Package 'SNPediaR'

July 19, 2025

Title Query data from SNPedia

Version 1.35.0

Date 2015-09-26

Description SNPediaR provides some tools for downloading and parsing data from the SNPedia web site <http://www.snpedia.com>.
The implemented functions allow users to import the wiki text available in SNPedia pages and to extract the most relevant information out of them.
If some information in the downloaded pages is not automatically processed by the library functions, users can easily implement their own parsers to access it in an efficient way.

Depends R (>= 3.0.0)

License GPL-2

LazyData true

URL https://github.com/genometra/SNPediaR

BugReports https://github.com/genometra/SNPediaR/issues

Suggests BiocStyle, knitr, rmarkdown, testthat

VignetteBuilder knitr

Imports RCurl, jsonlite

biocViews SNP, VariantAnnotation

RoxygenNote 6.0.1

git_url https://git.bioconductor.org/packages/SNPediaR

git_branch devel

git_last_commit 54647c0

git_last_commit_date 2025-04-15

Repository Bioconductor 3.22

Date/Publication 2025-07-18

Author David Montaner [aut, cre]

Maintainer David Montaner <david.montaner@gmail.com>

Contents

extractTags 2
getCategoryElements 3
getPages 4

Index 6

`extractTags` *Extract information form downloaded SNPedia pages.*

Description

SNPedia pages usually have a table in the right hand side which summarizes most relevant information in the page. This functions help extracting this kind of information for the given tags or rows.

Usage

```
extractTags (x, tags)
extractSnpTags (x, tags)
extractGenotypeTags (x, tags)
```

Arguments

`x`	a wiki page (single character vector)
`tags`	character vector of tags (row names) to be collected.

Details

`extractTags` is a general purpose function aimed to work at any page. `extractSnpTags` calls `extractTags` with a set of predefined tags suitable for SNP pages. `extractGenotypeTags` does the same for genotype pages.

This functions take a character vector of length one but return a vector with as many values as the tag list provided. They are devised to be used with `sapply` functions.

Notice that in SNPedia not all information presented in the HTML table is available in the JSON format retrieved by the R package. Risk information for instance needs to be collected from the genotype pages as it is not available in the JSON version of the SNP pages.

Value

A character vector with the value of each of the tags if available in the page and NA otherwise.

See Also

getPages, getCategoryElements

Examples

```
res <- getPages (c ("Rs1234", "Rs53576"))
t (sapply (res, extractSnpTags))

extractTags (res[[1]], tags = c("rsid", "Chromosome", "position"))

res <- getPages (c ("Rs1234(A;A)", "Rs1234(A;C)","Rs1234(C;C)"))
t (sapply (res, extractGenotypeTags))

getPages (c ("Rs1234(A;A)", "Rs1234(A;C)","Rs1234(C;C)"),
          wikiParseFunction = extractGenotypeTags)
```

```
getPages (c ("Rs1234(A;A)", "Rs1234(A;C)","Rs1234(C;C)"),
          wikiParseFunction = extractGenotypeTags,
          tags = c("rsid", "allele1", "allele2"))
```

`getCategoryElements` *Get all elements of a given category*

Description

A function to get all page names of SNPedia tagged under the indicated category.

Usage

```
getCategoryElements(category, verbose = FALSE, includeTemplates = FALSE,
  limit, baseURL, format, query, continue)
```

Arguments

`category`	The category to be used. Just one at a time.
`verbose`	If TRUE some messages are provided.
`includeTemplates`	If TRUE page templates are kept in the output.
`limit`	The maximum number of items to be queried at a time.
`baseURL`	SNPedia boots URL.
`format`	Downloading format. Currently just JSON is available.
`query`	The query to be iterated.
`continue`	To be used in multi-page queries.

Details

A list of all available categories may be found at:

http://www.snpedia.com/index.php/Special:Categories

Most used categories are:

- Is_a_medical_condition
- Is_a_medical_condition
- Is_a_medicine
- Is_a_topic
- Is_a_snp
- In_dbSNP
- Is_a_genotype

Some template pages are included in their corresponding category. By default those will be removed. Set includeTemplates to TRUE is you want to keep them.

Parameters other than category and verbose are not intended for standard users.

Value

A character vector containing the names of the pages under the required category.

See Also

getPages, extractTags

Examples

```
res <- getCategoryElements(category = "Is_a_medical_condition")
head(res)

## Not run:
res <- getCategoryElements(category = "Is_a_snp")

## End(Not run)
```

`getPages` *Download SNPedia pages*

Description

A function to download the (wiki) text content of a list of SNPedia pages.

Usage

```
getPages(titles, verbose = FALSE, limit = 50,
  wikiParseFunction = identity, baseURL, format, query, ...)
```

Arguments

`titles`	Titles of the pages to be downloaded.
`verbose`	If TRUE some messages are provided.
`limit`	The maximum number of items to be queried at a time.
`wikiParseFunction`	Function to be used to parse the wiki code at downloading time. Default is `identity` so the raw wiki text is provided.
`baseURL`	SNPedia boots URL.
`format`	Downloading format. Currently just JSON is available.
`query`	The query to be iterated.
`...`	any parameter to be pasted to the `wikiParseFunction`.

Details

JSON format is parsed to extract the wiki text returned by the function.

If the `wikiParseFunction` parameter is provided, parsing of the pages is done internally once each batch of pages is downloaded.

Pages do not need to be of the same class... but users may be aware of the type of pages they are queering, moreover when using their own `wikiParseFunction`.

Parameters `baseURL`, `format` and `query` are not intended for end users.

Value

A list containing the wiki content of the required pages or the formatted objects returned by the `wikiParseFunction` applied to each page.

See Also

extractTags, getCategoryElements

Examples

```
res <- getPages(titles = "Rs1234")
res

res <- getPages(titles = c("Rs1234", "Rs1234(A;A)", "Rs1234(A;C)"))
res

myfun <- function(x) substring(x, 1, 5)
lapply(res, myfun)

res <- getPages(titles = c("Rs1234", "Rs1234(A;A)", "Rs1234(A;C)"),
wikiParseFunction = myfun)
res
```

Index

extractGenotypeTags *(*extractTags*)*, 2
extractSnpTags *(*extractTags*)*, 2
extractTags, 2

getCategoryElements, 3
getPages, 4