

Package ‘statsDK’

September 30, 2017

Type Package

Title A Wrapper for the API of Statistics Denmark

Version 0.1.1

Author Mikkel Freltoft Krogsholm

Maintainer Mikkel Freltoft Krogsholm <mikkel@56n.dk>

Description Makes it possible through simple functions and commands to get and organize official danish statistics within a wide range of subjects. Visit the <<http://statbank.dk/>> and <<http://api.statbank.dk/console>> for further exploration of Statistics Denmark data.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports jsonlite, tibble, dplyr, tidyr, utils, stringr, ggplot2, httr, readr, lubridate

Depends R (>= 2.10)

RoxygenNote 6.0.1

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

URL <https://github.com/mikkelkrogsholm/statsDK>

BugReports <https://github.com/mikkelkrogsholm/statsDK/issues>

NeedsCompilation no

Repository CRAN

Date/Publication 2017-09-30 08:30:06 UTC

R topics documented:

fix_time	2
get_variables	2
retrieve_data	3

retrieve_metadata	4
retrieve_subjects	5
retrieve_tables	5

Index	7
--------------	----------

fix_time	<i>Fixes the time column of a data set</i>
----------	--------------------------------------------

Description

Statistics Denmark data often come with times/dates that are formatted to fit months or quarters. This function converts them in to proper date strings in the format YYYY-MM-DD.

Usage

```
fix_time(date_string, as_char = FALSE)
```

Arguments

date_string	a string of dates formatted as months or quarters.
as_char	whether to return the dates as data objects or character strings.

Value

a data frame

Examples

```
df <- statsDK::retrieve_data("FOLK1A", TID = "*", ALDER = "IALT",
                             CIVILSTAND = "TOT", lang = "da")
dplyr::glimpse(df)

df$TID <- statsDK::fix_time(df$TID)
dplyr::glimpse(df)
```

get_variables	<i>Extracts variables from the list of meta data</i>
---------------	------------------------------------------------------

Description

Extracts the variables and sets them up in a tibble so it is easy to see what settings each parameter should have to get the desired data.

Usage

```
get_variables(metadata)
```

Arguments

metadata A list retrieved using the [retrieve_metadata](#) function.

Value

a tibble

Examples

```
metadata <- statsDK::retrieve_metadata("BEV3A")
dplyr::glimpse(metadata)

# See the variables as a data frame
variables <- get_variables(metadata)
dplyr::glimpse(variables)
```

retrieve_data	<i>Retrieves a specific table from Statistics Denmark</i>
---------------	-----------------------------------------------------------

Description

The function retrieves a specific table from Statistics Denmark based on a table ID and some parameters to the API.

Usage

```
retrieve_data(table_id, ..., lang = "en",
              base_url = "http://api.statbank.dk/v1/")
```

Arguments

table_id is the id of the table you want to call. You can get table ids by calling the [retrieve_tables](#) function.

... are parameters you need to use to specify what data you want from the API. See the data created by the [retrieve_metadata](#) function. An * indicates ALL settings in the given parameter.

lang whether to return the data in english or danish.

base_url is the base url for the API you wish to call. Statistics Denmark can sometimes create custom API's that you can use by changing this parameter.

Value

a data frame

Examples

```

metadata <- statsDK::retrieve_metadata("FOLK1A")
dplyr::glimpse(metadata)

# See the variables as a data frame
variables <- statsDK::get_variables(metadata)
dplyr::glimpse(variables)

# Use the param and the settings columns from the variables data to set the
# righth values for the API call.
df_en <- statsDK::retrieve_data("PRIS111", VAREGR = "000000", ENHED = "*",
                                Tid = paste(paste0("2017M0", 1:8), collapse = ","))
dplyr::glimpse(df_en)

df_da <- statsDK::retrieve_data("PRIS111", VAREGR = "000000", ENHED = "*",
                                Tid = paste(paste0("2017M0", 1:8), collapse = ","),
                                lang = "da")
dplyr::glimpse(df_da)

```

retrieve_metadata	<i>Retrieves metadata for a specific table from Statistics Denmark</i>
-------------------	------------------------------------------------------------------------

Description

The function retrieves a list with a lot of metadata about a certain table.

Usage

```
retrieve_metadata(table_id, base_url = "http://api.statbank.dk/v1/")
```

Arguments

table_id is the id of the table you want to call. You can get table ids by calling the [retrieve_tables](#) function.

base_url is the base url for the API you wish to call. Statistics Denmark can sometimes create custom API's that you can use by changing this parameter.

Details

id The id of the table.

text A description of what the data in the table is about.

unit What unit the data is in.

contact Who to contact regarding the data.

documentation Link to a web page with detailed description of the data.

footnote A footnote if applicable.

variables Details about the variables in the table. Is very useful for when using the [retrieve_data](#) function.

Value

a list

Examples

```
metadata <- statsDK::retrieve_metadata("PRIS111")
dplyr::glimpse(metadata)
```

retrieve_subjects	<i>Retrieves an overview of the subjects from Statistics Denmark</i>
-------------------	----------------------------------------------------------------------

Description

The function retrieves a data frame with the over all statistical subjects that can be pulled from Statistics Denmark.

Usage

```
retrieve_subjects(base_url = "http://api.statbank.dk/v1/")
```

Arguments

`base_url` is the base url for the API you wish to call. Statistics Denmark can sometimes create custom API's that you can use by changing this parameter.

Value

a tibble

Examples

```
subjects <- statsDK::retrieve_subjects()
dplyr::glimpse(subjects)
```

retrieve_tables	<i>Retrieves an overview of the tables from Statistics Denmark</i>
-----------------	--------------------------------------------------------------------

Description

The function retrieves a data frame with all the available tables from the Statistics Denmark API.

Usage

```
retrieve_tables(base_url = "http://api.statbank.dk/v1/")
```

Arguments

`base_url` is the base url for the API you wish to call. Statistics Denmark can sometimes create custom API's that you can use by changing this parameter.

Details

id The id of the table. This is used when calling specific tables later with [retrieve_metadata](#) or [retrieve_data](#).

text A description of what the data in the table is about.

unit What unit the data is in.

updated When the table was last updated

firstPeriod The first period in the data, ie how far back the data goes.

latestPeriod The latest period in the data.

active If the table is still being updated

variables A list of the variables in the tables.

Value

a tibble

Examples

```
tables <- statsDK::retrieve_tables()
dplyr::glimpse(tables)
```

Index

`fix_time`, [2](#)

`get_variables`, [2](#)

`retrieve_data`, [3](#), [4](#), [6](#)

`retrieve_metadata`, [3](#), [4](#), [6](#)

`retrieve_subjects`, [5](#)

`retrieve_tables`, [3](#), [4](#), [5](#)