

Package ‘edm1’

July 19, 2024

Title Simplify Complex Data Manipulation

Version 2.0.0.0

Description Provides complex sorting algorithms. Provides date manipulation algorithms. In addition to providing handy functions to discretize variables, an SQL joins alternatives, a set of function to work with geographical coordinates, and other functions to work with text mining.

License GPL (==3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Imports stringr,
stringi,
dplyr,
openxlsx

Contents

col_to_row	2
just_anything	2
just_anything2	3
just_anything3	3
just_chr	4
just_chr2	4
just_chr3	5
just_nb	5
just_nb2	6
just_nb3	6
just_not_anything	7
just_not_anything2	7
just_not_anything3	8
row_to_col	8

Index	10
--------------	-----------

col_to_row	<i>col_to_row</i>
------------	-------------------

Description

Allow to reverse a dataframe (cols become rows and rows become cols)

Usage

```
col_to_row(inpt_datf)
```

Arguments

`inpt_datf` is the inout dataframe

Examples

```
datf_test <- data.frame(c(1:11), c(11:1))

print(col_to_row(inpt_datf = datf_test))

   X1 X2 X3 X4 X5 X6 X7 X8 X9 X10 X11
1   1  2  3  4  5  6  7  8  9  10  11
2  11 10  9  8  7  6  5  4  3  2   1
```

just_anything	<i>just_anything</i>
---------------	----------------------

Description

Extract only the letters from all elements of a vector, see examples

Usage

```
just_anything(inpt_v, symbol_ = "-", anything_v = c())
```

Arguments

`inpt_v` is the input vector
`symbol_` is the chosen symbol to replace numbers

Examples

```
print(just_anything(inpt_v = c("oui222jj644", "oui122jj"),
  symbol_ = "-", anything_v = letters))

[1] "oui-jj-" "oui-jj"
```

just_anything2	<i>just_anything2</i>
----------------	-----------------------

Description

Extract only the letters from all elements of a vector, see examples

Usage

```
just_anything2(inpt_v, symbol_ = "-", anything_v = c())
```

Arguments

inpt_v	is the input vector
symbol_	is the chosen symbol to replace numbers

Examples

```
print(just_anything2(inpt_v = c("oui222jj44", "oui122jj"),
  symbol_ = "-", anything_v = letters))

[1] "oui---jj--" "oui---jj"
```

just_anything3	<i>just_anything3</i>
----------------	-----------------------

Description

Extract only the letters from all elements of a vector, see examples

Usage

```
just_anything3(inpt_v, anything_v = c())
```

Arguments

inpt_v	is the input vector
--------	---------------------

Examples

```
print(just_anything3(inpt_v = c("oui222jj644", "oui122jj"),
  anything_v = letters))

[1] "ouiijj" "ouiijj"
```

just_chr	<i>just_chr</i>
----------	-----------------

Description

Extract only the letters from all elements of a vector, see examples

Usage

```
just_chr(inpt_v, symbol_ = "-")
```

Arguments

inpt_v	is the input vector
symbol_	is the chosen symbol to replace numbers

Examples

```
print(just_chr(inpt_v = c("oui222jj644", "oui122jj"),
  symbol_ = "-"))

[1] "oui-jj-" "oui-jj"
```

just_chr2	<i>just_chr2</i>
-----------	------------------

Description

Extract only the letters from all elements of a vector, see examples

Usage

```
just_chr2(inpt_v, symbol_ = "-")
```

Arguments

inpt_v	is the input vector
symbol_	is the chosen symbol to replace numbers

Examples

```
print(just_chr2(inpt_v = c("oui222jj44", "oui122jj"),
  symbol_ = "-"))

[1] "oui---jj--" "oui---jj"
```

just_chr3

just_chr3

Description

Extract only the letters from all elements of a vector, see examples

Usage

```
just_chr3(inpt_v)
```

Arguments

`inpt_v` is the input vector

Examples

```
print(just_chr3(inpt_v = c("oui222jj644", "oui122jj")))
[1] "oui jj" "oui jj"
```

just_nb

just_nb

Description

Extract only the letters from all elements of a vector, see examples

Usage

```
just_nb(inpt_v, symbol_ = "-")
```

Arguments

`inpt_v` is the input vector

`symbol_` is the chosen symbol to replace numbers

Examples

```
print(just_nb(inpt_v = c("oui222jj644", "oui122jj"),
  symbol_ = "-"))
[1] "-222-44" "-122-"
```

just_nb2

just_nb2

Description

Extract only the letters from all elements of a vector, see examples

Usage

```
just_nb2(inpt_v, symbol_ = "-")
```

Arguments

inpt_v is the input vector

symbol_ is the chosen symbol to replace numbers

Examples

```
print(just_nb2(inpt_v = c("oui222jj44", "oui122jj"),
  symbol_ = "-"))

[1] "---222--44" "---122--"
```

just_nb3

just_nb3

Description

Extract only the letters from all elements of a vector, see examples

Usage

```
just_nb3(inpt_v)
```

Arguments

inpt_v is the input vector

Examples

```
print(just_nb3(inpt_v = c("oui222jj644", "oui122jj")))

[1] 222644 122
```

```
just_not_anything just_not_anything
```

Description

Extract only the letters from all elements of a vector, see examples

Usage

```
just_not_anything(inpt_v, symbol_ = "-", anything_v = c())
```

Arguments

inpt_v	is the input vector
symbol_	is the chosen symbol to replace numbers

Examples

```
print(just_not_anything(inpt_v = c("oui222jj644", "oui122jj"),
  symbol_ = "-", anything_v = letters))

[1] "-222-644" "-122-"
```

```
just_not_anything2 just_not_anything2
```

Description

Extract only the letters from all elements of a vector, see examples

Usage

```
just_not_anything2(inpt_v, symbol_ = "-", anything_v = c())
```

Arguments

inpt_v	is the input vector
symbol_	is the chosen symbol to replace numbers

Examples

```
print(just_not_anything2(inpt_v = c("oui222jj44", "oui122jj"),
  symbol_ = "-", anything_v = letters))

[1] "---222-44" "---122--"
```

```
just_not_anything3 just_not_anything3
```

Description

Extract only the letters from all elements of a vector, see examples

Usage

```
just_not_anything3(inpt_v, anything_v = c())
```

Arguments

`inpt_v` is the input vector

Examples

```
print(just_not_anything3(inpt_v = c("oui222jj644", "oui122jj"),
  anything_v = letters))

[1] "222644" "122"
```

```
row_to_col row_to_col
```

Description

Allow to reverse a dataframe (rows become cols and cols become rows)

Usage

```
row_to_col(inpt_datf)
```

Arguments

`inpt_datf` is the inout dataframe

Examples

```
datf_test <- data.frame(c(1, 11), c(2, 10), c(3, 9), c(4, 8))

print(datf_test)

  c.1..11. c.2..10. c.3..9. c.4..8.
1         1         2         3         4
2        11        10         9         8

print(row_to_col(inpt_datf = datf_test))

1 2
```



```
1 1 11
2 2 10
3 3 9
4 4 8
```

Index

`col_to_row`, [2](#)

`just_anything`, [2](#)

`just_anything2`, [3](#)

`just_anything3`, [3](#)

`just_chr`, [4](#)

`just_chr2`, [4](#)

`just_chr3`, [5](#)

`just_nb`, [5](#)

`just_nb2`, [6](#)

`just_nb3`, [6](#)

`just_not_anything`, [7](#)

`just_not_anything2`, [7](#)

`just_not_anything3`, [8](#)

`row_to_col`, [8](#)