

Package ‘docxtools’

June 3, 2020

Title Tools for R Markdown to Docx Documents

Version 0.2.2

Language en-US

Description A set of helper functions for using R Markdown to create documents in docx format, especially documents for use in a classroom or workshop setting.

Depends R (>= 3.1.2)

License MIT + file LICENSE

Encoding UTF-8

LazyData TRUE

RoxygenNote 7.1.0

Imports dplyr, ggplot2, lubridate, purrr, rlang, stringr, tidyr

Suggests covr, knitr, rmarkdown, rprojroot, testthat

VignetteBuilder knitr

URL <https://github.com/graphdr/docxtools>

BugReports <https://github.com/graphdr/docxtools/issues>

NeedsCompilation no

Author Richard Layton [aut, cre]

Maintainer Richard Layton <graphdoctor@gmail.com>

Repository CRAN

Date/Publication 2020-06-03 18:40:03 UTC

R topics documented:

density	2
docxtools	2
format_engr	3
put_axes	4
put_gap	5
Index	7

density *Ideal gas sample data frame*

Description

Air density at room temperature. A data frame to illustrate features of the `format_engr()` and `align_pander()` functions.

Usage

```
density
```

Format

A data frame with columns:

date A date variable.

trial A character variable.

T_K Numeric air temperature in Kelvin.

p_Pa Numeric air pressure in Pascals.

R Numeric gas constant in J/(kg K)

density Numeric air density in kg/m³

docxtools *docxtools: R Markdown to docx helper functions.*

Description

The docxtools package provides a set of helper functions for using R Markdown to create documents in docx format, especially documents for use in a classroom or workshop setting.

Details

The package provides two categories of functions: one for formatting numbers and tables; and one for placing specific objects in a docx output document.

Number and table formatting

`format_engr()` Format numerical variables in engineering notation.

Insert objects into docx documents

`put_axes()` Insert a two-dimensional coordinate axes in a document. `put_gap()` Insert a gap or whitespace in a document.

format_engr	<i>Format numerical variables in engineering notation.</i>
-------------	------------------------------------------------------------

Description

Converts numeric variables in a data frame into character variables formatted in engineering notation. The formatted character variables are delimited with \dots for rendering as inline math.

Usage

```
format_engr(x, sigdig = NULL, ambig_0_adj = FALSE)
```

Arguments

x	A data frame with numeric variables to be formatted.
sigdig	An optional vector of significant digits.
ambig_0_adj	An optional logical to adjust the format to remove ambiguous trailing zeros. If TRUE, the decimal point is moved three places to the left. Default is FALSE.

Details

Engineering notation is a subset of scientific notation: the exponent of ten must be divisible by three. Powers of ten are written with a product symbol (the LaTeX \times symbol), not computer E-notation. The numerical output strings are bounded by \dots for rendering in math mode.

The output format is "coefficient x 10^{exponent}" except for exponents equal to 0, 1, or 2. In these cases, the output format is floating-point.

The input argument is a data frame with at least one numeric variable. Only double variables (not integers and not dates) are formatted in powers of ten.

The function distinguishes between integers that represent factors and those that represent numerical integers. Numerical integers are returned as characters in math mode. Factors and other non-numeric variables are returned unaffected.

Each double variable (except dates) can be assigned its own number of significant digits. The default is 4. If sigdig is a single value, it is used for all variables; if a vector, it must have the same number of elements as there are double variables in the data frame. Setting sigdig = 0 returns the double variables as characters but unaltered.

For numbers with no decimal point, trailing zeros can be ambiguous. Such numbers can be optionally reported with the decimal moved an additional three places to the left.

To learn more about docxtools, start with the vignettes: `browseVignettes(package = "docxtools")`.

Value

A data frame with numeric variables converted to formatted strings bounded by \dots .

Examples

```
# Factors unaffected; ambiguous trailing zeros.
data("CO2")
x <- as.data.frame(head(CO2, n = 5L))
format_engr(x)
format_engr(x, sigdig = c(0, 3))
format_engr(x, sigdig = c(3, 3), ambig_0_adj = TRUE)

# Ordered factor unaffected; ambiguous trailing zeros.
data("DNase")
x <- as.data.frame(tail(DNase, n = 5L))
format_engr(x)
format_engr(x, sigdig = c(6, 3))
format_engr(x, sigdig = c(6, 3), ambig_0_adj = TRUE)

# Integers returned unchanged but delimited; NA unchanged.
data("airquality")
x <- head(airquality, n = 6L)
format_engr(x, sigdig = 3)
```

put_axes

Insert a two-dimensional coordinate axes in a document.

Description

Creates an empty, 2-dimensional, coordinate axes using ggplot2: no scales, no tick marks, no axis labels.

Usage

```
put_axes(quadrant = NULL, col = NULL, size = NULL)
```

Arguments

quadrant : The quadrant to plot, 1, 2, 3, 4, or 0 for all four. Default is the first quadrant.
col : Line color, default is "gray60".
size : Line width, default is 0.5.

Details

The size of the figure is determined when printed, e.g., using `knitr` in an R Markdown script, the figure height in inches is set with the `fig.height` code chunk option.

Value

The empty axes.

Examples

```
put_axes()
put_axes(quadrant = 0)
put_axes(col = "red")
put_axes(size = 1)

## Not run:
# In an R Markdown script
```${r fig.height = 3.2}
put_axes()
```
## End(Not run)
```

put_gap

Insert a gap or whitespace in a document.

Description

Creates white space in a document by creating an empty plot in ggplot2. Border and fill can be specified to create a box. The defaults are transparent.

Usage

```
put_gap(col = NULL, fill = NULL)
```

Arguments

| | |
|------|----------------------------------------|
| col | Border color, default is "transparent" |
| fill | Fill color, default is "transparent" |

Details

Allows the author of Rmd to docx documents to insert a vertical white space of specified height, particularly useful in creating documents that users are expected to write in, such as workshop or student lab materials.

The dimensions of the gap are determined when it is printed, e.g., using knitr in an R Markdown script, the box height in inches is set with the fig.height code chunk option.

Value

Prints the box to the output document.

Examples

```
put_gap()
put_gap(col = "gray")
put_gap(col = "black", fill = "gray")
```

```
## Not run:
# In an R Markdown script
```${r fig.height = 0.75}
put_gap()
```
## End(Not run)
```

Index

*Topic **datasets**

density, [2](#)

density, [2](#)

docxtools, [2](#)

format_engr, [3](#)

put_axes, [4](#)

put_gap, [5](#)