

# Package ‘imf.data’

July 17, 2024

**Title** An Interface to IMF (International Monetary Fund) Data JSON API

**Version** 0.1.6

**Description** A straightforward interface for accessing the IMF (International Monetary Fund) data JSON API, available at <<https://data.imf.org/>>. This package offers direct access to the primary API endpoints: Dataflow, DataStructure, and CompactData. And, it provides an intuitive interface for exploring available dimensions and attributes, as well as querying individual time-series datasets. Additionally, the package implements a rate limit on API calls to reduce the chances of exceeding service limits (limited to 10 calls every 5 seconds) and encountering response errors.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**Imports** methods, utils, stats, curl, jsonlite

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**URL** <https://pedrobtz.github.io/imf.data/>

**BugReports** <https://github.com/pedrobtz/imf.data/issues>

**NeedsCompilation** no

**Author** Pedro Baltazar [aut, cre]

**Maintainer** Pedro Baltazar <pedrobtz@gmail.com>

**Repository** CRAN

**Date/Publication** 2024-07-16 22:10:11 UTC

## Contents

list_datasets . . . . .	2
load_datasets . . . . .	2
mt_compact_data . . . . .	3
mt_dataflow . . . . .	4
mt_data_structure . . . . .	4

**Index****5**

---

list_datasets	<i>List all available Time Series Datasets</i>
---------------	--

---

**Description**

List all available Time Series Datasets

**Usage**

```
list_datasets()
```

**Value**

a data.frame with columns 'Id' and 'Description'.

**Examples**

```
d <- list_datasets()
head(d)
```

---

load_datasets	<i>Loads Time Series Datasets</i>
---------------	-----------------------------------

---

**Description**

Loads Time Series Datasets

**Usage**

```
load_datasets(id, use_cache = TRUE)
```

**Arguments**

id	is character vector of Dataset Series identifier.
use_cache	is Boolean, defaults to TRUE, if TRUE it reloads the Dataset from cached values.

**Value**

a dataset object for the time-series identifier, i.e. a list containing a list 'dimensions' of data.frames with the valid dimensions values, and a function 'get\_series' to retrieve time-series data.

## Examples

```
DOT <- load_datasets("DOT")

DOT$get_series(freq = "M",
               ref_area = "GB",
               indicator = "TMG_CIF_USD",
               counterpart_area = c("B0", "W00"),
               start_period = "2022-01-01",
               end_period = "2022-12-31")
```

---

mt_compact_data	<i>Calls API endpoint 'CompactData' to get Dataset time series</i>
-----------------	--

---

## Description

Calls API endpoint 'CompactData' to get Dataset time series

## Usage

```
mt_compact_data(id, dimensions, start_period = NA, end_period = NA)
```

## Arguments

id	is a Dataset Series identifier.
dimensions	is a named list with the dimensions query.
start_period	is time series start date with formats.
end_period	is time series end date with format. See Details for the valid formats.

## Details

The 'start\_period' and 'end\_period' parameters should have formats: 'yyyy', 'yyyy-mm' or 'yyyy-mm-dd'.

## Value

a list with content of 'CompactData' response.

## Examples

```
DOT <- mt_compact_data("DOT", list("M", "GB", "TMG_CIF_USD", c("B0", "W00")))
```

---

mt_dataflow	<i>Calls API endpoint 'Dataflow'</i>
-------------	--------------------------------------

---

**Description**

Calls API endpoint 'Dataflow'

**Usage**

```
mt_dataflow()
```

**Value**

a list with content of 'Dataflow' response.

**Examples**

```
DF <- mt_dataflow()
```

---

mt_data_structure	<i>Calls API endpoint 'DataStructure'</i>
-------------------	---

---

**Description**

Calls API endpoint 'DataStructure'

**Usage**

```
mt_data_structure(id)
```

**Arguments**

`id` is a Dataset Series identifier.

**Value**

a list with content of 'DataStructure' response.

**Examples**

```
DOT <- mt_data_structure("DOT")
```

# Index

`list_datasets`, [2](#)  
`load_datasets`, [2](#)

`mt_compact_data`, [3](#)  
`mt_data_structure`, [4](#)  
`mt_dataflow`, [4](#)