

# Package ‘latexpdf’

July 22, 2025

**Type** Package

**Title** Convert Tables to PDF or PNG

**Version** 0.1.8

**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](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** 7.2.3

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2023-12-19 18:30:02 UTC

## Contents

|                             |    |
|-----------------------------|----|
| latexpdf-package . . . . .  | 2  |
| as.document . . . . .       | 3  |
| as.ltable . . . . .         | 5  |
| as.pdf . . . . .            | 7  |
| as.pdf.data.frame . . . . . | 9  |
| as.png . . . . .            | 12 |
| as.png.data.frame . . . . . | 13 |

|                              |    |
|------------------------------|----|
| as.tabular . . . . .         | 14 |
| breaks . . . . .             | 17 |
| command . . . . .            | 17 |
| ghostconvert . . . . .       | 18 |
| makePreamble . . . . .       | 19 |
| reserve . . . . .            | 21 |
| reserve.character . . . . .  | 21 |
| reserve.data.frame . . . . . | 22 |
| reserve.default . . . . .    | 23 |
| reserve.factor . . . . .     | 23 |
| reserve.list . . . . .       | 24 |
| tex2pdf . . . . .            | 24 |
| tex2png . . . . .            | 25 |
| viewtex . . . . .            | 26 |
| wrap . . . . .               | 27 |

|              |           |
|--------------|-----------|
| <b>Index</b> | <b>28</b> |
|--------------|-----------|

---

|                  |                                             |
|------------------|---------------------------------------------|
| latexpdf-package | <i>Create LaTeX Code and PDF Documents.</i> |
|------------------|---------------------------------------------|

---

## 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 '?ghost-convert') 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 <a href="#">as.tabular.data.frame</a> and <a href="#">as.document.character</a>                                       |
| 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 <b>reserved</b> 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 length in mm                                                                                                     |

**Value**

character  
character  
character

**Methods (by class)**

- `as.document(character)`: character method
- `as.document(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; tries 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**

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

- `as.ltable(data.frame)`: data.frame method
- `as.ltable(table)`: table method
- `as.ltable(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 <a href="#">system</a>                  |
| ignore.stderr          | passed to <a href="#">system</a>                  |
| show.output.on.console | passed to <a href="#">system</a>                  |

**Value**

the output file path (invisible)

**Methods (by class)**

- `as.pdf(document)`: document method
- `as.pdf(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)                                  |
| rowgroupprule | number of lines to set off row group column, if rowgroups supplied as character                                                 |
| colgroupprule | 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 <b>reserved</b> 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 precede source if specified                                                                                     |
| file.label         | optional text to precede 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 length 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 <a href="#">makePreamble</a> and <a href="#">reserve</a>                                                    |

**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)**

- `as.png(document)`: document method
- `as.png(character)`: character method

**See Also**

[as.png.character](#)  
[as.png.document](#)  
[as.png.data.frame](#)  
[as.pdf](#)  
`as.pdf.document`  
`ghostconvert`  
[tex2png](#)  
[viewtex](#)  
[ghostconvert](#)

---

|                                |                                       |
|--------------------------------|---------------------------------------|
| <code>as.png.data.frame</code> | <i>Coerce to PNG from Data Frame.</i> |
|--------------------------------|---------------------------------------|

---

**Description**

Coerces to PNG from 'data.frame'.

**Usage**

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

**Arguments**

|                      |                                                                                 |
|----------------------|---------------------------------------------------------------------------------|
| <code>x</code>       | data.frame                                                                      |
| <code>replace</code> | whether to delete the intermediate PDF if successful                            |
| <code>...</code>     | passed to <a href="#">as.pdf.data.frame</a> and to <a href="#">ghostconvert</a> |

**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)                                  |
| rowgroupprule | number of lines to set off row group column, if rowgroups supplied as character                                                 |
| colgroupprule | 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 <a href="#">format.data.frame</a>                                                                                     |
| 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 <b>reserved</b> 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 precede source if specified                                                                                     |
| file.label         | optional text to precede 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)**

- as.tabular(data.frame): data.frame method
- as.tabular(table): table method
- as.tabular(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 | <i>Calculate Row and Column Breaks</i> |
|--------|----------------------------------------|

---

**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 | <i>Format a Latex Command</i> |
|---------|-------------------------------|

---

**Description**

Format a latex command. x is formatted 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 | <i>Call Ghostscript.</i> |
|--------------|--------------------------|

---

**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 <code>find_gs_cmd</code> ; 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("dvipsnames", "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 length 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 | <i>Escape LaTeX Reserved Characters</i> |
|---------|-----------------------------------------|

---

**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 | <i>Escape LaTeX Reserved Characters for Character Objects</i> |
|-------------------|---------------------------------------------------------------|

---

**Description**

Escapes LaTeX reserved characters as suggested by [https://en.wikibooks.org/wiki/LaTeX/Basics#Reserved\\_Characters](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 | <i>Escape LaTeX Reserved Characters in Data Frames</i> |
|--------------------|--------------------------------------------------------|

---

**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 | <i>Escape LaTeX Reserved Characters Using Default Method</i> |
|-----------------|--------------------------------------------------------------|

---

**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 | <i>Escape LaTeX Reserved Characters for Factor Objects</i> |
|----------------|------------------------------------------------------------|

---

**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 <a href="#">as.pdf.character</a>         |

**Value**

invisible vector of paths to created files

**See Also**

[as.pdf.character](#)

[viewtex](#)

**Examples**

```
## Not run:
file <- file.path(tempdir(), 'test.tex')
writeLines(as.ltable(head(Theoph)), file)
tex2pdf(file)

## End(Not run)
```

---

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 <a href="#">tex2pdf</a> and <a href="#">ghostconvert</a> |

**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 ( <code>_doc.pdf</code> ) should persist |
| latency | how many seconds to wait before deleting temporary PDF,        |
| png     | view as png instead of pdf                                     |
| ...     | passed to <a href="#">tex2pdf</a>                              |

**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**

|                          |                                    |
|--------------------------|------------------------------------|
| <code>x</code>           | character                          |
| <code>environment</code> | name of environment                |
| <code>options</code>     | list or vector                     |
| <code>args</code>        | list or vector                     |
| <code>depth</code>       | integer (extra spaces on the left) |

**Value**

character

**See Also**

[command](#)

[as.ltable.data.frame](#)

**Examples**

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

# Index

- \* **reserve**
  - reserve, [21](#)
  - reserve.character, [21](#)
  - reserve.data.frame, [22](#)
  - reserve.default, [23](#)
  - reserve.factor, [23](#)
  - reserve.list, [24](#)
- as.document, [3](#)
- as.document.character, [4](#), [5](#), [7](#), [12](#)
- as.document.data.frame, [9](#), [11](#)
- as.ltable, [5](#), [16](#)
- as.ltable.data.frame, [18](#), [27](#)
- as.pdf, [7](#), [13](#)
- as.pdf.character, [8](#), [11](#), [24–26](#)
- as.pdf.data.frame, [5](#), [8](#), [9](#), [13](#)
- as.pdf.document, [7](#), [8](#), [11](#), [12](#)
- as.png, [8](#), [12](#)
- as.png.character, [13](#), [26](#)
- as.png.data.frame, [11](#), [13](#), [13](#)
- as.png.document, [12](#), [13](#)
- as.tabular, [7](#), [14](#), [17](#)
- as.tabular.data.frame, [4](#), [5](#), [11](#)
- breaks, [17](#)
- command, [17](#), [27](#)
- find\_gs\_cmd, [19](#)
- format.data.frame, [14](#), [15](#)
- ghostconvert, [12](#), [13](#), [18](#), [25](#)
- latexpdf-package, [2](#)
- makePreamble, [9](#), [11](#), [19](#)
- reserve, [11](#), [21](#), [22–24](#)
- reserve.character, [21](#), [21](#), [22–24](#)
- reserve.data.frame, [21](#), [22](#), [22](#), [23](#), [24](#)
- reserve.default, [21–23](#), [23](#), [24](#)
- reserve.factor, [21–23](#), [23](#), [24](#)
- reserve.list, [21–23](#), [24](#)
- system, [8](#)
- tex2pdf, [9](#), [24](#), [25](#), [26](#)
- tex2png, [13](#), [25](#), [26](#)
- viewtex, [9](#), [13](#), [25](#), [26](#), [26](#)
- wrap, [18](#), [27](#)