Package 'toponym'

May 5, 2023

Title What the Package Does (One Line, Title Case)

Version 0.0.0.9000

Description What the package does (one paragraph).

Depends R (>= 4.1.2)

License GPL (>= 2)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Import rnaturalearth,
sp,
dplyr,
ggplot2,
grDevices

R topics documented:

`candidates.maps` *Generates and saves all maps returned by* `top.candidates()`

Description

test test test

Usage

```
candidates.maps(
  countries = "DE",
  count = 10,
  len = 3,
  df = FALSE,
  csv = TRUE,
  rat = 0.5,
  lons = c(10.144314, 10.0399439, 10.5178491, 11.3143579, 11.8746607, 11.8087427,
   11.6274683, 11.5450708, 11.7757837, 11.6659204, 10.2140419, 9.917411, 9.8075477,
   10.6919471, 12.8617469, 14.9821083, 15.5204383, 14.6964637, 13.8834755, 12.9496376,
    11.6202919, 11.1039344, 10.144314),
  lats = c(54.3227499, 53.5107333, 53.324126, 53.0476312, 52.8755735, 52.5928491,
   52.2377021, 52.0826909, 51.9389938, 51.764248, 51.0454903, 50.8031092, 50.2724411,
   49.8067462, 49.7499912, 50.2408327, 51.6459284, 53.9825363, 54.6235699, 54.7378977,
    54.4323074, 54.5790222, 54.3227499)
)
```

Arguments

countries	Character string with country code abbreviations (check https://www.geonames.org/countries/ for a list of available countries) specifying, the toponyms of which countries are checked.
count	numeric. The number of the most frequent endings which will be tested, e.g. by default the top ten most frequent endings in Germany.
len	numeric. The character length of the endings, e.g. by default three-character-long endings.
df	logical. If TRUE then the filtered data frames will be saved in the global environment.
csv	logical. If TRUE then the filtered data frames will be saved as .csv in the current working directory.
rat	numeric. The ratio (a number between 0.0 and 1) of how many occurrences of one ending need to be in the polygon
lons	numeric. Vector of longitudinal coordinates defining the polygon.
lats	numeric. Vector of latitudinal coordinates defining the polygon.

Value

A number of data frames and plots saved in a sub folder (called "data frames" and "plots") in the working directory. It also stores the ratio surpassing endings in a data frame in the global environment.

---

get.coordinates *Filters locations by given regular expression*

Description

The function retrieves the coordinates (longitude and latitude) and country codes of all locations, which match the regular expression, given in `string`. The return is used by `simple_map()`.

Usage

```
get.coordinates(gn, strings, df, csv)
```

Arguments

gn	The data frame(s), which will be accessed.
strings	Character strings in form of regular expression that filter the data frames.
df	logical. If TRUE then the filtered data frame will be saved in the global environment.
csv	logical. If TRUE then the filtered data frame will be saved as .csv in the current working directory.

Value

A list with the coordinates (longitude and latitude) and country codes.

---

get.data *Get toponym data from* Rhrefhttps://www.geonames.org/GeoNames

---

Description

The function downloads and saves toponym data from the GeoNames download server, which is later used by `read.files()`.

Usage

```
get.data(countries, save = FALSE)
```

Arguments

countries	character string with country code abbreviations to download (check https://www.geonames.org/countries/ for a list of available countries).
save	logical. If TRUE then the data sets will be extracted to the package folder. It will be otherwise saved in the temporary folder.

Value

The data as .txt in the temporary or package folder.

Examples

```
## Not run:
get.data(countries = c("DK", "DE"))
## downloads and extracts data for DK and DE to the temporary folder

get.data(countries = c("DK", "DE", "PL"), save = TRUE)
## downloads and extracts data for PL but only extracts data for DK and DE
## from the zip files downloaded before to the package folder if used in the same session

## End(Not run)
```

read.files *Reads toponym data from temporary or package folder*

Description

The function accesses the data saved by `get.data()` and returns it as data frame, only with populated locations, in the global environment, which is used by `top()`. View this for further information on the used column names, including the population tag.

Usage

```
read.files(countries)
```

Arguments

`countries`	character string with country code abbreviations to be read (check https://www.geonames.org/countries/ for a list of available countries). Data needs to be saved by `get.data()` before.

Value

Data frames of the specified countries.

simple.map *Plots locations on a map*

Description

The function generates a map plotting all locations, filtered by `get.coordinates()`. The plot also displays additional information if used by `candidates.maps()`

Usage

```
simple.map(x, y, cc, color, strings, plot, ratio_string, fq)
```

Arguments

`x`	numeric. Latitude of locations filtered by `get.coordinates()`
`y`	numeric. Longitude of locations filtered by `get.coordinates()`
`cc`	character string. Country code of locations filtered by `get.coordinates()`
`color`	character string indicating, which color is assigned to each country code in order.
`strings`	Character strings in form of regular expression that filter the data frames. The first string is shown on the plot.
`plot`	logical. If `TRUE` then the plot will not be printed but saved as .png in the current working directory.
`ratio_string`	character string. Ratio of occurrences in each polygon returned by `top.candidates()`.
`fq`	character string. Number of occurrences in the designated polygon and in total.

Value

A plot in the current R session or as .png in the working directory. If directly generated by `top()`, it displays the first string and the total occurrences. If generated by `candidates.maps()`, it displays the string, the ratio as a percentage and the number of occurrences in the designated polygon and in total.

`top` *Plots locations with specified toponyms on a map*

Description

The function calls the `simple.map()` function to generate a map plotting all locations, filtered by `get.coordinates()`. The plot also displays additional information if used by `candidates.maps()`. The data used is downloaded by `get.data()` and is accessible on the GeoNames download server.

Usage

```
top(
  strings,
  countries = "DE",
  color = rainbow(length(countries)),
  df = TRUE,
  csv = FALSE,
  plot = FALSE,
  ratio_string = "",
  fq = ""
)
```

Arguments

`strings`	Character strings in form of regular expression that filter the data frames. The first string is shown on the plot.
`countries`	Character string with country code abbreviations (check `https://www.geonames.org/countries/` for a list of available countries) specifying, the toponyms of which countries are checked.
`color`	character string indicating, which color is assigned to each country code in order.
`df`	logical. If `TRUE` then the filtered data frame will be saved in the global environment.
`csv`	logical. If `TRUE` then the filtered data frame will be saved as .csv in the current working directory.
`plot`	logical. If `TRUE` then the plot will not be printed but saved as .png in the current working directory.
`ratio_string`	character string. Ratio of occurrences in each polygon returned by `top.candidates()`. This should not be specified manually.
`fq`	character string. Number of occurrences in the designated polygon and in total returned by `top.candidates()`. This should not be specified manually.

Value

A plot in the current R session or as .png in the working directory. If directly generated by this function `top()`, it displays the first string and the total occurrences. If generated by `candidates.maps()`, it displays the string, the ratio as a percentage and the number of occurrences in the designated polygon and in total.

Examples

```
## Not run:
get.data("DE", save = TRUE)
# saves data for DE in package directory
top("itz$")
## generates and prints a map with all populated places in Germany (default country) ending with "itz" and saves


top("^By", "DK", color = "green", df = FALSE, csv = TRUE, plot = TRUE)
## first downloads data for DK and extracts it to the temporary directory
## generates plot with all populated places colored in green in Denmark starting with "By" (case sensitive) and s


get.data(c("DE", "PL"), save = TRUE)
# saves for DE and PL in package directory if it's not already there
top(c("itz$", "ice$"), c("DE", "PL"))
# generates plot with all populated places in Germany colored in red and Poland colored in cyan ending with eithe

## End(Not run)
```

`top.candidates` *Retrieves the most frequent toponym endings in a given polygon*

Description

The function sorts the toponyms in the given countries by frequency. It then tests which lie in the given polygon, printing out a data frame with those endings which match the ratio criteria and are potential candidates for further examination. The coordinates form the polygon, which roughly resembles the Slavic settlement zone in Germany. It is generated with Google My Maps.

Usage

```
top.candidates(
  countries = "DE",
  count = 10,
  len = 3,
  rat = 0.5,
  lons = c(10.144314, 10.0399439, 10.5178491, 11.3143579, 11.8746607, 11.8087427,
   11.6274683, 11.5450708, 11.7757837, 11.6659204, 10.2140419, 9.917411, 9.8075477,
   10.6919471, 12.8617469, 14.9821083, 15.5204383, 14.6964637, 13.8834755, 12.9496376,
    11.6202919, 11.1039344, 10.144314),
  lats = c(54.3227499, 53.5107333, 53.324126, 53.0476312, 52.8755735, 52.5928491,
   52.2377021, 52.0826909, 51.9389938, 51.764248, 51.0454903, 50.8031092, 50.2724411,
   49.8067462, 49.7499912, 50.2408327, 51.6459284, 53.9825363, 54.6235699, 54.7378977,
    54.4323074, 54.5790222, 54.3227499)
)
```

Arguments

countries	Character string with country code abbreviations (check https://www.geonames.org/countries/ for a list of available countries) specifying, the toponyms of which countries are checked.
count	numeric. The number of the most frequent endings which will be tested, e.g. by default the top ten most frequent endings in Germany.
len	numeric. The character length of the endings, e.g. by default three-character-long endings.
rat	numeric. The ratio (a number between 0.0 and 1) of how many occurrences of one ending need to be in the polygon.
lons	numeric. Vector of longitudinal coordinates defining the polygon.
lats	numeric. Vector of latitudinal coordinates defining the polygon.

Value

A data frame printed out and saved in the global environment. It shows the ending surpassing the ratio, at what percentage and the frequency.

Examples

```
## Not run:
top.candidates()
# prints and saves a data frame of the top ten three-character-long endings in Germany if more than 50% of them li


top.candidates("DK", count = 100, len = 4, rat = .9,
 lons = c(9.788425, 10.216892, 10.019138, 9.744480, 9.832370, 10.205905, 10.722263, 10.832126, 11.128757, 10.89
 lats = c(54.83623, 54.85521, 55.05078, 55.27671, 55.59458, 55.71235, 55.71235, 55.93453, 56.42372, 56.92469, 5
 )
# prints and saves a data frame of the top 100 four-character-long endings in Denmark if more than 90% of them lie

## End(Not run)
```

Index