Package: utilsIPEA (via r-universe)

September 13, 2024

Type Package

Title IPEA Common Functions

Version 0.0.5

Maintainer Gustavo Coelho <gustavo.coelho@ipea.gov.br>

Description The most used functions on IPEA (Instituto de Pesquisa Economica Aplicada). Most of functions deal with brazilian names. It can guess the women single's name, extract prepositions or extract the first name.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

URL https://github.com/ipea/utilsIPEA

BugReports http://github.com/ipea/utilsIPEA/issues

Depends R (>= 3.0)

Imports data.table, stringr, utils, stringdist, RCurl, dplyr

Suggests testthat, covr

Repository https://ipea.r-universe.dev

RemoteUrl https://github.com/ipea/utilsipea

RemoteRef HEAD

RemoteSha 1be4a7b2360973f712654c6e968a16644532051d

Contents

abrevia_nome_meio *Abrevia o nome do meio.*

Description

abrevia_nome_meio return names .

Usage

```
abrevia_nome_meio(base, ..., suffixo = "_abrev")
```

Arguments

`base` A data table, data frame or character vector.

`...` columns for apply the function

`suffixo` A character indicating the final part of the new columns' names

Value

the base parameter with a new column.

Examples

```
base <- data.frame(nome = c("Carlos Pereira Neves", "Pedro Aparecido Anjos"))
base <- remove_preposicao_nomes(base, "nome")
```

extrai_NomeProprio *Check Brazilian names*

Description

extrai_NomeProprio Parse Brazilian names and returns given names, surnames and gender

Usage

```
extrai_NomeProprio(x, surname = FALSE, gender = FALSE, stringdist = TRUE,
  spaces = TRUE)
```

Arguments

`x` List, character or factor with names to be parsed.

`surname` If TRUE, the list of surnames is returned.

`gender` If TRUE, the list of gender based on the names is returned.

`stringdist` if TRUE, make a prediction based on the string distance of Jaro-Winkler between the source data and the input.

`spaces` if TRUE, returns the names without spaces. If FALSE, it compress all the blank spaces.

Value

Returns a data.table

`geocod_base` *Brazilian address*

Description

Some addresses from Brazil

Usage

```
geocod_base
```

Format

A data frame with 5 rows and 12 variables:

uf state of Brazil

MatchedAddress Address returned from GALILEO

cep Zip code

enderecofinal Prompted Address

Nome_Municipio City names

`ident_erros_munic_galileo` `ident_erros_munic_galileo` *Returns a new column called munmatch with true or false. This column identify where GALILEO failed*

Description

`ident_erros_munic_galileo` Returns a new column called munmatch with true or false. This column identify where GALILEO failed

Usage

```
ident_erros_munic_galileo(base, mun, match, uf)
```

Arguments

`base`	Data frame, data set with return from GALILEO
`mun`	character, the name of the municipio.
`match`	character, the colum MatchAdress from GALILEO.
`uf`	character, the name of the state.

Value

Returns a new column called munmatch with true or false.

nome_de_solteira *Return women single's name*

Description

nome_de_solteira Return women single's name using the husband last name.

Usage

```
nome_de_solteira(nome_casada, nome_conjuge)
```

Arguments

nome_casada	Character, married woman's name
nome_conjuge	character, husband's name.

Value

Returns a list of possible names

Examples

```
nome_de_solteira(nome_casada = "Maria Conceicao da Costa", nome_conjuge = "Mario Silva da Costa")
```

remove_preposicao_nomes *Remove de da e dos of names .*

Description

remove_preposicao_nomes return names without de, da e dos.

Usage

```
remove_preposicao_nomes(base, ..., suffixo = "_semD")
```

Arguments

base	A data table, data frame or character vector.
...	columns for apply the function
suffixo	Suffix name for the new column.

Value

the base parameter with a new column.

Examples

```
base <- data.frame(nome = c("João das Neves", "Pedro dos Anjos", "Maria das Gracas"))
base <- remove_preposicao_nomes(base, "nome")
```

Index