

Package: Pandora (via r-universe)

February 19, 2025

Title Retrieve Data using the API of the 'Pandora' Data Platform

Version 24.2.0

Date 2024-02-26

Description API wrapper that contains functions to retrieve data from the 'Pandora' databases. Web services for API:
<<https://pandora.earth/>>.

Maintainer Jan Abel <jan.abel@inwt-statistics.de>

URL <https://github.com/Pandora-IsoMemo/pandora-data>,
<https://pandora-isomemo.github.io/pandora-data/>

BugReports <https://github.com/Pandora-IsoMemo/pandora-data/issues>

License GPL (>= 3)

Encoding UTF-8

Imports dplyr, curl, jsonlite, magrittr, openxlsx, readODS, readr,
readxl, rlang, stats, yaml

Suggests knitr, qpdf, rmarkdown, testthat

RoxygenNote 7.2.3

VignetteBuilder knitr

NeedsCompilation no

Author Jan Abel [cre], Antonia Runge [aut], Andreas Neudecker [aut],
Ricardo Fernandes [aut]

Date/Publication 2024-02-26 17:10:06 UTC

Additional_repositories <https://cranhaven.r-universe.dev>

Config/pak/sysreqs libicu-dev libssl-dev libx11-dev

Repository <https://cranhaven.r-universe.dev>

RemoteUrl <https://github.com/cranhaven/cranhaven.r-universe.dev>

RemoteRef package/Pandora

RemoteSha 4d63d55bdd41ba2ca5beb88ab1851136b5ef92ba

RemoteSubdir Pandora

Contents

callAPI	2
dataOptions	3
filterPattern	4
filterResourceByName	4
filterValidFileType	5
formatRepositoryList	5
getData	6
getDatasetFields	6
getFileTypes	7
getNetworks	7
getNrow	8
getRepositories	8
getResources	9
isOldROnWindows	10
loadData	10
selectSingleFile	11
validateResource	12
Index	13

callAPI	<i>Call API</i>
---------	-----------------

Description

Call API

Usage

```
callAPI(
  action = c("current_package_list_with_resources", "group_list", "package_list",
            "organization_list", "tag_list"),
  ...
)
```

Arguments

action	(character) name of the endpoint "mapping"
...	parameters for the endpoint, e.g. all_fields = "true"

Value

(data.frame) output from the Pandora API

dataOptions	<i>Data Options</i>
-------------	---------------------

Description

Set options for `utils::read.csv()`, `openxlsx::read.xlsx()` or `readxl::read_excel`. Choose delimiter and decimal separator as well as sheet number and number of rows to read.

Usage

```
dataOptions(  
  nrows = NA_integer_,  
  colNames = TRUE,  
  sep = ",",  
  dec = ".",  
  fileEncoding = "",  
  sheet = 1  
)
```

Arguments

<code>nrows</code>	integer: the maximum number of rows to read in. Negative and other invalid values are ignored.
<code>colNames</code>	If TRUE, the first row of data will be used as column names.
<code>sep</code>	the field separator character. Values on each line of the file are separated by this character. If <code>sep = ""</code> (the default for <code>read.table</code>) the separator is 'white space', that is one or more spaces, tabs, newlines or carriage returns.
<code>dec</code>	the character used in the file for decimal points.
<code>fileEncoding</code>	character string: if non-empty declares the encoding used on a file (not a connection) so the character data can be re-encoded. See the 'Encoding' section of the help for file , the 'R Data Import/Export' manual and 'Note'.
<code>sheet</code>	The name or index of the sheet to read data from.

Value

a list of extra options for `utils::read.csv()` or `openxlsx::read.xlsx()` or `readxl::read_excel`, respectively

filterPattern *Filter Pattern*

Description

Search for pattern in all columns of datAPI and filter respective rows

Usage

```
filterPattern(datAPI, pattern = "")
```

Arguments

datAPI (list) output from the Pandora API
pattern (character) string for filtering all meta information

Value

(list) a data.frame with rows that contain the pattern

filterResourceByName *Filter Resource by Name*

Description

Filter Resource by Name

Usage

```
filterResourceByName(resource, name)
```

Arguments

resource (data.frame) resources data frame
name (character) name of a resource

Value

(data.frame) filtered resource

filterValidFileType *Filter Resource by Valid File Type*

Description

Filter Resource by Valid File Type

Usage

```
filterValidFileType(resource, name)
```

Arguments

resource (data.frame) resources data frame
name (character) name of a resource

Value

(data.frame) filtered resource

formatRepositoryList *Rename Repository Meta Columns*

Description

Apply names from the 'Additional Info' box from 'https://pandoradata.earth/dataset/' to the columns of returned data

Usage

```
formatRepositoryList(  
  packageList,  
  columns = getDatasetFields(),  
  renameColumns = TRUE  
)
```

Arguments

packageList (data.frame) optional, output of callAPI() e.g. from a previous call to the Pandora API.
columns (character) names of columns that should be returned
renameColumns (logical) apply names from the 'Additional Info' box from 'https://pandoradata.earth/dataset/' to the columns of returned data

Value

(data.frame) containing available repositories

getData *Get Data*

Description

Get Data

Usage

```
getData(name, repository = "", verbose = TRUE, options = dataOptions())
```

Arguments

name	(character) name of a resource, e.g. an entry of the output from getResources()\$name
repository	(character) name of a Pandora repository, e.g. an entry of the output from getRepositories()\$name
verbose	Logical, indicating whether to display processing messages. If TRUE, messages will be displayed; if FALSE, messages will be suppressed. Default is TRUE.
options	(list) a list of extra options for read.csv() or openxlsx::read.xlsx() and readxl::read_excel

Value

(data.frame) return data from the Pandora API

getDatasetFields *Get Dataset Meta Fields*

Description

Names of particular meta fields from the 'Additional Info' box from 'https://pandoradata.earth/dataset/'

Usage

```
getDatasetFields()
```

Value

(character vector) names of meta fields

getFileTypes

Get File Types

Description

Get all available file types of a repository or those within a specific network or within a specific repository optional filtering of meta information for a given string

Usage

```
getFileTypes(
  repository = "",
  network = "",
  pattern = "",
  order = TRUE,
  packageList = data.frame()
)
```

Arguments

repository	(character) name of a Pandora repository, e.g. an entry of the output from <code>getRepositories()\$name</code>
network	(character) name of a Pandora network, e.g. an entry of the output from <code>getNetworks()\$name</code>
pattern	(character) string for meta information search
order	(logical) if TRUE, order dataframe alphabetically by 'repository' and 'name'
packageList	(data.frame) optional, output of <code>callAPI()</code> e.g. from a previous call to the Pandora API.

Value

(data.frame) containing available file types within a repository

getNetworks

Get Networks

Description

Get all available networks (groups in CKAN terminology) optional filtering of names for a given string

Usage

```
getNetworks(pattern = "", order = TRUE, groupList = data.frame())
```

Arguments

pattern (character) string for meta information search
 order (logical) if TRUE, order dataframe alphabetically by 'repository' and 'name'
 groupList (data.frame) optional, output of callAPI() from a previous call to the Pandora API.

Value

(data.frame) giving the "name" and "display_name" of available Pandora networks (groups in CKAN terminology)

getNrow	<i>get nRow</i>
---------	-----------------

Description

get nRow

Usage

```
getNrow(type, nrow = NA_integer_)
```

Arguments

type (character) file type
 nrow integer: the maximum number of rows to read in. Negative and other invalid values are ignored.

getRepositories	<i>Get Repositories</i>
-----------------	-------------------------

Description

Get all available repositories or those within a specific network optional filtering of meta information for a given string

Usage

```

getRepositories(
  network = "",
  pattern = "",
  order = TRUE,
  columns = getDatasetFields(),
  renameColumns = TRUE,
  packageList = data.frame()
)

```


Arguments

network	(character) name of a Pandora network, e.g. an entry of the output from <code>getNetworks()</code> \$name
pattern	(character) string for meta information search
order	(logical) if TRUE, order dataframe alphabetically by 'repository' and 'name'
columns	(character) names of columns that should be returned
renameColumns	(logical) apply names from the 'Additional Info' box from 'https://pandoradata.earth/dataset/' to the columns of returned data
packageList	(data.frame) optional, output of <code>callAPI()</code> e.g. from a previous call to the Pandora API.

Value

(data.frame) containing available repositories

getResources

Get Resources

Description

Get all available resources within a repository or filtered by file type or those within a specific network or within a specific repository optional filtering of meta information for a given string

Usage

```
getResources(
  fileType = character(),
  repository = "",
  network = "",
  pattern = "",
  order = TRUE,
  packageList = data.frame()
)
```

Arguments

fileType	(character) list of relevant file types, e.g. <code>c("xls", "xlsx", "csv", "odt")</code>
repository	(character) name of a Pandora repository, e.g. an entry of the output from <code>getRepositories()</code> \$name
network	(character) name of a Pandora network, e.g. an entry of the output from <code>getNetworks()</code> \$name
pattern	(character) string for meta information search
order	(logical) if TRUE, order dataframe alphabetically by 'repository' and 'name'
packageList	(data.frame) optional, output of <code>callAPI()</code> e.g. from a previous call to the Pandora API.

Value

(data.frame) containing available resources within a repository

isOldROnWindows	<i>Is old windows</i>
-----------------	-----------------------

Description

Checks if package is used with an older R version which possibly leads to encryption errors on Windows. Gives a warning in that case.

Usage

```
isOldROnWindows()
```

Value

(logical) TRUE if system is Windows and R version is < 4.2.0

loadData	<i>Load Data</i>
----------	------------------

Description

Load Data

Usage

```
loadData(  
  path,  
  type = c("xlsx", "xls", "odt", "csv", "txt"),  
  nrows = NA_integer_,  
  sep = ",",  
  dec = ".",  
  fileEncoding = "",  
  colNames = TRUE,  
  sheet = 1,  
  verbose = TRUE  
)
```

Arguments

path	path to the file
type	(character) type of file, one of <code>c("xlsx", "xls", "odt", "csv", "txt")</code>
nrows	integer: the maximum number of rows to read in. Negative and other invalid values are ignored.
sep	the field separator character. Values on each line of the file are separated by this character. If <code>sep = ""</code> (the default for <code>read.table</code>) the separator is 'white space', that is one or more spaces, tabs, newlines or carriage returns.
dec	the character used in the file for decimal points.
fileEncoding	character string: if non-empty declares the encoding used on a file (not a connection) so the character data can be re-encoded. See the 'Encoding' section of the help for file , the 'R Data Import/Export' manual and 'Note'.
colNames	If TRUE, the first row of data will be used as column names.
sheet	The name or index of the sheet to read data from.
verbose	Logical, indicating whether to display processing messages. If TRUE, messages will be displayed; if FALSE, messages will be suppressed. Default is TRUE.

Value

(data.frame) data loaded from the file at path

selectSingleFile	<i>Select Single File from Resources</i>
------------------	--

Description

Select Single File from Resources

Usage

```
selectSingleFile(resource)
```

Arguments

resource	(data.frame) resources data frame
----------	-----------------------------------

Value

(data.frame) selected resource

validateResource	<i>Validate Resource</i>
------------------	--------------------------

Description

Validate Resource

Usage

```
validateResource(resource, repository)
```

Arguments

resource	(data.frame) resources data frame
repository	(character) name of a Pandora repository, e.g. an entry of the output from <code>getRepositories()</code> \$name

Value

(data.frame) resource, or error if empty

Index

callAPI, [2](#)

dataOptions, [3](#)

file, [3](#), [11](#)

filterPattern, [4](#)

filterResourceByName, [4](#)

filterValidFileType, [5](#)

formatRepositoryList, [5](#)

getData, [6](#)

getDatasetFields, [6](#)

getFileTypes, [7](#)

getNetworks, [7](#)

getNrow, [8](#)

getRepositories, [8](#)

getResources, [9](#)

isOldROnWindows, [10](#)

loadData, [10](#)

selectSingleFile, [11](#)

validateResource, [12](#)