Package 'ralger'

October 9, 2020

Type Package

Title Easy Web Scraping

Version 2.1.0

Maintainer Mohamed El Fodil Ihaddaden <ihaddaden.fodeil@gmail.com>

Description The goal of 'ralger' is to facilitate web scraping in R.
The user has the ability to extract a vector with scrap(), a tidy dataframe using tidy_scrap(), a table with table_scrap() and web links with weblink_scrap().

License MIT + file LICENSE

Encoding UTF-8

LazyData true

URL https://github.com/feddelegrand7/ralger

BugReports https://github.com/feddelegrand7/ralger/issues

VignetteBuilder knitr

Imports magrittr, rvest, xml2, testthat, tidyr, dplyr, stringr,
robotstxt, crayon, purrr

Suggests knitr, rmarkdown, covr

RoxygenNote 7.1.1

NeedsCompilation no

Author Mohamed El Fodil Ihaddaden [aut, cre],
Ezekiel Ogundepo [ctb],
Romain François [ctb]

Repository CRAN

Date/Publication 2020-10-09 21:10:03 UTC

R topics documented:

`paragraphs_scrap` *Website text paragraph scraping*

Description

This function is used to scrape text paragraphs from a website.

Usage

```
paragraphs_scrap(
  link,
  contain = NULL,
  case_sensitive = FALSE,
  collapse = FALSE,
  askRobot = FALSE
)
```

Arguments

`link`	the link of the web page to scrape
`contain`	filter the paragraphs according to the character string provided.
`case_sensitive`	logical. Should the contain argument be case sensitive ? defaults to FALSE
`collapse`	if TRUE the paragraphs will be collapsed into one element and the contain argument ignored.
`askRobot`	logical. Should the function ask the robots.txt if we're allowed or not to scrap the web page ? Default is FALSE.

Value

a character vector.

Examples

```
# Extracting the paragraphs displayed on the health page of the New York Times

link     <- "https://www.nytimes.com/section/health"

paragraphs_scrap(link)
```

scrap *Simple website scraping*

Description

This function is used to scrape one element from a website.

Usage

```
scrap(link, node, clean = FALSE, askRobot = FALSE)
```

Arguments

link	the link of the web page to scrape
node	the HTML or CSS element to consider, the SelectorGadget tool is highly recommended
clean	logical. Should the function clean the extracted vector or not ? Default is FALSE.
askRobot	logical. Should the function ask the robots.txt if we're allowed or not to scrape the web page ? Default is FALSE.

Value

a character vector

Examples

```
# Extracting imdb top 250 movie titles

link <- "https://www.imdb.com/chart/top/"
node <- ".titleColumn a"

scrap(link, node)
```

table_scrap *HTML table scraping*

Description

This function is used to scrape an html table from a website.

Usage

```
table_scrap(link, choose = 1, header = T, askRobot = FALSE, fill = FALSE)
```

Arguments

link	the link of the web page containing the table to scrape
choose	an integer indicating which table to scrape
header	do you want the first line to be the leader (default to TRUE)
askRobot	logical. Should the function ask the robots.txt if we're allowed or not to scrape the web page ? Default is FALSE.
fill	logical. Should be set to TRUE when the table has an inconsistent number of columns.

Value

a data frame object.

Examples

```
# Extracting premier ligue 2019/2020 top scorers

link     <- "https://www.topscorersfootball.com/premier-league"
table_scrap(link)
```

---

tidy_scrap *Website Tidy scraping*

Description

This function is used to scrape a tibble from a website.

Usage

```
tidy_scrap(link, nodes, colnames, clean = FALSE, askRobot = FALSE)
```

Arguments

link	the link of the web page to scrape
nodes	the vector of HTML or CSS elements to consider, the SelectorGadget tool is highly recommended.
colnames	the names of the expected columns.
clean	logical. Should the function clean the extracted tibble or not ? Default is FALSE.
askRobot	logical. Should the function ask the robots.txt if we're allowed or not to scrape the web page ? Default is FALSE.

Value

a tidy data frame.

Examples

```
# Extracting imdb movie titles and rating

link     <- "https://www.imdb.com/chart/top/"
my_nodes <- c(".titleColumn a", "strong")
names    <- c("title", "rating")

tidy_scrap(link, my_nodes, names)
```

`titles_scrap` *Website title scraping*

Description

This function is used to scrape titles (h1, h2 & h3 html tags) from a website. Useful for scraping daily electronic newspapers' titles.

Usage

```
titles_scrap(link, contain = NULL, case_sensitive = FALSE, askRobot = FALSE)
```

Arguments

`link`	the link of the web page to scrape
`contain`	filter the titles according to a character string provided.
`case_sensitive`	logical. Should the contain argument be case sensitive ? defaults to FALSE
`askRobot`	logical. Should the function ask the robots.txt if we're allowed or not to scrape the web page ? Default is FALSE

Value

a character vector

Examples

```
# Extracting the current titles of the New York Times

link     <- "https://www.nytimes.com/"

titles_scrap(link)
```

`weblink_scrap` *Website web links scraping*

Description

This function is used to scrape web links from a website.

Usage

```
weblink_scrap(link, contain = NULL, case_sensitive = FALSE, askRobot = FALSE)
```

Arguments

`link`	the link of the web page to scrape
`contain`	filter the web links according to the character string provided.
`case_sensitive`	logical. Should the contain argument be case sensitive ? defaults to FALSE
`askRobot`	logical. Should the function ask the robots.txt if we're allowed or not to scrape the web page ? Default is FALSE.

Value

a character vector.

Examples

```
# Extracting the web links within the World Bank research and publications page

link     <- "https://www.worldbank.org/en/research"

weblink_scrap(link)
```

Index