Package 'latexpdf'

October 13, 2022

Type Package

Title Convert Tables to PDF or PNG

Version 0.1.7

Author Tim Bergsma

Maintainer Tim Bergsma <bergsmat@gmail.com>

Description Converts table-like objects to stand-alone PDF or PNG. Can be used to embed tables and arbitrary content in PDF or Word documents. Provides a low-level R interface for creating 'LaTeX' code, e.g. command() and a high-level interface for creating PDF documents, e.g. as.pdf.data.frame(). Extensive customization is available via mid-level functions, e.g. as.tabular(). See also 'package?latexpdf'. Support for PNG is experimental; see 'as.png.data.frame'. Adapted from 'metrumrg' <https://r-forge.r-project.org/R/?group_id=1215>. Requires a compatible installation of 'pdflatex', e.g. <https://miktex.org/>.

License GPL-3

Encoding UTF-8

RoxygenNote 7.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2021-08-17 18:40:02 UTC

R topics documented:

`latexpdf-package` *Create LaTeX Code and PDF Documents.*

Description

latexpdf helps you create pdf documents in R using LaTeX techniques; this is especially useful for making stand-alone PDF images of data.frames. For report-length PDF, some flavor of markup or markdown (e.g. Sweave, Rmarkdown) is probably a more attractive mechanism for the main document; in this context, **latexpdf** can be used to embed tables with the same aesthetics as stand-alone versions.

Details

"Anything in LaTeX can be expressed in terms of commands and environments", (<http://en.wikibooks.org/wiki/LaTeX/Abso
Accordingly, **latexpdf** provides R functions to generate LaTeX commands (command()) and environments (wrap()). For commands, care is taken to support options and arguments. These can be used to create character vectors containing arbitrary LaTeX code.

In fact, the package itself uses these functions to convert data frames to LaTeX code, providing reasonable defaults and supporting many aesthetic interventions. See '?as.tabular' and '?as.ltable' (tabular and table environments, respectively). See also 'vignette('tabular') for a demonstration of options.

While **latexpdf** can be useful for low-level operations, creating PDF documents directly is more powerful. Pre-generated LaTeX code can be inserted into literate programming documents (Sweave, Rmarkdown) or can be auto-converted to stand-alone PDF documents (see especially '?as.pdf.data.frame'). Functions tex2pdf() and viewtex() create and visualize arbitrary tex code by converting to PDF documents. They rely on as.pdf.document(), which places a system call to 'pdflatex'.

Some file formats, such as MS Word and Powerpoint, cannot embed PDF images. Experimental support is available for converting PDF to PNG. For example, if you switch from Rmarkdown's pdf_document() to word_document(), you may want to switch from as.pdf() to as.png()

for your images. PNG functionality relies requires the local Ghostscript installation (see '?ghostconvert') which is almost certainly present if 'pdflatex' is. Ghostscript may be hard to find. See '?tools::find_gs_cmd'; for Miktex installations, consider 'find_gs_cmd(gs_cmd = 'mgs')'.

Author(s)

Tim Bergsma, <bergsmat@gmail.com>

`as.document` *Coerce to LaTeX Document*

Description

Coerces to LaTeX document. Generic, with methods for character and data.frame.

Coerces to LaTex document from character.

Coerces to LaTeX document from data.frame.

Usage

```
as.document(x, ...)

## S3 method for class 'character'
as.document(
  x,
  preamble = makePreamble(...),
  thispagestyle = command("thispagestyle", args = "empty"),
  pagestyle = command("pagestyle", args = "empty"),
  prolog = NULL,
  epilog = NULL,
  ...
)

## S3 method for class 'data.frame'
as.document(
  x,
  rules = c(2, 1, 1),
  walls = 0,
  grid = FALSE,
  rowgroups = factor(rownames(x)),
  colgroups = factor(names(x)),
  rowbreaks = if (grid) breaks(rowgroups, ...) else 0,
  colbreaks = if (grid) breaks(colgroups, ...) else 0,
  rowcolors = NULL,
  charjust = "left",
  numjust = "right",
  justify = ifelse(sapply(x, is.numeric), numjust, charjust),
  colwidth = NA,
```

```
  paralign = "top",
  na = "",
  verbatim = ifelse(sapply(x, is.numeric), TRUE, FALSE),
  escape = "#",
  reserve = TRUE,
  trim = TRUE,
  wide = NULL,
  long = NULL,
  wider = 0,
  longer = 0,
  ...
)
```

Arguments

x	object to be converted, typically data.frame (paths of tex files for tex2pdf and viewtex)
...	passed to as.tabular.data.frame and as.document.character
preamble	latex markup to include before beginning the document
thispagestyle	thispagestyle command
pagestyle	pagestyle command
prolog	latex markup to include before x
epilog	latex markup to include after x
rules	numeric; will be recycled to length 3. indicates number of horizontal lines above and below the header, and below the last row.
walls	numeric, recycled to length 2. Number of vertical lines on left and right of table.
grid	logical, whether to have lines between rows and columns
rowgroups	a vector as long as nrow(x), non-repeats trigger horizontal lines
colgroups	a vector as long as names(x), non-repeats trigger vertical lines
rowbreaks	numeric: a manual way to specify numbers of lines between rows (ignores grid and rowgroups)
colbreaks	numeric: a manual way to specify numbers of lines between columns (ignores grid and colgroups)
rowcolors	character vector of color names, recycled as necessary to color all rows (NULL: no color)
charjust	default justification for character columns
numjust	default justification for numeric columns
justify	manual specification of column justifications: left, right, center, or decimal (vector as long as ncol(x))
colwidth	manual specification of column width. (vector of length ncol(x).) Overrides justify where not NA.
paralign	used with colwidth to align paragraphs: top, middle, or bottom.
na	string to replace NA elements

verbatim	whether to use verbatim environment for numeric fields. Makes sense for decimal justification; interacts with `trim` and `justify`.
escape	symbol used by 'verb' command as delimiter. A warning is issued if it is found in non-NA text.
reserve	substitute escape sequences for LaTeX reserved characters
trim	passed to the format command: true by default, so that alignment is the responsibility of just the tabular environment arguments
wide	nominal page width in mm
long	nominal page length in mm
wider	additional page width in mm
longer	additional page lenth in mm

Value

character

character

character

Methods (by class)

- `character`: character method
- `data.frame`: data.frame method

See Also

`as.tabular.data.frame`

`as.document.character`

`as.pdf.data.frame`

Examples

```
as.document(head(Theoph))
```

`as.ltable` *Coerce to LaTeX Table Environment*

Description

Coerces to LaTeX table environment. Generic, with methods for data.frame, table, and matrix.

Coerces to LaTeX table from data.frame. See details.

Coerces to ltable from table, reclassifying its argument as matrix.

Coerces to ltable from matrix; tires to capture the column names as a caption, and (like `as.tabular.matrix`) converts its argument to data.frame, capturing rownames as a column in the first position if rownames are suitably named.

Usage

```
as.ltable(x, ...)

## S3 method for class 'data.frame'
as.ltable(
  x,
  caption = NULL,
  cap = caption,
  cap.top = TRUE,
  label = NULL,
  options = "H",
  environments = "center",
  source = NULL,
  file = NULL,
  source.label = "source: ",
  file.label = "file: ",
  basefile = FALSE,
  footnote.size = "tiny",
  ...
)

## S3 method for class 'table'
as.ltable(x, ...)

## S3 method for class 'matrix'
as.ltable(x, caption = names(dimnames(x))[[2]], ...)
```

Arguments

x	object
...	passed arguments
caption	full version of the caption
cap	short version of the caption, for list of tables
cap.top	Should caption be placed at the top, instead of bottom?
label	optional label
options	options for latex table environment, e.g. H or !htpb
environments	extra environments to nest between 'table' and 'tabular'
source	optional source attribution
file	optional file name
source.label	optional text to preceed source if specified
file.label	optional text to preceed file if specified
basefile	if TRUE, strip path from file for display purposes
footnote.size	font size for source and file, etc.

Details

Converts data.frame to tabular, then wraps it in specified environments, then wraps result in a latex table environment. Result is returned visibly, or if `file` is specified it is printed to file and returned invisibly. If `source` and `source.label` are defined, they will be printed in a tiny font immediately under the table (bound to the tabular element). If `file` and `file.label` are defined as well, they will be printed (tiny) under source. Set `source.label` to NULL to suppress embedding of `source`; set to empty string to suppress source label. Set `file.label` to NULL to suppress embedding of `file`; set to empty string to suppress file label. Note that `file` controls file destination, whether or not represented in the result.

Extra arguments(...) are passed to `as.tabular`.

Value

character

Methods (by class)

- `data.frame`: data.frame method
- `table`: table method
- `matrix`: matrix method

See Also

`as.tabular`

`as.tabular`

Examples

```
as.ltable(head(Theoph))
as.ltable(table(1:3,4:6))
```

---

as.pdf *Coerce to PDF*

Description

Coerces to PDF. Generic, with methods for character, document, and data.frame.

Coerces to PDF from document. Makes a system call to 'pdflatex'. Extra arguments ignored.

Coerces character to document and then to PDF. Extra arguments passed to `as.document.character` and `as.pdf.document`.

Usage

```
as.pdf(x, ...)

## S3 method for class 'document'
as.pdf(
  x,
  stem = "latexpdf-doc",
  dir = ".",
  clean = TRUE,
  ignore.stdout = FALSE,
  ignore.stderr = FALSE,
  show.output.on.console = TRUE,
  ...
)

## S3 method for class 'character'
as.pdf(x, stem, ...)
```

Arguments

x	object
...	passed arguments
stem	the stem of a file name (no extension)
dir	output directory
clean	whether to delete system files after pdf creation
ignore.stdout	passed to system
ignore.stderr	passed to system
show.output.on.console	passed to system

Value

the output file path (invisible)

Methods (by class)

- document: document method
- character: character method

See Also

as.pdf.character

as.pdf.document

as.pdf.data.frame

as.png

tex2pdf

viewtex

`as.pdf.data.frame` *Coerce to PDF from Data Frame*

Description

Coerces data.frame to document and then to PDF. Extra arguments are passed to `makePreamble`. `as.document.data.frame` will try to guess an appropriate width and length(`wide`, `long`) for the page, but you may need something `wider` or `longer`, especially if you adjust aesthetics. Negative values for `wider` and `longer` are meaningful.

Usage

```
## S3 method for class 'data.frame'
as.pdf(
  x,
  rules = c(2, 1, 1),
  walls = 0,
  grid = FALSE,
  rowgroups = factor(rownames(x)),
  colgroups = factor(names(x)),
  rowbreaks = if (grid) breaks(rowgroups) else 0,
  colbreaks = if (grid) breaks(colgroups) else 0,
  rowgrouprule = 0,
  colgrouprule = 0,
  rowcolors = NULL,
  rowgrouplabel = " ",
  charjust = "left",
  numjust = "right",
  justify = ifelse(sapply(x, is.numeric), numjust, charjust),
  colwidth = NA,
  paralign = "top",
  na = "",
  verbatim = ifelse(sapply(x, is.numeric), TRUE, FALSE),
  escape = "#",
  reserve = TRUE,
  trim = TRUE,
  source = NULL,
  file = NULL,
  source.label = "source: ",
  file.label = "file: ",
  basefile = FALSE,
  tabularEnvironment = "tabular",
  footnote.size = "tiny",
  geoLeft = "1mm",
```

```
  geoRight = "1mm",
  geoTop = "1mm",
  geoBottom = "1mm",
  wide = NULL,
  long = NULL,
  wider = 0,
  longer = 0,
  thispagestyle = command("thispagestyle", args = "empty"),
  pagestyle = command("pagestyle", args = "empty"),
  prolog = NULL,
  epilog = NULL,
  stem = "latexpdf-doc",
  dir = ".",
  clean = TRUE,
  ...
)
```

Arguments

x	data.frame
rules	numeric; will be recycled to length 3. indicates number of horizontal lines above and below the header, and below the last row.
walls	numeric, recycled to length 2. Number of vertical lines on left and right of table.
grid	logical, whether to have lines between rows and columns
rowgroups	a vector as long as nrow(x), non-repeats trigger horizontal lines
colgroups	a vector as long as names(x), non-repeats trigger vertical lines
rowbreaks	numeric: a manual way to specify numbers of lines between rows (ignores grid and rowgroups)
colbreaks	numeric: a manual way to specify numbers of lines between columns (ignores grid and colgroups)
rowgrouprule	number of lines to set off row group column, if rowgroups supplied as character
colgrouprule	number of lines to set off col group header, if colgroups supplied as character
rowcolors	character vector of color names, recycled as necessary to color all rows (NULL: no color); not compatible with rowgroups
rowgrouplabel	character string (at least one character) to label rowgroup column
charjust	default justification for character columns
numjust	default justification for numeric columns
justify	manual specification of column justifications: left, right, center, or decimal (vector as long as ncol(x))
colwidth	manual specification of column width. (vector of length ncol(x).) Overrides `justify` where not `NA`.
paralign	used with colwidth to align paragraphs: top, middle, or bottom.
na	string to replace NA elements

verbatim	whether to use verbatim environment for numeric fields. Makes sense for decimal justification; interacts with `trim` and `justify`.
escape	symbol used by 'verb' command as delimiter. A warning is issued if it is found in non-NA text.
reserve	substitute escape sequences for LaTeX reserved characters
trim	passed to the format command: true by default, so that alignment is the responsibility of just the tabular environment arguments
source	optional source attribution
file	optional file name
source.label	optional text to preceed source if specified
file.label	optional text to preceed file if specified
basefile	if TRUE, strip path from file for display purposes
tabularEnvironment	default `tabular`; consider also `longtable`
footnote.size	font size for source and file attributions
geoLeft	geometry package: left margin
geoRight	geometry package: right margin
geoTop	geometry package: top margin
geoBottom	geometry package: bottom margin
wide	nominal page width in mm
long	nominal page length in mm
wider	additional page width in mm
longer	additional page lenth in mm
thispagestyle	thispagestyle command
pagestyle	pagestyle command
prolog	latex markup to include before x
epilog	latex markup to include after x
stem	the stem of a file name (no extension)
dir	output directory
clean	whether to delete system files after pdf creation
...	passed eventually to `makePreamble` and `reserve`

See Also

`as.pdf.character`

`as.pdf.document`

`as.document.data.frame`

`as.tabular.data.frame`

`as.png.data.frame`

Examples

```
## Not run: as.pdf(head(Theoph))
## Not run: as.pdf(Theoph[0,])
```

`as.png` *Coerce to PNG*

Description

Coerces to PNG. Generic, with methods for character, document, and data.frame.

Coerces to PNG from document. Makes a system call to 'pdflatex', converts resulting file to PNG. Extra arguments passed to `as.pdf.document` and `ghostconvert`.

Coerces character to document, PDF, and then PNG. Extra arguments passed to `as.document.character` and `as.png.document`.

Usage

```
as.png(x, ...)

## S3 method for class 'document'
as.png(
  x,
  stem = "latexpdf-doc",
  dir = ".",
  clean = TRUE,
  replace = TRUE,
  multipage = TRUE,
  ...
)

## S3 method for class 'character'
as.png(x, stem, ...)
```

Arguments

`x`	object
`...`	passed arguments
`stem`	the stem of a file name (no extension)
`dir`	output directory
`clean`	whether to delete system files after PNG creation
`replace`	whether to delete the pdf file
`multipage`	whether to convert multiple pages of the PDF

Methods (by class)

- `document`: document method
- `character`: character method

See Also

`as.png.character`

`as.png.document`

`as.png.data.frame`

`as.pdf`

as.pdf.document

ghostconvert

`tex2png`

`viewtex`

`ghostconvert`

`as.png.data.frame` *Coerce to PNG from Data Frame.*

Description

Coerces to PNG from 'data.frame'.

Usage

```
## S3 method for class 'data.frame'
as.png(x, replace = TRUE, ...)
```

Arguments

`x`	data.frame
`replace`	whether to delete the intermediate PDF if successful
`...`	passed to `as.pdf.data.frame` and to `ghostconvert`

Value

path to the file created

See Also

`as.pdf.data.frame`

`ghostconvert`

Examples

```
## Not run:
browseURL(as.png(head(Theoph), dir = tempdir(), gs_cmd = 'mgs'))

## End(Not run)
```

`as.tabular` *Coerce to tabular*

Description

Coerces to tabular. Generic, with methods for data.frame, table, and matrix.

Coerces to tabular from data.frame. Extra arguments passed to `format.data.frame`.

Coerces to tabular from table.

Coerces to tabular from matrix.

Usage

```
as.tabular(x, ...)

## S3 method for class 'data.frame'
as.tabular(
  x,
  rules = c(2, 1, 1),
  walls = 0,
  grid = FALSE,
  rowgroups = factor(rownames(x)),
  colgroups = factor(names(x)),
  rowbreaks = if (grid) breaks(rowgroups) else 0,
  colbreaks = if (grid) breaks(colgroups) else 0,
  rowgrouprule = 0,
  colgrouprule = 0,
  rowcolors = NULL,
  rowgrouplabel = " ",
  charjust = "left",
  numjust = "right",
  justify = ifelse(sapply(x, is.numeric), numjust, charjust),
  decimal.mark = getOption("OutDec"),
  colwidth = NA,
  paralign = "top",
  na = "",
  verbatim = ifelse(sapply(x, is.numeric), TRUE, FALSE),
  escape = "#",
  reserve = TRUE,
  trim = TRUE,
  source = NULL,
```

```
  file = NULL,
  source.label = "source: ",
  file.label = "file: ",
  basefile = FALSE,
  tabularEnvironment = "tabular",
  footnote.size = "tiny",
  ...
)

## S3 method for class 'table'
as.tabular(x, ...)

## S3 method for class 'matrix'
as.tabular(x, ...)
```

Arguments

x	object
...	passed arguments
rules	numeric; will be recycled to length 3. indicates number of horizontal lines above and below the header, and below the last row.
walls	numeric, recycled to length 2. Number of vertical lines on left and right of table.
grid	logical, whether to have lines between rows and columns
rowgroups	a vector as long as nrow(x), non-repeats trigger horizontal lines
colgroups	a vector as long as names(x), non-repeats trigger vertical lines
rowbreaks	numeric: a manual way to specify numbers of lines between rows (ignores grid and rowgroups)
colbreaks	numeric: a manual way to specify numbers of lines between columns (ignores grid and colgroups)
rowgrouprule	number of lines to set off row group column, if rowgroups supplied as character
colgrouprule	number of lines to set off col group header, if colgroups supplied as character
rowcolors	character vector of color names, recycled as necessary to color all rows (NULL: no color)
rowgrouplabel	character string (at least one character) to label rowgroup column
charjust	default justification for character columns
numjust	default justification for numeric columns
justify	manual specification of column justifications: left, right, center, or decimal (vector as long as ncol(x))
decimal.mark	passed to format.data.frame
colwidth	manual specification of column width. (vector of length ncol(x).) Overrides justify where not NA.
paralign	used with colwidth to align paragraphs: top, middle, or bottom.
na	string to replace NA elements

verbatim	whether to use verbatim environment for numeric fields. Makes sense for decimal justification; interacts with trim and justify.
escape	symbol used by 'verb' command as delimiter. A warning is issued if it is found in non-NA text.
reserve	substitute escape sequences for LaTeX reserved characters
trim	passed to the format command: true by default, so that alignment is the responsibility of just the tabular environment arguments
source	optional source attribution
file	optional file name
source.label	optional text to preceed source if specified
file.label	optional text to preceed file if specified
basefile	if TRUE, strip path from file for display purposes
tabularEnvironment	default tabular; consider also longtable
footnote.size	font size for source and file attributions

Value

tabular

Methods (by class)

- data.frame: data.frame method
- table: table method
- matrix: matrix method

See Also

as.ltable

Examples

```
as.tabular(head(Theoph))
as.tabular(table(1:3,4:6))
as.tabular(head(Theoph,source='foo/bar',footnote.size='huge'))
## Not run: as.pdf(head(Theoph))
```

breaks *Calculate Row and Column Breaks*

Description

Calculates row and column breaks by finding first elements among repeats.

Usage

```
breaks(x, ...)
```

Arguments

x	vector
...	ignored

Value

integer

See Also

as.tabular

command *Format a Latex Command*

Description

Format a latex command. x is formated as a latex command, with the options (possibly named) inserted in square brackets, and the arguments each enclosed in curly braces. depth spaces are added to the left end of the string.

Usage

```
command(x, options = NULL, args = NULL, depth = 0)
```

Arguments

x	length one character
options	vector or list
args	vector or list
depth	integer

Value

character

See Also

wrap

as.ltable.data.frame

Examples

```
command('caption',options='short',args='long')
```

---

ghostconvert *Call Ghostscript.*

Description

Call Ghostscript, converting by default from PDF to PNG.

Usage

```
ghostconvert(
  x,
  y = file.path(gdir, out),
  gdir = dirname(x),
  out = sub("\\.[^.]+$", paste0(if (multipage) multifix else NULL, ".", suffix),
    basename(x)),
  gs_cmd = "",
  device = "pngalpha",
  multipage = FALSE,
  multifix = "-%03d",
  suffix = "png",
  antialias = 4,
  resolution = 300,
  replace = TRUE,
  other = "",
  ...
)
```

Arguments

x	path for file to be converted
y	path for output file
gdir	directory for png output
out	filename for output file

gs_cmd	passed to find_gs_cmd; perhaps 'gs' or 'gswin32c' or 'mgs' (from Miktex)
device	output device type
multipage	whether to convert multiple pages
multifix	a filename suffix when converting multiple pages
suffix	file extension for output
antialias	font antialiasing
resolution	raster image resolution
replace	whether to delete x if successful
other	other arguments to ghostscript
...	ignored

Value

the name of the file created

Examples

```
## Not run:
pdf <- as.pdf(head(Theoph),dir = tempdir())
png <- ghostconvert(pdf, gs_cmd = 'mgs')
browseURL(png)

## End(Not run)
```

---

makePreamble *Make a Preamble for a LaTeX Document*

Description

Makes a preamble for a LaTeX Document.

Usage

```
makePreamble(
  landscape = FALSE,
  wide = if (landscape) 279.4 else 215.9,
  long = if (landscape) 215.9 else 279.4,
  geoLeft = "1mm",
  geoRight = "1mm",
  geoTop = "1mm",
  geoBottom = "1mm",
  documentclass = command("documentclass", args = "article"),
  xcolorPackage = command("usepackage", options = list("usenames", "dvispnames",
    "svgnames", "table"), args = "xcolor"),
 geometryPackage = command("usepackage", options = list(left = geoLeft, top = geoTop,
```

```
    bottom = geoBottom, right = geoRight), args = "geometry"),
  geometry = command("geometry", args = list(paste0("papersize=", paste0("{", wide,
    "mm", ",", long, "mm}")))),
  multirow = command("usepackage", args = "multirow"),
  float = command("usepackage", args = "float"),
  longtable = command("usepackage", args = "longtable"),
  inputenc = command("usepackage", options = "utf8", args = "inputenc"),
  fontenc = command("usepackage", options = "T1", args = "fontenc"),
  morePreamble = NULL,
  ...
)
```

Arguments

landscape	if TRUE, default orientation is 'landscape' not 'portrait'
wide	page width in mm
long	page lenth in mm
geoLeft	geometry package: left margin
geoRight	geometry package: right margin
geoTop	geometry package: top margin
geoBottom	geometry package: bottom margin
documentclass	document class command
xcolorPackage	xcolor package command
geometryPackage	geometry package command
geometry	geometry specification
multirow	multirow specification
float	float specification
longtable	longtable specification
inputenc	input encoding
fontenc	output encoding
morePreamble	additional preamble before beginning the document
...	ignored

Value

character

Examples

```
makePreamble()
```

reserve *Escape LaTeX Reserved Characters*

Description

Escapes LaTeX reserved characters. Generic, with default, character, and data.frame methods.

Usage

```
reserve(x, ...)
```

Arguments

x	object
...	passed arguments

See Also

Other reserve: `reserve.character()`, `reserve.data.frame()`, `reserve.default()`, `reserve.factor()`, `reserve.list()`

reserve.character *Escape LaTeX Reserved Characters for Character Objects*

Description

Escapes LaTeX reserved characters as suggested by https://en.wikibooks.org/wiki/LaTeX/Basics#Reserved_Characters.

Usage

```
## S3 method for class 'character'
reserve(
  x,
  target = c("#", "$", "%", "^", "&", "_", "{", "}", "~", "\\"),
  replacement = c("\\#", "\\$", "\\%", "\\^{}", "\\&", "\\_", "\\{",
    "\\}", "\\~{}", "\\textbackslash{}"),
  ...
)
```

Arguments

x	character
target	sequence to find (regular expression unless fixed is false)
replacement	sequence to use
...	passed arguments

See Also

Other reserve: reserve.data.frame(), reserve.default(), reserve.factor(), reserve.list(), reserve()

---

reserve.data.frame *Escape LaTeX Reserved Characters in Data Frames*

Description

Applies reserve to the columns of a data.frame.

Usage

```
## S3 method for class 'data.frame'
reserve(x, ...)
```

Arguments

x	object
...	passed arguments

See Also

Other reserve: reserve.character(), reserve.default(), reserve.factor(), reserve.list(), reserve()

Examples

```
foo <- c('#','$%^','&_{','}~\\')
bar <- data.frame(
 stringsAsFactors = FALSE,
 a = as.numeric(factor(foo)),
 b = foo,
 c = factor(foo)
)

reserve(bar)
# as.pdf(bar, wider = 10)
# as.pdf(bar, target = '#') # fails
```

reserve.default *Escape LaTeX Reserved Characters Using Default Method*

Description

The default method returns its argument.

Usage

```
## Default S3 method:
reserve(x, ...)
```

Arguments

x	object
...	passed arguments

See Also

Other reserve: reserve.character(), reserve.data.frame(), reserve.factor(), reserve.list(), reserve()

reserve.factor *Escape LaTeX Reserved Characters for Factor Objects*

Description

Coerces to character and escapes reserved characters.

Usage

```
## S3 method for class 'factor'
reserve(x, ...)
```

Arguments

x	factor
...	passed arguments

See Also

Other reserve: reserve.character(), reserve.data.frame(), reserve.default(), reserve.list(), reserve()

reserve.list *Escape LaTeX Reserved Characters in Lists*

Description

Applies `reserve` to the elements of a list.

Usage

```
## S3 method for class 'list'
reserve(x, ...)
```

Arguments

`x`	object
`...`	passed arguments

See Also

Other reserve: `reserve.character()`, `reserve.data.frame()`, `reserve.default()`, `reserve.factor()`, `reserve()`

tex2pdf *Convert TEX to PDF*

Description

Converts TEX to PDF. `tex2pdf` accepts the file names of TEX fragments. It reads those fragments, wraps them like documents and makes PDF files.

Usage

```
tex2pdf(x, stem = NULL, dir = NULL, clean = TRUE, onefile = FALSE, ...)
```

Arguments

`x`	vector of file names
`stem`	the stem of a file name (no extension)
`dir`	output directory
`clean`	whether to delete system files after PDF creation
`onefile`	whether to combine TEX snippets into a single file
`...`	passed to `as.pdf.character`

Value

invisible vector of paths to created files

See Also

as.pdf.character

viewtex

Examples

```
file <- file.path(tempdir(),'test.tex')
writeLines(as.ltable(head(Theoph)), file)
tex2pdf(file)
```

---

tex2png *Convert TEX to PNG*

Description

Converts TEX to PNG. tex2png accepts the file names of TEX fragments. It reads those fragments, wraps them like documents and makes PNG files (converted from PDF files).

Usage

```
tex2png(
  x,
  stem = NULL,
  dir = NULL,
  clean = TRUE,
  onefile = FALSE,
  replace = TRUE,
  ...
)
```

Arguments

x	vector of file names
stem	the stem of a file name (no extension)
dir	output directory
clean	whether to delete system files after PNG creation
onefile	whether to combine tex snippets into a single file
replace	whether to delete the intermediate PDF files
...	passed to tex2pdf and ghostconvert

See Also

as.png.character

viewtex

Examples

```
## Not run:
file <- file.path(tempdir(),'test.tex')
writeLines(as.ltable(head(Theoph)), file)
tex2png(file, gs_cmd = 'mgs')

## End(Not run)
```

---

viewtex *Render and View TEX Files*

Description

Renders and TEX files as PDF and opens them for viewing.

Usage

```
viewtex(x, delete = TRUE, latency = 1, png = FALSE, ...)
```

Arguments

x	vector of file names
delete	whether temporary pdf (_doc.pdf) should persist
latency	how many seconds to wait before deleting temporary PDF,
png	view as png instead of pdf
...	passed to tex2pdf

See Also

tex2pdf

tex2png

as.pdf.character

Examples

```
file <- file.path(tempdir(),'test.tex')
writeLines(as.ltable(head(Theoph)), file)
## Not run:
viewtex(file)
viewtex(file, png = TRUE, gs_cmd = 'mgs')

## End(Not run)
```

wrap *Wrap Text in a Latex Environment*

Description

Wrap text in a latex environment. `x` is wrapped in the specified environment, with options and arguments formatted appropriately.

Usage

```
wrap(x, environment, options = NULL, args = NULL, depth = 0)
```

Arguments

`x`	character
`environment`	name of environment
`options`	list or vector
`args`	list or vector
`depth`	integer (extra spaces on the left)

Value

character

See Also

`command`

`as.ltable.data.frame`

Examples

```
wrap('Hello','center')
```

Index