'tidbitR'

May 4, 2017

`CopyFromClipboard` *Copies from clipboard*

Description

Copies a tsv table from clipboard.

Usage

```
CopyFromClipboard(sep = "\t", quote = "\"", stringsAsFactors = F,
  header = T, ...)
```

Arguments

`sep`	Column separator. Defaulted to tab.
`quote`	Informs the quote character.
`stringsAsFactors`	Informs if string has to be treated as factors.
`header`	Informs if a header is the first row.
`...`	Other parameters.

Value

Gets clipboard.

Examples

```
CopyFromClipboard()
ds <- CopyFromClipboard(quote = "'")
```

CopyToClipboard *Copies to clipboard*

Description

Copies an object to clipboard as a tsv table.

Usage

```
CopyToClipboard(x, sep = "\t", quote = T, na = "", row.names = F,
  col.names = T, ...)
```

Arguments

x	tbl to copy.
sep	Column separator. Defaulted to tab.
quote	T (default) informs that the result will be within quotes.
na	Value for NAs. Default is an empty string.
row.names	Informs if row names must be included.
col.names	Informs if column names must be included.
...	Other parameters.

Value

Copies the object to the clipboard.

Examples

```
CopyToClipboard(census)
census %>%
  CopyToClipboard()
```

Freq *Frequency table of all columns*

Description

Generates a frequency distribution table of all columns in a data set.

Usage

```
Freq(x, records = 10, as = "l")
```

Arguments

x	Dataset to be processed.
records	Number of rows to show in the resulting table
as	return as "l" list or "d" dataset

Value

Returns a frequency table in a key/value pair format for all the columns. Number of columns returned is length of the dataset x 2.

Examples

```
Freq(census)
Freq(census)$State
Freq(census, 50, "d")
Freq(census, 100, "d") #will throw an error, use Freq(census, 100, "l") instead
```

`GetXMLNodes` *Gets all XML node text*

Description

Parses through all XML end nodes and returns the text of those nodes in a tibble.

Usage

```
GetXMLNodes(x, nodepath = "*")
```

Arguments

`x`	XML document.
`nodepath`	Xpath to node for which text has to be collected. Defaults to * that will collect text of all nodes.

Value

Returns xpath to a node, node name and text value in a tibble.

Examples

```
GetXMLNodes(xml2::read_xml("https://www.w3schools.com/xml/simple.xml")) %>%
  View()
GetXMLNodes(xml2::read_xml("https://www.w3schools.com/xml/simple.xml"), "calories") %>%
  View()
```

%=% *Compares two datasets*

Description

Merges and compares two datasets. #'

Usage

```
x %=% y
```

Arguments

x	Dataset to compare.
y	Dataset to compare with,

Value

Returns a dataset after merging and two column that denotes which of the merged rows exists in x or y or both.

Examples

```
x %=% y
```

Recon *Compare two dataframe to show differences highlighted*

Description

Compares two data frames to show two additional columns lhs_matches, rhs_matches to inform the number of rows that matches against x and y. These columns informs if the row is present in x or y or both. Both the data frames must have the same set of columns.

Usage

```
Recon(x, y, col.names = c("lhs_matches", "rhs_matches"),
  check_duplicates_of = 1)
```

Arguments

x	First data frame to compare.
y	Second data frame to compare with.
col.names	Informs the title of the new columns informs if matches exists. Defaulted to lhs_matches and rhs_matches.
check_duplicates_of	Informs the primary key columnindex for which duplicates has to be checked. The result will be listed in a column named duplicates.

Value

Returns a tibble with the difference.

Examples

```
Recon(list1, list2)
```

Stat *Shows a quick summary of a data frame*

Description

Shows element type, row count, na row count, unique row count, mean, min, Q1, median, Q3, max, min string length, max string length and set of samples from each of the columns in a data frame.

Usage

```
Stat(df, sample = 20, sample_delim = "; ")
```

Arguments

df	Data frame.
sample	Number of samples from each column. Defaults to 20.
sample_delim	Delimiter for each sample.

Value

Stat(census)

StringToDate *Converts an date string to a date*

Description

Converts a string in date format(dd-MMM-yy) into a R Date.

Usage

```
StringToDate(x, century = F)
```

Arguments

x	Date string to be converted.
century	Informs if the string has the century included (dd-MM-yyyy). Default is F.

Value

Returns the date string as a date

Examples

```
StringToDate("01-JAN-16")
StringToDate("01-JAN-2017", T)
```

Index