Package 'utilsIPEA'

October 12, 2022

Type Package

Title IPEA Common Functions

Version 0.0.6

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

NeedsCompilation no

Author Gustavo Coelho [aut, cre],
Lucas Mation [aut],
Daniel Lima [ctb],
Igor Noberto [ctb],
João Victor Machado [ctb]

Repository CRAN

Date/Publication 2018-04-20 19:51:54 UTC

R topics documented:

Index   **6**

---

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")
```

---

contar_letras   *Remove de da e dos of names .*

Description

contar_letras return number of any letters on alphabet for each column.

Usage

```
contar_letras(base, columns, suffix = NULL)
```

Arguments

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

Value

a data.table number of any letters on alphabet for each column

Examples

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

---

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 mun-*
match 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 column 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