

Package: herramientas (via r-universe)

November 17, 2024

Title Conjunto de funciones para el procesamiento de datos en la DNMyE

Version 0.0.0.9013

Description Caja de herramientas para el procesamiento de datos en la Dirección Nacional de Mercados y Estadísticas del Ministerio de Turismo y Deportes de la Nación.

License GPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Imports mgsub, stringr, dplyr, RCurl, haven, glue, readr, usethis, collapsibleTree, tidyr, readxl, openxlsx, arrow, sf, assertthat, tibble, textclean

Depends R (>= 2.10)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

URL <https://dnme-minturdep.github.io/herramientas/>

Config/pak/sysreqs cmake libgdal-dev gdal-bin libgeos-dev git make libgit2-dev libicu-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev libx11-dev zlib1g-dev

Repository <https://tuqmano.r-universe.dev>

RemoteUrl <https://github.com/dnme-minturdep/herramientas>

RemoteRef HEAD

RemoteSha 0c7c9c8292c7ca98505c9e5dbf2f4d2cfe4cc304

Contents

archivo_plano_srv	2
crear_carpetas	2
crear_proyecto	3
etiquetar_provincia	3

herramientas	4
limpiar_texto	4
ls_srv	4
numeros_en_palabras	5
read_file_srv	5
read_rds_srv	6
read_sav_srv	6
remover_tildes	6
set_user_srv	7
sync_proyecto_git	7
tasa_variacion	8
view_tree_srv	8
write_file_srv	9
%out%	9

Index **10**

archivo_plano_srv *Leer archivos planos del server:*

Description

Hace una llamada getUrl a la ruta indicada dentro de /srv/DataDNMYE/. Usarlo con read_csv o read_csv2 u otros.

Usage

archivo_plano_srv(ruta)

Arguments

ruta Texto con la ruta específica del archivo a leer (ej.: "aerocomercial/anac/tabla_final.txt")

crear_carpetas *skeleton de proyecto*

Description

arma estructura de carpetas para proyecto

Usage

crear_carpetas()

crear_proyecto	<i>Crear esqueleto de un nuevo proyecto de R</i>
----------------	--

Description

Crear esqueleto de un nuevo proyecto de R

Usage

```
crear_proyecto(ruta = NULL)
```

Arguments

ruta	nombre de la carpeta y proyecto que se quiere crear
------	---

Value

directorio base para un nuevo proyecto

etiquetar_provincia	<i>Etiquetado de variable provincias:</i>
---------------------	---

Description

Genera una columna con el nombre oficial de la provincia y otra columna, denominada "provincia_codigo", con su código numérico INDEC (dos posiciones)

Usage

```
etiquetar_provincia(base, id_col, new_name = "provincia_nombre", add_codes = T)
```

Arguments

base	Dataframe con una variable que refiere a las provincias
id_col	Columna con el nombre de la provincia o su código INDEC
new_name	Define el nuevo nombre de la columna con las etiquetas de provincia. Por default es "provincia_nombre"
add_codes	Indica si se desea agregar la columna con el código de la provincia. Por default es TRUE

herramientas	herramientas <i>package</i>
--------------	-----------------------------

Description

Paquete de R con utilidades para "domar los datos" de la DNMYE See the README on [Github](#)

limpiar_texto	<i>Limpieza general de texto:</i>
---------------	-----------------------------------

Description

Aplica tolower(), str_replace_all() para quitar caracteres especiales y remover_tildes(). También aplica opcionalmente str_replace_all() convertir "ñ" en "n" (por default se aplica). Ej.:

Usage

```
limpiar_texto(texto, enie = TRUE)
```

Arguments

texto	variable de clase 'character' a la que se le quiere aplicar la función
enie	valor lógico (TRUE o FALSE) indicando si reemplazar "ñ" por "n"

Examples

```
limpiar_texto("TéxTÔ con una Ñ?!") #"texto con una n"
limpiar_texto("TéxTÔ con una Ñ?!", enie = FALSE) #"texto con una ñ"
```

ls_srv	<i>Listar carpetas y archivos del server:</i>
--------	---

Description

Devuelve un vector con las ubicaciones en la ruta indicada dentro de /srv/DataDNMYE/.

Usage

```
ls_srv(ruta = NULL, full_names = F)
```

Arguments

ruta	Texto con la ruta específica (ej.: "evyth/microdatos/").
full_names	Lógico Indica si devolver la ruta entera del archivo o directorio o solo el nombre del elemento.

numeros_en_palabras *Limpieza de digitos en texto:*

Description

Transforma digitos del 0 al 100 en palabras. Ej.: "1" en "uno", "25" en "veinticinco".

Usage

```
numeros_en_palabras(texto, intervalo = 0:100)
```

Arguments

texto	variable de clase 'character' a la que se le quiere aplicar la función
intervalo	rango entre 0 y 100 con los numeros que se desean reemplazar. Ej. intervalo = 37:50

Examples

```
numeros_en_palabras("25 de mayo") #"veinticinco de mayo"
numeros_en_palabras("6 de abril del 98") #"seis de abril del noventa y ocho"
```

read_file_srv *Leer archivos del server:*

Description

Hace una conexión a la ruta indicada dentro de /srv/DataDNMYE/ y lee el archivo con una función acorde a la extensión del mismo.

Usage

```
read_file_srv(ruta, ...)
```

Arguments

ruta	Texto con la ruta específica del archivo a leer con su extensión (acepta los siguientes formatos: "rds", "csv", "sav", "txt", ".parquet", "xlsx", "xls", "gpkg", "geojson" y "kml"). Ej.: "aerocomercial/anac/base_anac_agrupada.rds".
...	Parámetros para pasarle a la función de lectura subyacente: <ul style="list-style-type: none"> • csv/txt: readr::read_delim • rds: base::readRDS • sav: readr::read_sav • xlsx/xls: readxl::read_excel • parquet: arrow::read_parquet • gpkg/geojson/kml: sf::read_sf

read_rds_srv *Leer archivos rds del server:*

Description

Hace una conexión a la ruta indicada dentro de /srv/DataDNMYE/ y lee el archivo RDS con read-RDS().

Usage

```
read_rds_srv(ruta)
```

Arguments

ruta Texto con la ruta específica del archivo a leer (ej.: "aerocomercial/anac/base_anac_agrupada.rds")

read_sav_srv *Leer archivos sav del server:*

Description

Hace una conexión a la ruta indicada dentro de /srv/DataDNMYE/ y lee el archivo .sav con haven::read_sav().

Usage

```
read_sav_srv(ruta)
```

Arguments

ruta Texto con la ruta específica del archivo a leer (ej.: "evyth/microdatos/evyth_microdatos.sav")

remover_tildes *Limpieza general de tildes:*

Description

Elimina tildes y diéresis de las vocales y la letra "y". Ej.: "ÿÚðâ" en "yUoa".

Usage

```
remover_tildes(texto)
```

Arguments

texto variable de clase 'character' a la que se le quiere aplicar la función

Examples

```
remover_tildes("ÿÜòâ") #"yuoâ"
```

```
set_user_srv
```

Configurar usuario del server de dnmye:

Description

Guia para que el usuario configure sus credenciales en .Renviron".

Usage

```
set_user_srv()
```

```
sync_proyecto_git
```

Sincronizar proyecto con Github

Description

Vincula proyecto de R local con repositorio de organización en Github.

Usage

```
sync_proyecto_git(repo = "d4t4tur", privado = TRUE, protocolo = "ssh")
```

Arguments

repo	Por defecto estos se crean en repositorio DEP ('d4t4tur'). Se puede pasar 'repo = "dnme-minturdep"' como argumento alternativo para vincular con repositorio PROD
privado	Por defecto el repositorio será privado. Se puede pasar como argumento 'privado = FALSE' para que sea público'
protocolo	Por defecto el repositorio usa protocolo de conexión 'ssh'. Se puede pasar como argumento 'protocolo = "https"' como alternativa

Value

Vínculo origin/remote para un repositorio .git sincronizado con Organización en Github

tasa_variacion	<i>Tasa de variación:</i>
----------------	---------------------------

Description

Calcula la tasa de variación con respecto al período previo según se defina, aplicando `dplyr::lag()`

Usage

```
tasa_variacion(
  variable,
  periodo = "anual",
  porcentaje = FALSE,
  presentacion = FALSE
)
```

Arguments

variable	variable numérica para calcular la variación
periodo	indicar cuántos períodos para atrás se calcula. Por defecto es anual (1 período), puede elegirse "mensual" o "trimestral" o directamente el número (12 para mensual por ej.)
porcentaje	variación en términos porcentuales (multiplica por 100)
presentacion	resultado en formato character con separador de decimales y miles usuales <code>tasa_variacion(c(10,10,10,10))</code> <code>tasa_variacion(c(10,10,10,10), periodo = trimestral)</code>

view_tree_srv	<i>Diagrama de carpetas y archivos del server:</i>
---------------	--

Description

Devuelve un gráfico interactivo con los directorios y archivos en `/srv/DataDNMYE/`. El último nodo contiene la ruta completa de la carpeta o archivo para poder copiar y pegar.

Usage

```
view_tree_srv()
```

write_file_srv *Leer archivos rds del server:*

Description

Hace una conexión a la ruta indicada dentro de /srv/DataDNMYE/ y escribe un archivo CSV, RDS, XLSX, TXT, PARQUET o SAV.

Usage

```
write_file_srv(x, ruta, ...)
```

Arguments

x	Objeto a escribir (ej: data.frame)
ruta	Texto con la ruta del archivo, incluyendo nombre y extensión del mismo (acepta ".csv", ".rds", ".xlsx", ".txt", ".parquet", ".sav", "gpkg", "geojson" y "kml"). Ej: "aerocomercial/anac/base_final.csv")
...	Parametros para pasarle a la funcion de escritura subyacente: <ul style="list-style-type: none"> • csv: readr::write_csv • txt: utils::write.table • rds: readr::write_rds • sav: haven::write_sav • xlsx: openxlsx::write.xlsx • parquet: arrow::write_parquet • gpkg/geojson/kml: sf::write_sf

%out% *Negación de %in% :*

Description

elementos no contenidos dentro de un conjunto

Usage

```
a %out% b
```

Arguments

a	vector con conjunto de datos
b	vetor con conjunto de datos 2

Index

`%out%`, 9

`archivo_plano_srv`, 2

`crear_carpetas`, 2

`crear_proyecto`, 3

`etiquetar_provincia`, 3

`herramientas`, 4

`limpiar_texto`, 4

`ls_srv`, 4

`numeros_en_palabras`, 5

`read_file_srv`, 5

`read_rds_srv`, 6

`read_sav_srv`, 6

`remover_tildes`, 6

`set_user_srv`, 7

`sync_proyecto_git`, 7

`tasa_variacion`, 8

`view_tree_srv`, 8

`write_file_srv`, 9