Package 'miscset'

February 24, 2017

Type Package

Title Miscellaneous Tools Set

Version 1.1.0

Date 2017-02-24

Author Sven E. Templer

Maintainer Sven E. Templer <sven.templer@gmail.com>

Description A collection of miscellaneous methods to simplify various tasks, including plotting, data.frame and matrix transformations, environment functions, regular expression methods, and string and logical operations, as well as numerical and statistical tools. Most of the methods are simple but useful wrappers of common base R functions, which extend S3 generics or provide default values for important parameters.

License GPL-3

Imports data.table, devtools, ggplot2, gridExtra, parallel, Rcpp (>= 0.11.1), stats, xtable

LinkingTo Rcpp

Suggests knitr, rmarkdown, stringr

VignetteBuilder knitr

URL https://github.com/setempler/miscset

BugReports https://github.com/setempler/miscset/issues

RoxygenNote 6.0.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2017-02-24 16:46:57

R topics documented:

`miscset-package` *Miscellaneous R Tools*

Description

A collection of miscellaneous methods to simplify various tasks, including plotting, data.frame and matrix transformations, environment functions, regular expression methods, and string and logical operations, as well as numerical and statistical tools. Most of the methods are simple but useful wrappers of common base R functions, which extend S3 generics or provide default values for important parameters.

Details

The package vignette provides a comprehensive overview and examples for the usage of all available functions in the package. View with `vignette("miscset")`.

Author(s)

Sven E. Templer

`ciplot` *Barplot with Confindence Intervals*

Description

Create barplots of a list of numeric values and error bars according to the confidence interval, standard deviation, interquartile range, etc.

Usage

```
ciplot(x, ...)

## Default S3 method:
ciplot(x, ..., ylim, height.fun = mean,
  height.args = list(), error.fun = confint, error.args = list(),
  arrows.args = list(code = 3, angle = 90), na.rm = TRUE)
```

Arguments

`x`	List of numeric values
`...`	Arguments forwarded to barplot in default method.
`ylim`	A range for the y-axis limits.
`height.fun`	Function to apply on each list object to calculate the height of the bars from.
`height.args`	Arguments forwarded to height.fun, as a named list.
`error.fun`	Function to calculate the error size. See also details.
`error.args`	Arguments forwarded to error.fun, as a named list.
`arrows.args`	Arguments forwarded to arrows, as a named list.
`na.rm`	Logical, remove missing values.

Details

Example for quantiles:

```
interquartile <- function(x) {quartile(x,.75)-mean(x)}
quantileQ <- function(x, q) {abs(quartile(x,q[1])-mean(x))}
```

Author(s)

Sven E. Templer

collapse *Collapse objects*

Description

Collapse objects as in the paste function.

Usage

```
collapse(x, sep, ...)

## Default S3 method:
collapse(x, sep = "", ..., .unique = FALSE,
  .sort = FALSE, .decreasing = FALSE)

## S3 method for class 'data.frame'
collapse(x, sep = "", by = names(x), ...,
  .unique = FALSE, .sort = FALSE, .decreasing = FALSE, .unlist = FALSE,
  .sortby = FALSE)
```

Arguments

x	Any R object.
sep	A character string to separate value columns. NULL retains a vector.
...	Forwarded to or from other methods.
.unique	Logical, return only unique values.
.sort	Logical, sort the values.
.decreasing	Logical, if sorting, then by decreasing values.
by	Column names to split data frame by, before applying collapse on each remaining column within each piece. Using the default (all columns), then unique(x) is returned. Columns can be specified by names or integer with the column numbers. Using 0 or NULL collapses all columns.
.unlist	Logical, if value columns need to be unlisted before collapsing.
.sortby	Logical, sort the output on the by columns. This applies , If x was a data.table, then the keys are set as the by values.

Details

For the data.frame method, x is converted to a data.table before applying the piece- and column-wise collapses. If the input is already inheriting from data.table, then the class is retained. .sortby is causing setkeyv(x, by) to be applied to x after converting to a data.table.

Author(s)

Sven E. Templer

Examples

```
#

### some data

set.seed(12)
s <- s2 <- sample(LETTERS[1:4], 9, replace = TRUE)
s2[1:2] <- rev(s2[1:2])
d <- data.frame(group = rep(letters[c(3,1,2)], each = 3),
                value = s,
                level = factor(s2),
                stringsAsFactors = FALSE)

### collapse vectors

collapse(letters)
collapse(1:3)               # coerced to character
collapse(LETTERS[1:5], "-") # separated by '-'

### collapse data.frames

# by all columns (same as unique)
collapse(d)
# by a grouping column
collapse(d, by = 1)
# by multiple, but not all columns
collapse(d, by = c("group", "value"))
# return single row
collapse(d, by = 0)
# return single row, unique and sorted values
collapse(d, by = 0, .unique = TRUE, .sort = TRUE)

#
```

`confint` *Confidence Intervals for Numeric Vectors*

Description

Calculate confidence intervals for values of a numeric vector.

Usage

```
## S3 method for class 'numeric'
confint(object, parm = qnorm, level = 0.95, ...,
  na.rm = TRUE, ret.attr = TRUE)
```

Arguments

object	A numeric vector.
parm	Function for quantile calculation. e.g. qnorm, qt
level	Size of confidence (0 < size < 1).
...	Unused.
na.rm	Logical, remove missing values for sd and mean.
ret.attr	Logical, to include the mean value and function arguments as attributes of the returned object.

Value

Returns a numeric vector with the lower and upper range of the confidence interval.

Author(s)

Sven E. Templer

Examples

```
#

confint(1:3)
confint(1:3, ret.attr = FALSE)

#
```

---

do.rbind *Bind data.frames in a List by Rows*

Description

Same as do.call(rbind, x), but adding a column with the name of each table. Missing names are replaced by integers.

Usage

```
do.rbind(x, idcol = "Name", keep.rownames = FALSE)
```

Arguments

x	List with data.frames. Non data.frame objects are dropped.
idcol	Name for column with ids in output.
keep.rownames	Logical, keep rownames.

Value

Returns a data.frame

Author(s)

Sven E. Templer

`duplicates` *Determine Duplicates*

Description

Determine duplicates. `duplicates` returns a logical vector, `duplicatei` an integer vector.

Usage

```
duplicates(x)

duplicatei(x, first = TRUE)
```

Arguments

`x`	A vector or data.frame to search for duplicates.
`first`	Logical, `TRUE` to return the index also for the first occurrence of values. Otherwise, a `0` is the index for the first occurrence.

Value

`duplicates` returns a logical vector as duplicated, but with `TRUE` values also for the first occurrence of duplicated values.
`duplicatei` returns the index of the first occurrence of each unique value.

Author(s)

Sven E. Templer

Examples

```
#

x <- c(7, 7, 7, 2, 3, 2)
data.frame(
  data = x,
  duplicated = duplicated(x),
  duplicates = duplicates(x),
  duplicatei = duplicatei(x),
  duplicatei0 = duplicatei(x, FALSE))

#
```

enpaire *Create a Pairwise List from a Matrix*

Description

Transform a `matrix` or `dist` object to a pairwise list.

Usage

```
enpaire(x, ...)

## Default S3 method:
enpaire(x, ...)

## S3 method for class 'dist'
enpaire(x, upper = T, lower = T, ...)

## S3 method for class 'matrix'
enpaire(x, upper = T, lower = T, ...)
```

Arguments

`x`	Object of class `matrix`.
`...`	Arguments passed to methods.
`upper`	Logical, return values from upper triangle.
`lower`	Logical, return values from lower triangle.

Value

Returns a `data.frame`. The first and second column represent the dimension names for a value in `x`. The following columns contain the values for the upper or lower triangle.

Author(s)

Sven E. Templer

See Also

squarematrix

Examples

```
#

m <- matrix(letters[1:9], 3, 3, dimnames = list(1:3,1:3))
enpaire(m)
enpaire(m, lower = FALSE)
```

```
#
```

`factorNA` *Create a Factor with NA as Level*

Description

Create a factor with `NA` values included and positioned as last level.

Usage

```
factorNA(x, ...)
```

Arguments

`x`	A vector coerced to character.
`...`	Forwarded to factor. `x` and `levels` are defined.

Author(s)

Sven E. Templer

`gghcl` *HTML Colours Like ggplot2*

Description

Calculate HTML colour code from a palette like ggplot2 uses.

Usage

```
gghcl(n, sub = 1:n, h = c(0, 360) + 15, c = 100, l = 65, ...)
```

Arguments

`n`	Numeric value to determine size of palette.
`sub`	Numeric vector with values within range from `1` to `n` to subset palette.
`h`	Hue of the colour. Within range of a circle's degrees.
`c`	Chroma of the colour.
`l`	Luminance of the colour. Within range from `1` to `100`.
`...`	Further arguments passed to function `hcl`.

Details

See ?hcl for explanation of h, c and l.

Value

Returns a character vector containing HTML colour code of the standard ggplot colour palette.

Author(s)

Sven E. Templer

See Also

hcl

Examples

```
#

# Plot some palettes:
par(mfrow = c(3,1), mai = c(.1,.1,1,.1))
p <- matrix(1:10, 10)
image(p, col = gghcl(5), axes = FALSE, main ="gghcl(5)")
image(p, col = gghcl(10), axes = FALSE, main = "gghcl(10)")
image(p, col = gghcl(10, 1:5), axes = FALSE, main ="gghcl(10, 1:5)")
# dev.off() # to reset \code{par}

#
```

---

ggplotGrid *Arrange a List of ggplots*

Description

Arrange a list of ggplots with grid.arrange and output on local graphic device or as pdf/png when a path is supplied. ggplotGridA4 writes the plots to a DIN A4 (8 x 11 inches) pdf file directly.

Usage

```
ggplotGrid(l, path, ncol = 1, nrow = 1, width = 8, height = 11,
  res = 300, pdf.cairo = TRUE, onefile = TRUE, ...)

ggplotGridA4(l, path, ncol = 2, nrow = 1, wide = TRUE)

ggplotlist(x, ncol = 1, path, width = 11, height = 8)
```

Arguments

l	List with ggplot objects.
path	Plot to file of type pdf or png. Determine type by path ending (.pdf or .png). Optional in ggplotlist: A character string that gives the path to export the plot to a file, ending with 'pdf' or 'png' (case insensitive). If missing, then the grid is returned to the current graphic device.
ncol	Number of columns.
nrow	Number of rows per page, only for pdfs.
width	For pdfs/pngs the width in inches, else ignored.
height	For pdfs/pngs the height in inches, else ignored.
res	Resolution in dpi for pngs.
pdf.cairo	Use cairo_pdf (or cairo_ps, svg) instead of pdf
onefile	Create one file, see cairo_pdf.
...	Forwarded to cairo_pdf
wide	Wide format pdf pages (11x8 inches).
x	A list containing at least one ggplot object of class gg.

Author(s)

Sven E. Templer

Examples

```
#

## Not run:
library(ggplot2)
d <- data.frame(a=1:5,b=1:5)
x <- list(
  ggplot(d, aes(x=a,y=b,col=b)) + geom_line(),
  ggplot(d, aes(x=a,y=b,shape=factor(b))) + geom_point())
ggplotlist(x, 2)
## End(Not run)

#
```

---

gregexprind *Pattern Matching and Extraction*

Description

Function to extract a certain index from gregexpr().

Usage

```
gregexprind(pattern, text, n, ...)
```

Arguments

pattern	Character string containing a regular expression to be searched in text.
text	Character vector where the search is performed.
n	Numeric value or character string "last" to extract nth or last position of pattern in each value of text.
...	Arguments passed to function gregexpr().

Value

Numeric vector of length length(text).

Author(s)

Sven E. Templer

See Also

See gregexpr for further information on arguments.
See regex for the use of regular expressions.

Examples

```
#

gregexprind(c("a"),c("ababa","ab","xyz",NA), 1)
gregexprind(c("a"),c("ababa","ab","xyz",NA), 2)
gregexprind(c("a"),c("ababa","ab","xyz",NA), "last")

#
```

---

help.index *Open The Package Help Index Page*

Description

Given a package name or string, start the package help index page in a browser.

Usage

```
help.index(pkg, browser = NULL)
```

Arguments

pkg	A character string or expression with the name of a package.
browser	The browser to display. text and pdf don't use a browser, but builtin text/pdf (help_type). Otherwise a character string for the browser program binary to call or function.

Author(s)

Sven E. Templer

---

info *Print enhanced session information*

Description

Based on and enhancing devtools::session_info.

Usage

```
info(..., width = 120)
```

Arguments

...	Forwarded to other methods.
width	Console width in columns.

Author(s)

Sven E. Templer

See Also

session_info

Examples

```
info()
devtools::session_info()
sessionInfo()
```

leading0 *Numeric to Character with Leading Zero(s)*

Description

Transform numeric values to character string prepending leading zero(s).

Usage

```
leading0(num, digits = 2)
```

Arguments

num	Numeric vector (character also possible) to transform.
digits	Numeric value of minimum length of output strings.

Value

Character vector with same length of strings of each value. Original "string" is prepended by zero(s). String length is at least `max(nchar(as.character(num)))`.

Author(s)

Sven E. Templer <sven.templer@gmail.com>

Examples

```
#

# use with paste to generate strings of equal size:
paste0("observation", leading0(1:10, 3))

#
```

lload *Load RData Objects to a List*

Description

Load multiple .RData files and return a (simplified) list.

Usage

```
lload(path = ".", pattern = ".RData", recursive = FALSE,
  simplify = TRUE, verbose = TRUE)
```

Arguments

path	Character string with the path, as used in list.files.
pattern	A regular expression for file name patterns, as used in list.files.
recursive	Logical. Search the path recursive.
simplify	Logical, unlist when there are only unique object names.
verbose	Logical. Print information on screen about loading process.

Value

Returns a list of length n, when there are n data files loaded. All objects are stored in sublists. Names are according to files, and names of sublists to objects per file. If simplified, the list is of length m, when there are m objects in total loaded.

Author(s)

Sven E. Templer

See Also

load

---

lsall *List Object Details*

Description

Return a data.frame with a list of all objects of a specified environmet.

Usage

```
lsall(envir = .GlobalEnv, ...)
```

Arguments

envir	An environment where to look for objects.
...	Arguments forwarded to ls.

Value

Returns a data.frame with object names, lengths, classes, modes and sizes or NULL if the environment is empty.

Author(s)

Sven E. Templer

See Also

ls

Examples

```
#

lsall()
obj1 <- 1:3
obj2 <- data.frame(1:3)
obj3 <- list(1:3)
lsall()

#
```

`mgrepl` *Multiple Pattern Matching and Replacement*

Description

`mgrepl` allows multiple patterns search in character vectors, offering multicore support to parallelize search over all `patterns` using mclapply.

Usage

```
mgrepl(patterns, text, log.fun = all, na.replace = FALSE,
  use.which = FALSE, cores = 1, ...)
```

Arguments

`patterns`	A vector or list containing regular expressions (regex) to be searched in `text`. Coerced to `character`.
`text`	Character vector on which the search is performed.
`log.fun`	A function to apply on the result of matching each pattern on each element of `text`. Determines the output. See section **Value**.
`na.replace`	A single value to replace each `NA` with in the result.
`use.which`	A logical value. `TRUE` to convert result with which. Only if output is.atomic, otherwise ignored. Deprecated.
`cores`	Numeric value for how many cores to use for computation using `mclapply`.
`...`	Further arguments passed to functions grepl.

Value

Depending on the function defined with `log.fun`, the return value is either

- a `vector`, e.g. for functions like any, all or sum.
- a `matrix` is obtained with e.g. identity or as.integer. Each row holds the result of a single pattern.
- a `list` is returned for functions which create results of different lengths for each element, such as which.

Author(s)

Sven E. Templer

See Also

grepl, mclapply

Examples

```
#

# strings
s <- c("ab","ac","bc", NA)

# match all patterns (default)
mgrepl(c("a", "b"), s)

# match any of the patterns
mgrepl(c("a", "b"), s, any)
grepl("a|b", s)

# return logical matrix, one column for each pattern
mgrepl(c("a", "b"), s, identity)

# return count of matches
mgrepl(c("a", "b"), s, sum)

#
```

`ntri` *Return Triangular Numbers*

Description

Return the series of triangular (/triangle) numbers up to a number of `n` rows of a triangle. The series has the entry number "A000217" at `https://oeis.org/A000217` and starts like this: 0, 1, 3, 6, 10, ...

Usage

```
ntri(n)
```

Arguments

n	Positive integer value for sequence length.

Value

Returns an integer vector of length n.

Author(s)

Sven E. Templer (<sven.templer@gmail.com>)

---

nunique *Amount and Index of Unique Values*

Description

Return the index or amount of unique values in a vector.

Usage

```
nunique(x, na = TRUE, ...)

uniquei(x, na = TRUE, ...)
```

Arguments

x	Numeric vector to transform.
na	Logical, TRUE to include/count NA.
...	Arguments forwarded to unique.

Author(s)

Sven E. Templer

Examples

```
#

v <- c("a","b","a", NA)
nunique(v)
nunique(v, FALSE)
uniquei(v)
uniquei(v, FALSE)

#
```

p2star *P Value Significance Level Indicator*

Description

Transform p-values to character (e.g. stars) indicators by significance levels with the function symnum.

Usage

```
p2star(p, breaks = c(0, 0.001, 0.01, 0.05, 0.1, 1), symbols = c("***", "**",
  "*", ".", "n.s."))
```

Arguments

p	Vector with p values
breaks	The breaks from min (0) to max (1).
symbols	Symbols to use for values between breaks from min to max.

Author(s)

Sven E. Templer

Examples

```
#

p2star(c(1e-5,.1,.9))

#
```

plotn *Plot Nothing (but a Plot)*

Description

Create a plot, with empty elements by presetting default parameters.

Usage

```
plotn(x = 0:1, y = NULL, type = "n", xlab = "", ylab = "",
  xaxt = "n", yaxt = "n", frame.plot = F, ...)
```

Arguments

x	Coordinates of the points.
y	Coordinates of the y-axis.
type	Plot type.
xlab, ylab	Axis titles.
xaxt, yaxt	Axis types.
frame.plot	Plot the frame.
...	Forwarded arguments to plot.

Details

For details about the function see plot, which is called from plotn. More detailed information in plot.default and par.

Author(s)

Sven E. Templer

---

rmall *Remove All Objects from Global Environment*

Description

Remove all objects from the global environment.

Usage

```
rmall(...)
```

Arguments

...	Arguments forwarded to ls to get all objects.

Author(s)

Sven E. Templer

See Also

rm, ls

Examples

```
#

a <- b <- letters
ls()
rmall()
ls()

#
```

`scale0` *Scale Numeric Values to Defined Ranges*

Description

Scale numeric values to a range from 0 to 1 with the function `scale0` or to a chosen range with `scaler`.

Usage

```
scale0(x)

scaler(x, r = c(0, 1), b = range(x, na.rm = TRUE))
```

Arguments

`x`	Numeric vector to transform.
`r`	Numeric vector of length 2 for range to scale values of `x` to.
`b`	Numeric vector of length 2 to define the border of `x` to use as scaling minimum and maximum.

Author(s)

Sven E. Templer

Examples

```
#

scale0(0:10)
scale0(-1:3)
scale0(2:3)

scaler(0:10)
scaler(0:10, 1:2)
scaler(0:10, 1:2, c(0, 20))

#
```

`sort` *Sort data.frame Objects*

Description

Sort a data.frame by any column(s).

Usage

```
## S3 method for class 'data.frame'
sort(x, decreasing = FALSE, by = NULL, bye = NULL,
  na.last = NA, ...)
```

Arguments

`x`	A data.frame.
`decreasing`	Logical, sort in decreasing order. See also sort.
`by`	Index (integer) or names of columns in `x` to sort by in that order. If both `by` and `bye` are missing, all columns are used to sort in their order.
`bye`	Unquoted column name or `list()` or `.()` with unquoted column names to sort `x` by. Not evaluated if `by` is supplied.
`na.last`	`TRUE` to put missing values last, `FALSE` to put first or `NA` to remove.
`...`	Ignored for the data.frame method.

Author(s)

Sven E. Templer

Examples

```
#

d <- data.frame(a=c(1,1,1,2,NA),b=c(2,1,3,1,1),c=5:1)
d
sort(d) # sort by every column (a, then b, then c)
sort(d, TRUE, by="c") # decreasing by column 'c'
sort(d, bye=.(a,c)) # increasing by columns 'a' and then 'c'

#
```

squarematrix *Create a Square Matrix*

Description

Transform any m x n matrix to a square matrix by column/row names. Stops if no or duplicated dimnames are provided in x.

Usage

```
squarematrix(x)
```

Arguments

x	Object of class matrix.

Value

Returns a matrix.

Author(s)

Sven E. Templer

Examples

```
#

m <- matrix(1:6, 2, dimnames=list(2:3,1:3))
m
squarematrix(m)

#
```

strextr *Extract a Substring*

Description

This function extracts substring(s) which match a given pattern.

Usage

```
strextr(x, pattern, sep = " ", mult = F, unlist = F, cores = 1)
```

Arguments

x	Character vector.
pattern	Regular expression.
sep	Character string which separates the fields.
mult	Logical, if multiple matching fields should be returned, or otherwise NA.
unlist	Logical, unlists multiple results.
cores	Integer for number of computational cores to use.

Details

The function is deprecated and will be removed with `miscset` version 2. It is recommended to use str_extract or str_extract_all instead.

Value

A list of character vectors containing the substrings that are matching `pattern` and are separated by `sep` or `NA` if the pattern could not be found.

Author(s)

Sven E. Templer

Examples

```
#

library(stringr)

s <- c("A1 B1 C1","A2 B2", "AA A1", "AA", "B1 A1", "BB AB A1")

strextr(s, "^[AB][[:digit:]]$") # deprecated
str_extract(s, "[AB][:digit:]")

strextr(s, "^[AB][[:digit:]]$", mult = TRUE) # deprecated
str_extract_all(s, "[AB][:digit:]")

strextr(s, "^[AB][[:digit:]]$", mult = TRUE, unlist = TRUE) # deprecated
unlist(str_extract_all(s, "[AB][:digit:]")) # has no <NA> values

strextr(s, "^[C][[:digit:]]$") # deprecated
str_extract(s, "[C][:digit:]")

#
```

str_part *Split String and Return Part*

Description

Return the nth part of a splitted string.

Usage

```
str_part(x, split, n, ..., roll = F)

strpart(x, split, n, ..., roll = F)
```

Arguments

x	Character vector.
split	Regular expression splitting strings.
n	Number of part to extract.
...	Arguments passed to strsplit.
roll	Logical, if to use the last when less than maximum parts.

Value

A character vector of the extracted parts.

Author(s)

Sven E. Templer

See Also

strsplit

Examples

```
#

s <- c("abc","abcd","abc")

str_part(s, "", 4)
str_part(s, "", 4, roll=TRUE)

#
```

str_rev *Reverse Text Strings*

Description

Create a reverse version of strings.

Usage

```
str_rev(x)

strrev(x)
```

Arguments

x	vector with strings. Is coerced to character.

Value

Returns a character vector with reversed strings.

Author(s)

Sven E. Templer

See Also

rev

Examples

```
#

s <- c("abc","asdf")

str_rev(s)

#
```

`textable` *Table to Latex*

Description

This function enhances `xtable`: It can write the latex code of the table directly to a file and optionally adds a header/footer for the document structure. Also a system command can be given to convert the tex file to a pdf document, for example.

Usage

```
textable(d, file, caption = NULL, label = NULL, align = NULL,
  rownames = FALSE, topcapt = TRUE, digits = NULL, as.document = FALSE,
  landscape = FALSE, margin = 2, pt.size = 10, cmd = NULL, ...)
```

Arguments

`d`	Object (will be coerced to data.frame) to transform to a latex table.
`file`	Character string with output file name. If missing or `""`, the output is printed to the screen.
`caption`	Character vector with title of table.
`label`	Character vector with the latex label or HTML anchor.
`align`	Character vector with `'l'`, `'c'`, `'r'` for aligning the columns left, centered or right. Length is either one or 1 (for rownames column) + number of columns in `d` (even if `rownames = FALSE`)
`rownames`	Logical, include row names of `d`.
`topcapt`	Logical, put caption and label before 'tabular'.
`digits`	Number of digits to print from numeric values.
`as.document`	Logical. `TRUE` to add the document definition to the output. The document class is an article and the package a4paper is included.
`landscape`	Logical, use a landscape format for wider tables. Only with `as.document=TRUE`.
`margin`	Margin between table and page border in cm. Only with `as.document=TRUE`.
`pt.size`	Integer from 10 to 13 for the size of the characters. Only with `as.document=TRUE`.
`cmd`	A character vector with the system command to apply on the output file. Only if `file` is given and `as.document` is `TRUE`. `NULL` or an empty string system is not called.
`...`	Forwarded arguments to print.xtable.

Details

Example for a system call:
`cmd = "pdflatex -output-directory /path/to/files/"`

Value

Returns a character vector invisible. If `file` is set, then the content is written to a file. Else it is printed to the console.

Author(s)

Sven E. Templer

See Also

xtable

Examples

```
#

## Not run:
d <- head(trees)
dc <- 'R "trees" dataset.'
textable(d, rownames=TRUE, digits=4, caption=dc)
textable(d, '/tmp/trees.tex', caption=dc, as.document=TRUE,
  cmd='pdflatex --output-directory /tmp')

## End(Not run)

#
```

Index