Package 'RWebData'

June 27, 2017

Type Package

Title A High-Level Interface to the Programmable Web

Depends R (>= 3.3.2)

Imports igraph,
RCurl,
jsonlite,
XML,
plyr,
stringr,
XML2R,
httr,
mime,
yaml,
RJSONIO,
methods

Version 0.1

Author Ulrich Matter

Maintainer Ulrich Matter <ulrich.matter@unibas.ch>

Description RWebAPI provides various high- and low-level functions to write open source interfaces to RESTful web APIs, interactively download data from web APIs, and extract data from various web data formats as data frames.

License GPL-2, GPL-3

RoxygenNote 6.0.1

R topics documented:

apiData 2
apiDownload 2
generateQueryFunction 3
getdata 4
getTabularData 5
is.apidata 6
is.apirequest 6
is.apiresponse 7
JSONtoDataFrame 7
saveAPIkey 8
XMLtoDataFrame 8
YAMLtoDataFrame 9

Index **10**

`apiData` *Query data from an API*

Description

A high-level function that automates the querying and extraction of data from a web API.

Usage

```
apiData(x, shortnames=FALSE, method="RWebData", alignVariables=FALSE)
```

Arguments

`x`	an apirequest object
`shortnames`	logical, indicating whether the resulting tables (data frames) should have short variable names (default is FALSE, variable names contain nesting hierarchy)
`method`	character, either "RWebData", "jsonlite", or "XML2R"
`alignVariables`	logical, indicating whether variables/values should be rearranged in case the raw data was malformed (missing variable names)

Value

an apidata-object containing the returned data in a flat representation

Examples

```
## Not run: apidata <- apiData(x)
```

`apiDownload` *Download and transform data from a web API*

Description

A high-level function that automates the querying and extraction of data for multiple apirequest objects.

Usage

```
apiDownload(x, chunksize=50, pause=0, backupfile="apiDL.list.Rdata",
shortnames=FALSE, progress="bar", ...)
```

Arguments

x	a list of apirequest objects
chunksize	numeric, the size (number of requests) that should be processed in one batch (default is 50)
pause	numeric, the number of seconds the download process should be paused after each request (default is 0)
backupfile	character string with the path/name of the backup-file where the data should be saved batchwise during the download process.
shortnames	logical, indicating whether the resulting tables (data frames) should have short variable names (default is FALSE, variable names contain nesting hierarchy)
progress,	either character "bar" (indicates progress with a progress-bar), "text" (textual indication of progress)
...	currently only one parameter (simplify) passed down to the mapping algorithm if simplify is TRUE, the document tree is made simpler if possible (by removing unnecessary nodes)

Details

The core of the function is based on mapiDatalight(). However, it is specifically designed for the processing of various requests and a save download of the data. the requests are split up in batches and saved batchwise on disk during the download process.

Value

either one data frame or a list containing several data.frames into which the tree structured web-data has been transformed.

Examples

```
## Not run: apidata <- apiDownload(x)
```

---

generateQueryFunction *Writes a function to interact with an API*

Description

A function that generates a function to send specific API-requests and handle the returned data.

Usage

```
generateQueryFunction(x, base.url, multiparam=NULL, key.param=NA,
key.object=NA, .vectorizeIt=FALSE)
```

Arguments

x	a two-column data frame containing the parameter names and values (see details).
base.url	a character string containing the host url of the api.
multiparam	a character string with the name of a parameter that can take multiple input values in the resulting function (defaults to NULL).
key.param	a character string containing the api key parameter required in the url.
key.object	the name of the object the api.key is saved in (as character).
.vectorizeIt	logical, indicating whether the resulting function should contain an implicit loop over parameters with no default value (i.e., allows vectors as inputs). Default is FALSE. TRUE only works if x is a data frame.

Details

x should contain the parameter names in the first column and respective default-values in the second column (both as character strings) Parameters that have no default value have NAs in the second column.The function attempts to get the api-key from the environment "apikeys" (therefore has to be defined there before; see saveAPIkey()).

Value

a function

Examples

```
# First, make sure the necessary API key is saved in your R session:
# (This example is based on the Project Vote Smart API [PVS API])
saveAPIkey(key.var="pvs", key="YOUR-KEY-HERE")
pvsmeasure <- "http://api.votesmart.org/Measure.getMeasure?"
measureparameters <- data.frame(parameter="measureId", value=NA)
## Not run: getMeasure <- generateOSIFunction(x=measureparameters, base.url=pvsmeasure,
 key.param="key", key.object="pvs")
## End(Not run)
```

---

getdata *Extract API Data*

Description

Extracts the converted data from an apidata object.

Usage

```
getdata(x)
```

Arguments

x	an object of class apidata

Value

either a data frame or a list containing data frames.

Examples

```
## Not run: apidata <- apiData(x) # only works with a proper PVS API key
## Not run: getdata(apidata)
```

`getTabularData` *Get web data in tabular form (data frames)*

Description

A high-level function that directly queries and transforms data from any web API to a data frame or list of data frames.

Usage

```
getTabularData(x, base.url, shortnames=TRUE, alignVariables=FALSE)
```

Arguments

`x`	either a charachter string containing the whole url for the request or a named list containing the parameter names and values (see details)
`base.url`	a character string containing the basic url for the api
`shortnames`	logical, indicating whether the resulting tables (data frames) should have short variable names (default is FALSE, variable names contain nesting hierarchy)
`alignVariables`	logical, indicating whether variables/values should be rearranged in case the raw data was malformed (missing variable names)

Value

a list of data-frames, containing the returned data in a flat representation

Examples

```
## Not run: apidata <- getTabularData(x)
```

is.apidata *Check if object is an apidata*

Description

A function to check if an object is of class "apidata".

Usage

```
is.apidata(x)
```

Arguments

x	any R object

Value

logical, TRUE if x is of class apidata.

is.apirequest *Check if object is an apirequest*

Description

A function to check if an object is of class "apisreqest".

Usage

```
is.apirequest(x)
```

Arguments

x	any R object

Value

logical, TRUE if x is of class apirequest.

is.apiresponse *Check if object is an apiresponse*

Description

A function to check if an object is of class "apiresponse".

Usage

```
is.apiresponse(x)
```

Arguments

x	any R object

Value

logical, TRUE if x is of class apiresponse.

JSONtoDataFrame *Extract the data from a JSON document as one (or several) data frame(s)*

Description

Given JSON document, this function maps the potentially nested JSON data to one or several data frames. The data extraction algorithm applied in this function is based on the assumption that the JSON document describes either one or several entity types containing a set of observations described by different variables.

Usage

```
JSONtoDataFrame(x, alignVariables=FALSE)
```

Arguments

x	either a string containing JSON or the name of a file containing the JSON
alignVariables	logical, indicating whether variables/values should be rearranged in case the raw data was malformed (missing variable names)

Value

one data frame or a list of several data frames

Examples

```
JSON.ex <- system.file("exdata", "microcapital.json", package = "RWebData")
JSONtoDataFrame(JSON.ex, alignVariables=FALSE)
```

saveAPIkey *Save API keys*

Description

A function that saves the api key as character in a specific variable in the API keys environment.

Usage

```
saveAPIkey(key.var, key)
```

Arguments

key.var	a character string with the variable name the key should be saved in.
key	the api-key as character string

Details

The name of the environment the key is assigned to defaults to "apikeys".

Value

either one data frame or a list containing several data.frames into which the tree structured web-data has been transformed.

Examples

```
saveAPIkey(key.var="disq", key="1234" )
get("disq", pos="apikeys")
```

XMLtoDataFrame *Extract the data from a XML document as one (or several) data frame(s)*

Description

Given a XML document, this function maps the potentially nested XML data to one or several data frames. The data extraction algorithm applied in this function is based on the assumption that the XML document describes either one or several entity types containing a set of observations described by different variables.

Usage

```
XMLtoDataFrame(x, alignVariables=FALSE)
```

Arguments

x	a string with the path of a XML document
alignVariables	logical, indicating whether variables/values should be rearranged in case the raw data was malformed (missing variable names)

Details

The data extraction algorithm applied in this function partly relies on a nested (tree-structured) data representation
it is favorable for element based XML. XML documents that largely build on attributes can also be processed.
Several attributes of the same tag will, however, be collected in the same data-frame column.

Value

one data frame or a list of several data frames

Examples

```
XML.ex <- system.file("exdata", "microcapital.xml", package = "RWebData")
XMLtoDataFrame(XML.ex, alignVariables=FALSE)
```

YAMLtoDataFrame *Extract the data from a YAML document as one (or several) data frame(s)*

Description

Given YAML document, this function maps the potentially nested YAML data to one or several data frames. The data extraction algorithm applied in this function is based on the assumption that the YAML document describes either one or several entity types containing a set of observations described by different variables.

Usage

```
YAMLtoDataFrame(x, alignVariables=FALSE)
```

Arguments

x	either a string containing YAML or the name of a file containing the YAML
alignVariables	logical, indicating whether variables/values should be rearranged in case the raw data was malformed (missing variable names)

Value

one data frame or a list of several data frames

Examples

```
yaml.ex <- system.file("exdata", "microcapital.yaml", package = "RWebData")
YAMLtoDataFrame(yaml.ex, alignVariables=FALSE)
```

Index

apiData, 2
apiDownload, 2

generateQueryFunction, 3
getdata, 4
getTabularData, 5

is.apidata, 6
is.apirequest, 6
is.apiresponse, 7

JSONtoDataFrame, 7

saveAPIkey, 8

XMLtoDataFrame, 8

YAMLtoDataFrame, 9