

Package ‘junr’

August 29, 2016

Encoding UTF-8

Type Package

Title Access Open Data Through the Junar API

Description The Junar API is a commercial platform to organize and publish data <<http://junar.com>>. It has been used in a number of national and local government Open Data initiatives in Latin America and the USA. This package is a wrapper to make it easier to access data made public through the Junar API.

Version 0.1.1

Date 2016-05-13

BugReports <https://github.com/fvd/junr/issues>

LazyData TRUE

Depends R (>= 3.0.0)

Imports httr, jsonlite

Suggests knitr, rmarkdown, testthat (>= 0.8.0)

VignetteBuilder knitr

License MIT + file LICENCE

URL <https://github.com/fvd/junr>

RoxygenNote 5.0.1

NeedsCompilation no

Author Frans van Dunné [aut, cre]

Maintainer Frans van Dunné <frans@ixpantia.com>

Repository CRAN

Date/Publication 2016-05-14 00:40:57

R topics documented:

clean_currency	2
get_currency_character	3
get_data	3
get_dimensions	4
get_index	4
list_guid	5
list_titles	5

Index	6
--------------	----------

clean_currency	<i>Clean currency values</i>
----------------	------------------------------

Description

Often Junar data involving currency data is not clean because for some reason content and presentation of currency values is not separated). To clean up this data quickly the following helper function accepts the position of the character value for the currency and the thousands and decimal delimiters.

Usage

```
clean_currency(currency_column, currency_symbol_pos = 1,
  thousand_separator = ",", decimal_separator = ".")
```

Arguments

currency_column
The column in the data frame that contains the currency values.

currency_symbol_pos
The position from the left of the currency symbol used. If any spaces are included, please include them here as well.

thousand_separator
The character value that separates thousands (defaults to ",")

decimal_separator
The character value that separates decimals (defaults to ".")

Details

The currency character position defaults to the first character from the left. And it refers directly to the symbol as present in the data frame column, because some characters (such as the symbol for the Costa Rican Colon) give are uncommon and lead to multiple encoding and font errors.

`get_currency_character`*Get the currency symbol*

Description

The currency symbol is not always obvious, depending on the currency you are working with. The symbol for Costa Rican Colon, for instance may not display correctly or the same. It may have different appearances in the same R session, and cannot be included here because they lead to built errors.

Usage

```
get_currency_character(currency_value, currency_symbol_pos)
```

Arguments

`currency_value` A single value from the currency column

`currency_symbol_pos`

The number of positions from the left for the currency symbol (include any spaces)

Details

To have a check in place you can request the first character of the currency strings with this function, and use it either to assign the currency value, or to do a sanity check.

`get_data`*Get data for a given GUID*

Description

Get the data for any given GUID and return it as a data frame. Note that we use the "ajson" JSON format from the API. The "JSON" format has a more complex structure.

Usage

```
get_data(base_url, api_key, guid)
```

Arguments

`base_url` The base URL of the Junar service

`api_key` The user's API key for the Junar service

`guid` The GUID of the data set of interest

Details

We do use the JSON response to get the fLength value, which indicates the length of the dataset. This way we can include a fixed way to get around the default limit of 1000 rows of the Junar API.

Note that this removes all meta-data from the JSON response given by the API.

get_dimensions	<i>Get dimensions of all data sets</i>
----------------	--

Description

Get the size of the data sets behind each of the GUID's offered by the base URL. This function will iterate through all the GUID's available and present the results as a data frame with the GUID, the number of rows, number of columns and dimension (as total number of cells) for each data set.

Usage

```
get_dimensions(base_url, api_key)
```

Arguments

base_url	The base URL of the Junar service
api_key	The user's API key for the Junar service

get_index	<i>Get a list of GUID's</i>
-----------	-----------------------------

Description

Get a list of all the available GUID's with datasets or views from the base URL

Usage

```
get_index(base_url, api_key)
```

Arguments

base_url	The base URL of the Junar service
api_key	The user's API key for the Junar service

list_guid	<i>Show GUID list</i>
-----------	-----------------------

Description

Show a list of all available GUID's identifying the available data sets. It only shows GUID's so that the list will fit in the console window.

Usage

```
list_guid(base_url, api_key)
```

Arguments

base_url	The base URL of the Junar service
api_key	The user's API key for the Junar service

list_titles	<i>Show GUID titles</i>
-------------	-------------------------

Description

Show a list of all available GUID titles. It only shows the titles so that the list will fit in the console window.

Usage

```
list_titles(base_url, api_key)
```

Arguments

base_url	The base URL of the Junar service
api_key	The user's API key for the Junar service

Index

- *Topic **Cleaning**,
 - [clean_currency, 2](#)
- *Topic **Currency**,
 - [clean_currency, 2](#)
- *Topic **Data**
 - [clean_currency, 2](#)
- *Topic **GUID**
 - [get_data, 3](#)
 - [get_dimensions, 4](#)
 - [get_index, 4](#)
 - [list_guid, 5](#)
 - [list_titles, 5](#)
- *Topic **Scrubbing**
 - [clean_currency, 2](#)

[clean_currency, 2](#)

[get_currency_character, 3](#)
[get_data, 3](#)
[get_dimensions, 4](#)
[get_index, 4](#)

[list_guid, 5](#)
[list_titles, 5](#)