA Alimentação no domicílio
- IPCA Bens industrializados
- IPCA Livres
- IPCA Serviços
- IPCA-15
- IPC-FIPE
- PIB Agropecuária
- PIB Despesa de consumo da administração pública
- PIB despesa de consumo das famílias
- PIB Exportação de bens e serviços
- PIB Formação Bruta de Capital Fixo
- PIB Importação de bens e serviços
- PIB Indústria
- PIB Serviços
- PIB Total
- Produção industrial
- Resultado nominal
- Resultado primário
- Selic
- Taxa de desocupação

Check <https://olinda.bcb.gov.br/olinda/servico/Expectativas/versao/v1/documentacao#ExpectativasMercadoAnuais> for more details

Value

A `data.frame` with the requested data.

Examples

```
## Not run:
indic <- c("PIB Total", "Fiscal")
end_date <- "2018-01-31"
x <- get_annual_market_expectations(indic, end_date = end_date, `$top` = 10)

# return all indicators for the specified date range
start_date <- "2021-01-01"
x <- get_annual_market_expectations(start_date = start_date, `$top` = 20)

## End(Not run)
```

---

get_currency *Get currency values for a given period*

Description

Given a currency symbol and a time interval (in dates) this function returns the bid and ask time series of currency rates.

Usage

```
get_currency(
  symbol,
  start_date,
  end_date,
  as = c("tibble", "xts", "data.frame", "text")
)
```

Arguments

symbol	currency symbol
start_date	time interval initial date
end_date	time interval last date
as	the object's returning type The symbol argument is a three digits character which represents one currency. The symbols can be obtained with list_currencies. The time series date range is defined by start_date and end_date.

Value

The time series with the bid and ask currency rates regarding the given symbol quoted in BRL. The default returning is a tibble-fashioned data.frame with the three columns: date, ask and bid. The as argument also accepts data.frame to return old fashioned data frames, xts to return a xts object with two variables (bid and ask) and text which returns the text content download from BCB site.

Examples

```
## Not run:
get_currency("USD", "2017-03-01", "2017-03-10")

## End(Not run)
```

`get_currency_cross_rates`

Get currency matrix from BCB

Description

The currency matrix has the currency cross rates for all currencies present in the BCB system.

Usage

```
get_currency_cross_rates(date, ref = c("ask", "bid"))
```

Arguments

`date`	reference date
`ref`	reffers to `bid` or `ask` rates (default `ask`) `date` is the reference date by which the currency rates must be downloaded. `ref` defaults to `ask` and `bid` returns all currency cross rates calculated with bid rates.

Value

A square `matrix` with `colnames` and `rownames` filled with currency symbols The cells must be read as `ROW` in `COL`, for example, BRL (row) in USD (column) means Brazilian Reals in American Dollars.

Examples

```
## Not run:
x <- get_currency_cross_rates("2017-03-10")
currencies <- c("USD", "BRL", "AUD", "EUR", "CAD")
x[currencies, currencies]

## End(Not run)
```

`get_market_expectations` *Get market expectations*

Description

General function to get statistics of market expectations. The API provides requests for annual, monthly, and quarterly expectations. Is is also proveided expectations for 12 months ahead, specific requests for the top 5 indicators for annual and monthly expectations and data provided by financial institutions.

Usage

```
get_market_expectations(
  type = c("annual", "quarterly", "monthly", "inflation-12-months", "top5s-monthly",
    "top5s-annual", "selic", "top5s-selic"),
  indic = NULL,
  start_date = NULL,
  end_date = NULL,
  keep_names = TRUE,
  ...
)
```

Arguments

`type` a character with one of the following: `annual`, `quarterly` `monthly`, `inflation-12-months`, `top5s-monthly`, `top5s-annual`.

`indic` a character vector with economic indicators names. They are case sensitive and don't forget the accents.

`start_date` series initial date. Accepts ISO character formated date and `Date`.

`end_date` series final date. Accepts ISO character formated date and `Date`.

`keep_names` if `TRUE` keeps the column names returned by the API (in portuguese), if `FALSE` the columns are renamed to standardized names (in english).

`...` additional parameters to be passed to the API

`type` defines the API used to fetch data.

- `selic`: refers to the API *Expectativas de Mercado Selic - Estatísticas* for SELIC rate expectations
- `annual`: refers to the API *Expectativas de Mercado Anuais* for annual market expectations
- `quarterly`: refers to the API *Expectativas de Mercado Trimestrais* for quarterly market expectations
- `monthly`: refers to the API *Expectativas de Mercado Mensais* for monthly market expectations

- `inflation-12-months`: refers to the API *Expectativas de mercado para inflação nos próximos 12 meses* for market expectations of inflation indexes for the next 12 months.
- `top5s-selic`: refers to the API *Expectativas de Mercado Selic Top5* for SELIC rate expectations of top 5's
- `top5s-monthly`: refers to the API *Expectativas de mercado mensais para os indicadores do Top 5* for monthly market expectations of top 5's
- `top5s-annual`: refers to the API *Expectativas de mercado anuais para os indicadores do Top 5* for annual market expectations of top 5's

`indic` argument must be one of indicators listed in Details. Respecting the case, blank spaces and accents.

The `...` is to be used with API's parameters. `$top` to specify the maximum number of rows to be returned, this returns the `$top` rows, in chronological order. `$skip` can be used to ignore the first rows. If provided `$filter` applies filters according to <https://olinda.bcb.gov.br/olinda/servico/ajuda>.

Details

All statistics are computed based on expectations provided by many financial institutions in Brazil: banks, funds, risk managers, so on and so forth. These expections and its statistics are used to build the FOCUS Report weekly released by the Brazilian Central Bank.

There are market expectations available for the following indicators:

- Balança Comercial
- Câmbio
- Conta corrente
- Dívida bruta do governo geral
- Dívida líquida do setor público
- IGP-DI
- IGP-M
- INPC
- Investimento direto no país
- IPA-DI
- IPA-M
- IPCA
- IPCA Administrados
- IPCA Alimentação no domicílio
- IPCA Bens industrializados
- IPCA Livres
- IPCA Serviços
- IPCA-15
- IPC-FIPE

- PIB Agropecuária
- PIB Despesa de consumo da administração pública
- PIB despesa de consumo das famílias
- PIB Exportação de bens e serviços
- PIB Formação Bruta de Capital Fixo
- PIB Importação de bens e serviços
- PIB Indústria
- PIB Serviços
- PIB Total
- Produção industrial
- Resultado nominal
- Resultado primário
- Selic
- Taxa de desocupação

Check <https://olinda.bcb.gov.br/olinda/servico/Expectativas/versao/v1/documentacao> for more details

Value

A `data.frame` with the requested data.

Examples

```
## Not run:
indic <- c("IPCA", "Câmbio")
x <- get_market_expectations("annual", indic, `$top` = 10)

x <- get_market_expectations("monthly", "Selic", `$top` = 20)

# get monthly expectations for top 5 indicators since 2021
x <- get_market_expectations("top5s-monthly", start_date = "2021-01-01")

# get annual expectations for top 5 indicators since 2021
x <- get_market_expectations("top5s-annual", `$top` = 20)

# get all inflation expectations for 12 months ahead starting on 2021-01
x <- get_market_expectations("inflation-12-months", start_date = "2021-01-01")

# get all SELIC expectations informed by financial institutions since 2022
x <- get_market_expectations("selic", start_date = "2022-01-01")

# get TOP5 SELIC expectations starting on 2022
x <- get_market_expectations("top5s-selic", start_date = "2022-01-01")

## End(Not run)
```

`get_monthly_market_expectations`

Get monthly market expectations of economic indicators

Description

Statistics for the monthly expectations of economic indicators. All statistics are computed based on monthly expectations provided by many financial institutions in Brazil: banks, funds, risk managers, so on and so forth. These expections and its statistics are used to build the FOCUS Report weekly released by the Brazilian Central Bank.

Usage

```
get_monthly_market_expectations(
  indic = NULL,
  start_date = NULL,
  end_date = NULL,
  ...
)
```

Arguments

`indic` a character vector with economic indicators names. They are case sensitive and don't forget the accents.

`start_date` series initial date. Accepts ISO character formated date and `Date`.

`end_date` series final date. Accepts ISO character formated date and `Date`.

`...` additional parameters to be passed to the API

`indic` argument must be one of indicators listed in Details. Respecting the case, blank spaces and accents.

The `...` is to be used with API's parameters. `$top` to specify the maximum number of rows to be returned, this returns the `$top` rows, in chronological order. There is also `$skip` to ignore the first rows.

Details

There are monthly expectations available for the following indicators:

- Câmbio
- IGP-DI
- IGP-M
- INPC
- IPA-DI
- IPA-M
- IPCA

- IPCA Administrados
- IPCA Alimentação no domicílio
- IPCA Bens industrializados
- IPCA Livres
- IPCA Serviços
- IPCA-15
- IPC-Fipe
- Produção industrial
- Selic
- Taxa de desocupação

Check <https://olinda.bcb.gov.br/olinda/servico/Expectativas/versao/v1/documentacao#ExpectativaMercadoMensais> for more details

Value

A `data.frame` with the requested data.

Examples

```
## Not run:
indic <- c("IPCA", "IPC-Fipe")
end_date <- "2018-01-31"
x <- get_monthly_market_expectations(indic, end_date = end_date, `$top` = 10)

# return all indicators for the specified date range
start_date <- "2021-01-01"
x <- get_monthly_market_expectations(start_date = start_date, `$top` = 20)

## End(Not run)
```

`get_quarterly_market_expectations`

Get quarterly market expectations of economic indicators

Description

Statistics for the quarterly expectations of economic indicators. All statistics are computed based on quarterly expectations provided by many financial institutions in Brazil: banks, funds, risk managers, so on and so forth. These expections and its statistics are used to build the FOCUS Report weekly released by the Brazilian Central Bank.

Usage

```
get_quarterly_market_expectations(
  indic = NULL,
  start_date = NULL,
  end_date = NULL,
  ...
)
```

Arguments

`indic` a character vector with economic indicators names. They are case sensitive and don't forget the accents.

`start_date` series initial date. Accepts ISO character formated date and `Date`.

`end_date` series final date. Accepts ISO character formated date and `Date`.

`...` additional parameters to be passed to the API

`indic` argument must be one of indicators listed in Details. Respecting the case, blank spaces and accents.

The `...` is to be used with API's parameters. `$top` to specify the maximum number of rows to be returned, this returns the `$top` rows, in chronological order. There is also `$skip` to ignore the first rows.

Details

There are quarterly expectations available for the following indicators:

- Câmbio
- IPCA
- IPCA Administrados
- IPCA Alimentação no domicílio
- IPCA Bens industrializados
- IPCA Livres
- IPCA Serviços
- PIB Agropecuária
- PIB Indústria
- PIB Serviços
- PIB Total
- Taxa de desocupação

Check <https://olinda.bcb.gov.br/olinda/servico/Expectativas/versao/v1/documentacao#ExpectativasMercadoTrimestrais> for more details

Value

A `data.frame` with the requested data.

Examples

```
## Not run:
indic <- c("PIB Industrial", "PIB Total")
end_date <- "2018-01-31"
x <- get_quarterly_market_expectations(indic, end_date = end_date, `$top` = 10)

# return all indicators for the specified date range
start_date <- "2021-01-01"
x <- get_quarterly_market_expectations(start_date = start_date, `$top` = 20)

## End(Not run)
```

---

get_selic_market_expectations

Get Selic market expectations

Description

Statistics of Selic market expectations. All statistics are computed based on expectations provided by many financial institutions in Brazil: banks, funds, risk managers, so on and so forth. These expections and its statistics are used to build the FOCUS Report weekly released by the Brazilian Central Bank.

Usage

```
get_selic_market_expectations(start_date = NULL, end_date = NULL, ...)
```

Arguments

`start_date`	series initial date. Accepts ISO character formated date and `Date`.
`end_date`	series final date. Accepts ISO character formated date and `Date`.
`...`	additional parameters to be passed to the API The `...` is to be used with API's parameters. `$top` to specify the maximum number of rows to be returned, this returns the `$top` rows, in chronological order. There is also `$skip` to ignore the first rows.

Details

Check <https://olinda.bcb.gov.br/olinda/servico/Expectativas/versao/v1/documentacao#ExpectativasMercadoSelic> for more details

Value

A `data.frame` with the requested data.

Examples

```
## Not run:
# return all indicators for the specified date range
start_date <- "2022-01-01"
x <- get_selic_market_expectations(start_date = start_date, `$top` = 20)

## End(Not run)
```

`get_series` *Get the series from BCB*

Description

Get the series from BCB

Usage

```
get_series(
  code,
  start_date = NULL,
  end_date = NULL,
  last = 0,
  as = c("tibble", "xts", "ts", "data.frame", "text")
)
```

Arguments

`code` series code

`start_date` series initial date. Accepts ISO character formated date and `Date`.

`end_date` series final date. Accepts ISO character formated date and `Date`.

`last` last items of the series

`as` the returning type: data objects (`tibble`, `xts`, `data.frame`, `ts`) or `text` for raw JSON

`code` argument can be obtained in the SGS system site. In this site searches can be executed in order to find out the desired series and use the series code in the `code` argument.

The arguments `start_date`, `end_date` and `last` are optional. If none of these arguments are set, then the entire time series is downloaded. Define `start_date` and `end_date` to download a period of data and to download the last `N` registers define the `last` argument to `N` a positive integer. Once `last` is provided it overrides the arguments `start_date` and `end_date`.

Value

`tibble` is the default returning class, but the argument `as` can be set to `xts`, `data.frame`, `ts`, or `text` to return these other types. `text` returns the JSON data provided by the remote API.

Examples

```
# download the entire series
## Not run:
get_series(1)

## End(Not run)
# download a period of dates
## Not run:
get_series(1, start_date = "2016-12-01")

## End(Not run)
## Not run:
x <- get_series(1, start_date = "2016-12-01", end_date = "2016-12-31")

## End(Not run)
# downlaod the last register
## Not run:
x <- get_series(1, last = 1)

## End(Not run)
```

---

get_top5s_annual_market_expectations

Get annual market expectations from top 5 providers

Description

Statistics of annual expectations for top 5 indicators. All statistics are computed based on expectations provided by many financial institutions in Brazil: banks, funds, risk managers, so on and so forth. These expections and its statistics are used to build the FOCUS Report weekly released by the Brazilian Central Bank.

Usage

```
get_top5s_annual_market_expectations(
  indic = NULL,
  start_date = NULL,
  end_date = NULL,
  ...
)
```

Arguments

`indic`	a character vector with economic indicators names. They are case sensitive and don't forget the accents.
`start_date`	series initial date. Accepts ISO character formated date and `Date`.
`end_date`	series final date. Accepts ISO character formated date and `Date`.

`...` additional parameters to be passed to the API

`indic` argument must be one of indicators listed in Details. Respecting the case, blank spaces and accents.

The `...` is to be used with API's parameters. `$top` to specify the maximum number of rows to be returned, this returns the `$top` rows, in chronological order. There is also `$skip` to ignore the first rows.

Details

There are inflation's expectations available for the following indicators:

- Câmbio
- IGP-DI
- IGP-M
- IPCA
- Selic

Check <https://olinda.bcb.gov.br/olinda/servico/Expectativas/versao/v1/documentacao#ExpectativasMercadoTop5Anuais> for more details

Value

A `data.frame` with the requested data.

Examples

```
## Not run:
indic <- "IPCA"
end_date <- "2018-06-22"
x <- get_top5s_annual_market_expectations(indic, end_date = end_date, `$top` = 10)

# return all indicators for the specified date range
start_date <- "2021-01-01"
x <- get_top5s_annual_market_expectations(start_date = start_date, `$top` = 20)

## End(Not run)
```

get_top5s_monthly_market_expectations

Get monthly market expectations from top 5 providers

Description

Statistics of monthly expectations for top 5 indicators. All statistics are computed based on expectations provided by many financial institutions in Brazil: banks, funds, risk managers, so on and so forth. These expections and its statistics are used to build the FOCUS Report weekly released by the Brazilian Central Bank.

Usage

```
get_top5s_monthly_market_expectations(
  indic = NULL,
  start_date = NULL,
  end_date = NULL,
  ...
)
```

Arguments

`indic` a character vector with economic indicators names. They are case sensitive and don't forget the accents.

`start_date` series initial date. Accepts ISO character formated date and `Date`.

`end_date` series final date. Accepts ISO character formated date and `Date`.

`...` additional parameters to be passed to the API

`indic` argument must be one of indicators listed in Details. Respecting the case, blank spaces and accents.

The `...` is to be used with API's parameters. `$top` to specify the maximum number of rows to be returned, this returns the `$top` rows, in chronological order. There is also `$skip` to ignore the first rows.

Details

There are inflation's expectations available for the following indicators:

- Câmbio
- IGP-DI
- IGP-M
- IPCA
- Selic

Check <https://olinda.bcb.gov.br/olinda/servico/Expectativas/versao/v1/documentacao#ExpectativasMercadoTop5Mensais> for more details

Value

A `data.frame` with the requested data.

Examples

```
## Not run:
indic <- "IPCA"
end_date <- "2018-06-22"
x <- get_top5s_monthly_market_expectations(indic, end_date = end_date, `$top` = 10)

# return all indicators for the specified date range
start_date <- "2021-01-01"
```

```
x <- get_top5s_monthly_market_expectations(start_date = start_date, `$top` = 20)

## End(Not run)
```

get_top5s_selic_market_expectations

Get Selic market expectations from top 5 providers

Description

Statistics of Selic expectations for top 5 indicators. All statistics are computed based on expectations provided by many financial institutions in Brazil: banks, funds, risk managers, so on and so forth. These expections and its statistics are used to build the FOCUS Report weekly released by the Brazilian Central Bank.

Usage

```
get_top5s_selic_market_expectations(start_date = NULL, end_date = NULL, ...)
```

Arguments

`start_date`	series initial date. Accepts ISO character formated date and `Date`.
`end_date`	series final date. Accepts ISO character formated date and `Date`.
`...`	additional parameters to be passed to the API The `...` is to be used with API's parameters. `$top` to specify the maximum number of rows to be returned, this returns the `$top` rows, in chronological order. There is also `$skip` to ignore the first rows.

Details

Check <https://olinda.bcb.gov.br/olinda/servico/Expectativas/versao/v1/documentacao#ExpectativasMercadoTop5Selic> for more details

Value

A `data.frame` with the requested data.

Examples

```
## Not run:
# return all indicators for the specified date range
start_date <- "2022-01-01"
x <- get_top5s_selic_market_expectations(start_date = start_date, `$top` = 20)

## End(Not run)
```

get_twelve_months_inflation_expectations

Get inflation's market expectations for the next 12 months

Description

Statistics of inflation's market expectations for the next 12 months. All statistics are computed based on expectations provided by many financial institutions in Brazil: banks, funds, risk managers, so on and so forth. These expections and its statistics are used to build the FOCUS Report weekly released by the Brazilian Central Bank.

Usage

```
get_twelve_months_inflation_expectations(
  indic = NULL,
  start_date = NULL,
  end_date = NULL,
  ...
)
```

Arguments

`indic`	a character vector with economic indicators names. They are case sensitive and don't forget the accents.
`start_date`	series initial date. Accepts ISO character formated date and `Date`.
`end_date`	series final date. Accepts ISO character formated date and `Date`.
`...`	additional parameters to be passed to the API

`indic` argument must be one of indicators listed in Details. Respecting the case, blank spaces and accents.

The `...` is to be used with API's parameters. `$top` to specify the maximum number of rows to be returned, this returns the `$top` rows, in chronological order. There is also `$skip` to ignore the first rows.

Details

There are inflation's expectations available for the following indicators:

- IGP-DI
- IGP-M
- INPC
- IPA-DI
- IPA-M
- IPCA
- IPCA Administrados

- IPCA Alimentação no domicílio
- IPCA Bens industrializados
- IPCA Livres
- IPCA Serviços
- IPCA-15
- IPC-FIPE

Check <https://olinda.bcb.gov.br/olinda/servico/Expectativas/versao/v1/documentacao#ExpectativasMercadoInflacao12Mes for more details

Value

A `data.frame` with the requested data.

Examples

```
## Not run:
indic <- c("IPCA", "IGP-M")
end_date <- "2018-06-22"
x <- get_twelve_months_inflation_expectations(indic, end_date = end_date, `$top` = 10)

# return all indicators for the specified date range
start_date <- "2021-01-01"
x <- get_twelve_months_inflation_expectations(start_date = start_date, `$top` = 20)

## End(Not run)
```

helpers *Helpers to access time series columns*

Description

Extraxt bid and ask data from time series objects.

Usage

```
Bid(x, ...)

## S3 method for class 'data.frame'
Bid(x, ...)

## S3 method for class 'xts'
Bid(x, ...)

## S3 method for class 'olinda_df'
Bid(x, ...)
```

```
Ask(x, ...)

## S3 method for class 'data.frame'
Ask(x, ...)

## S3 method for class 'xts'
Ask(x, ...)

## S3 method for class 'olinda_df'
Ask(x, ...)
```

Arguments

`x`	time series objects containing data to be extracted
`...`	additional arguments

Value

tibble with time series

Examples

```
## Not run:
x <- get_currency("EUR", "2018-06-22", "2018-06-28")
Bid(x)
Ask(x)

## End(Not run)
```

`list_currencies` *List all currencies*

Description

Lists all currencies and presents their name, symbol, numeric code, country name and county numeric code

Usage

```
list_currencies()
```

Value

A `data.frame` with information of all currencies

Examples

```
## Not run:
list_currencies()

## End(Not run)
```

`olinda_get_currency` *Get currency values for a given period*

Description

Given a currency symbol and a time interval (in dates) this function returns the bid and ask time series of currency rates.

Usage

```
olinda_get_currency(
  symbol,
  start_date,
  end_date = NULL,
  as = c("tibble", "xts", "data.frame", "text"),
  parity = FALSE
)
```

Arguments

`symbol`	currency symbol
`start_date`	time interval initial date
`end_date`	time interval last date
`as`	the object's returning type
`parity`	`TRUE` returns the parity quotation (default `FALSE` currency quoted in BRL)

The `symbol` argument is a three digits character which represents one currency. The symbols can be obtained with `list_currencies`.

The time series date range is defined by `start_date` and `end_date`. If `end_date` is not passed, it is set equals to `start_date`.

The `parity` argument defaults to `FALSE`, which means that the returned data is quoted in BRL. If it is `TRUE` the returned data is quoted in USD, for type A currencies and for type B currencies it is quoted as 1 USD in CURRENCY. For example, AUD, which is type B, returns 1 USD in AUD.

Value

The time series with the bid and ask currency rates regarding the given symbol quoted in BRL. The default returning is a `tibble`-fashioned `data.frame` with the three columns: `date`, `ask` and `bid`. The `as` argument also accepts `data.frame` to return old fashioned data frames, `xts` to return a xts object with two variables (bid and ask) and `text` which returns the text content download from BCB site.

Examples

```
## Not run:
olinda_get_currency("USD", "2017-03-01", "2017-03-10")

## End(Not run)
```

olinda_list_currencies *List all currencies*

Description

Lists all currencies and presents their name, symbol, numeric code, country name and county numeric code

Usage

```
olinda_list_currencies()
```

Value

A `data.frame` with information of all currencies.

The `currency_type` refers to the currency's parity quotation. Parity quotations relates currency values with USD.

Examples

```
## Not run:
list_currencies()

## End(Not run)
```

rbcb-options *rbcb options*

Description

Options used in rbcb inside some of its functions.

Details

* ‘rbcb_cache‘: all downloaded data is stored in temporary directories, if ‘rbcb_cache‘ is FALSE downloaded data overwrites files if it already exists. Otherwise, download is not executed and the existing file is returned. Defaults to TRUE. * ‘rbcb_verbose‘: if TRUE verbose messages are displayed when http requests are executed with httr. Defaults to FALSE.

Examples

```
## Not run:
options(rbcb_cache = FALSE)
options(rbcb_verbose = TRUE)

## End(Not run)
```

---

rbcb_dataset *rbcb dataset*

Description

‘rbcb_dataset(name)‘ opens an [RStudio gadget](https://shiny.rstudio.com/articles/gadgets.html) and [addin](http://rstudio.github.io/rstudioaddins/) that allows you to view a few attributes that help to explain the desired data.

Usage

```
rbcb_dataset(name)
```

Arguments

name	dataset name

Value

Addin has no return

---

rbcb_get *Gets data from BCB open data services*

Description

Gets SGS, currency, market expectations and many other datasets from the Brazilian Central Bank open data services.

Usage

```
rbcb_get(x, ...)

## S3 method for class 'sgs'
rbcb_get(x, from = NULL, to = NULL, last = 0, ...)
```

Arguments

`x`	an object that represents the kind of data to be downloaded
`...`	others arguments
`from`	series initial date. Accepts ISO character formated date and `Date`.
`to`	series final date. Accepts ISO character formated date and `Date`.
`last`	last items of the series To use the SGS API a ‘sgs‘ object should be passed.

Value

a dataset with the corresponding data (usually a ‘tibble‘)

Examples

```
## Not run:
x <- sgs(USD = 1, SELIC = 1178)
rbcb_get(x, from = Sys.Date() - 10)

## End(Not run)
```

`rbcb_search` *rbcb Search*

Description

‘rbcb_search(text)‘ opens an [RStudio gadget](https://shiny.rstudio.com/articles/gadgets.html) and [addin](http://rstudio.github.io/rstudioaddins/) that allows you to query for specific terms and see a suitable rbcb command to fetch the desired data.

Usage

```
rbcb_search(text = "")
```

Arguments

`text`	text to search

Value

Addin has no return

sgs *Create SGS code*

Description

SGS code is an objects that represents the SGS code used to download datasets from the SGS API.

Usage

```
sgs(..., load_info = TRUE)
```

Arguments

`...`	numeric codes (preferably named)
`load_info`	‘logical‘ indicating with the dataset info shoud be loaded (default TRUE)

Value

an SGS object representing SGS codes

Examples

```
## Not run:
sgs(USD = 1, IPCA = 433)

## End(Not run)
```

sgs_untidy *Convert tidy dataframe into a list*

Description

Convert a tidy dataframe into a list of separate time series objects like ‘xts‘ and ‘ts‘ or even a ‘tibble‘ with a time series representation (two columns with dates and values).

Usage

```
sgs_untidy(x, tidydf, as = c("tibble", "xts", "ts"))
```

Arguments

`x`	sgs object with the corresponding codes
`tidydf`	tidy dataframe returned by ‘rbcb_get‘
`as`	the returning type (‘tibble‘, ‘xts‘, ‘ts‘)

Value

a list with time series objects

Examples

```
## Not run:
x <- sgs(USD = 1, SELIC = 1178)
df <- rbcb_get(x, from = Sys.Date() - 10)
sgs_untidy(x, df, as = "xts")

## End(Not run)
```

Index