

# Package: moexer (via r-universe)

October 23, 2024

**Type** Package

**Title** Interact with Moscow Exchange Informational and Statistical Server (ISS')

**Version** 0.3.0.9000

**Maintainer** Dmitry Zotikov <dmitry.zotikov@ungrund.org>

**Description** This is a thin wrapper around the 'MOEX' ISS' REST interface, see <<https://iss.moex.com>>. It allows to quickly fetch price candles for a particular security, obtain its profile information and so on.

**License** GPL-3

**Language** en-GB

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**Depends** R (>= 4.2.0)

**Imports** glue, purrr, httr, jsonlite, tibble, readr, rlang, stringr, assertthat, cli, dplyr, lubridate, ggplot2

**Roxygen** list(markdown = TRUE)

**Suggests** testthat, knitr, rmarkdown

**URL** <https://github.com/x1o/moexer>

**BugReports** <https://github.com/x1o/moexer/issues>

**Repository** <https://x1o.r-universe.dev>

**RemoteUrl** <https://github.com/x1o/moexer>

**RemoteRef** HEAD

**RemoteSha** ae45146be2d03763cb099b1d08f88e5d2983f5f7

## Contents

fetching_fully . . . . .	2
following_cursor . . . . .	3

get_candles . . . . .	3
get_candle_borders . . . . .	5
get_candle_durations . . . . .	5
get_security_info . . . . .	6
plot.MoexCandles . . . . .	7
query_iss . . . . .	7
search_security . . . . .	8
<b>Index</b>	<b>9</b>

---

fetching_fully	<i>A Decorator for Fetching All Available Data</i>
----------------	----------------------------------------------------

---

## Description

A decorator for [query\\_iss](#) to increase the start parameter in a loop until no more data is received.

## Usage

```
fetching_fully(query_iss_fn)
```

## Arguments

query\_iss\_fn    A function object corresponding to [query\\_iss](#).

## Value

Decorated query\_iss\_fn.

## Examples

```
## Not run:
fetching_fully(query_iss)(
    'engines/stock/markets/shares/boards/TQBR/securities/SBER/candles',
    params = list(
        from = URLEncode('2020-01-10 10:00:00'),
        till = URLEncode('2020-01-10 23:59:59'),
        interval = 1,
        start = 10
    )
)

## End(Not run)
```

---

following_cursor	<i>A Decorator for Following Response Cursor</i>
------------------	--------------------------------------------------

---

**Description**

Iterative queries will be issued to fetch all section pages as indicated by <section>.cursor; the cursor section itself will be removed from the response.

**Usage**

```
following_cursor(query_iss_fn)
```

**Arguments**

query\_iss\_fn    A function object corresponding to [query\\_iss](#).

**Value**

Decorated query\_iss\_fn.

**Examples**

```
## Not run:
following_cursor(query_iss)(
    'history/engines/stock/markets/shares/securities/MOEX',
    params = list(
        from = '2021-09-01',
        till = '2021-12-31',
        start = 10
    )
)

## End(Not run)
```

---

get_candles	<i>Get Candles for a Set of Securities</i>
-------------	--------------------------------------------

---

**Description**

REST path: /engines/[engine]/markets/[market]/boards/[board]/securities/[security]/candles  
(see <http://iss.moex.com/iss/reference/46>).

**Usage**

```
get_candles(secid, from, till = NULL, interval = "monthly", ...)
```

**Arguments**

secid	A vector of security ID's.
from	A date or a datetime object, or something that can be coerced to it with <a href="#">readr::parse_date</a> or <a href="#">readr::parse_datetime</a> .
till	A date or a datetime object, or something that can be coerced to it with <a href="#">readr::parse_date</a> or <a href="#">readr::parse_datetime</a> , or NULL. If NULL, interpreted by ISS as "today".
interval	A character value specifying the candle duration (see <code>moexer.candle.intervals</code> option.)
...	Further arguments to <a href="#">query_iss</a> .

**Details**

To get the engine-market-board path a separate [get\\_security\\_info](#) query is made and the board with `is_primary = 1` is selected.

All candles for the specified period will be fetched, see [fetching\\_fully](#).

**Value**

A tibble as with candles in OHLCV format, prepended with a column containing the corresponding security ID.

**Examples**

```
## Not run:
# Get daily candles for `SBER`, `FXGD` from 2020-01-01 until today. Note that
# an unknown symbol `XXXX` is skipped with a warning.
get_candles(
  secid = c('XXXX', 'SBER', 'FXGD'),
  from = '2020-01-01',
  debug_output = TRUE
)

# Get SBER minute candles for one trading day (all 526 of them)
get_candles(
  secid = 'SBER',
  from = '2020-01-10',
  till = '2020-01-10',
  interval = 'per_minute'
)

# Get SBER minute candles for the specified time period (1 hour)
get_candles(
  secid = 'SBER',
  from = '2020-01-10 11:00:00',
  till = '2020-01-10 12:00:00',
  interval = 'per_minute'
)

## End(Not run)
```

---

get\_candle\_borders     *Get Possible Candle from-till Values for a Security*

---

### Description

REST path: /engines/[engine]/markets/[market]/boards/[board]/securities/[security]/candleborders (see <http://iss.moex.com/iss/reference/48>).

### Usage

```
get_candle_borders(secid, ...)
```

### Arguments

secid	A vector of security ID's.
...	Further arguments to <a href="#">query_iss</a> .

### Details

To get the engine-market-board path a separate [get\\_security\\_info](#) query is made and the board with is\_primary = 1 is selected.

### Value

A tibble with possible from-till values for each interval; additionally the intervals-durations mapping tibble is joined.

### Examples

```
## Not run:  
get_candle_borders(secid = c('SBER', 'FXGD'))  
  
## End(Not run)
```

---

get\_candle\_durations     *Get Candle Durations-Intervals Mapping*

---

### Description

REST path: /index?iss.only=durations (see <http://iss.moex.com/iss/reference/28>)

### Usage

```
get_candle_durations(...)
```

**Arguments**

... Further arguments to [query\\_iss](#).

**Value**

A tibble with the durations-intervals mapping.

**Examples**

```
## Not run:
get_candle_durations()

## End(Not run)
```

---

get\_security\_info      *Get a Security's Info*

---

**Description**

REST path: securities/[security] (see <http://iss.moex.com/iss/reference/13>)

**Usage**

```
get_security_info(secid, ...)
```

**Arguments**

secid                  A security ID.  
 ...                    Further arguments to [query\\_iss](#)

**Value**

A list with two tibbles:

- description: Full instrument profile incl. its type, listing level, etc.
- boards: The security's boards with boardid, history\_from, ... columns.

**Examples**

```
## Not run:
get_security_info(secid = 'SBER')

## End(Not run)
```

---

plot.MoexCandles	<i>Plot Candles</i>
------------------	---------------------

---

**Description**

Plot Candles

**Usage**

```
## S3 method for class 'MoexCandles'
plot(x, ...)
```

**Arguments**

x	A candles tibble as returned by <code>get_candles()</code>
...	For compatibility with the generic; ignored

**Value**

A ggplot2 object.

**Examples**

```
## Not run:
get_candles(secid = 'SBER', from = '2020-01-01') |>
  plot()

## End(Not run)
```

---

query_iss	<i>Submit a REST Query to ISS</i>
-----------	-----------------------------------

---

**Description**

See <http://iss.moex.com/iss/reference/> for the list of available endpoints.

**Usage**

```
query_iss(rest_path, params = list(), debug_output = getOption("moexer.debug"))
```

**Arguments**

rest_path	A REST path appended to <code>iss_base_url</code> .
params	A HTTP GET query parameters string, passed as a <code>list</code> (not URL-escaped!)
debug_output	Print REST URLs as they are queried.

**Value**

A list of tibbles, corresponding to the blocks in the response.

**Examples**

```
## Not run:
query_iss(
  rest_path = 'securities/SBER',
  params = list(iss.only = 'description'),
  debug_output = TRUE
)

## End(Not run)
```

---

search_security	<i>Search for a Single Security</i>
-----------------	-------------------------------------

---

**Description**

REST path: '/securities' (See <http://iss.moex.com/iss/reference/5>)

**Usage**

```
search_security(query, ...)
```

**Arguments**

query	(A part of the) ID, name, ISIN, issuer ID, reg. number of a security.
...	Further arguments to <a href="#">query_iss</a>

**Value**

A tibble with a list of matched securities.

**Examples**

```
## Not run:
search_security(query = 'SBER')

## End(Not run)
```



# Index

fetching\_fully, [2](#), [4](#)  
following\_cursor, [3](#)  
  
get\_candle\_borders, [5](#)  
get\_candle\_durations, [5](#)  
get\_candles, [3](#)  
get\_candles(), [7](#)  
get\_security\_info, [4](#), [5](#), [6](#)  
  
plot.MoexCandles, [7](#)  
  
query\_iss, [2-6](#), [7](#), [8](#)  
  
readr::parse\_date, [4](#)  
readr::parse\_datetime, [4](#)  
  
search\_security, [8](#)