

Package ‘formatdown’

November 21, 2022

Title Formatting Tools for 'rmarkdown' Documents

Version 0.1.1

Language en-US

Description Provides a small set of tools for formatting tasks when creating documents in 'rmarkdown' or 'quarto'. Convert the elements of a numerical vector to character strings in which the numbers are formatted using powers-of-ten notation in scientific or engineering form and delimited for rendering as inline equations.

Depends R (>= 3.5.0)

License MIT + file LICENSE

Encoding UTF-8

LazyData TRUE

LazyDataCompression bzip2

RoxygenNote 7.2.2

Imports checkmate, data.table, wrapr

Suggests covr, knitr, rmarkdown, tinytest

VignetteBuilder knitr

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

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

NeedsCompilation no

Author Richard Layton [aut, cre]

Maintainer Richard Layton <graphdoctor@gmail.com>

Repository CRAN

Date/Publication 2022-11-21 09:50:05 UTC

R topics documented:

density	2
format_power	2

Index	5
--------------	----------

density	<i>Ideal gas sample data frame</i>
---------	------------------------------------

Description

Air density at room temperature. A data frame to illustrate features of the `format_power()`.

Usage

```
data(density, package = "formatdown")
```

Format

A data frame with 5 rows and 7 columns:

date A date variable.

trial A character variable.

humidity A factor 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

format_power	<i>Format powers of ten</i>
--------------	-----------------------------

Description

Convert the elements of a numerical vector to character strings in which the numbers are formatted using powers-of-ten notation in scientific or engineering form and delimited for rendering as inline equations in an R Markdown document.

Usage

```
format_power(x, digits = 3, ..., format = "engr", omit_power = c(-1, 2))
```

Arguments

<code>x</code>	Numeric vector to be formatted.
<code>digits</code>	Numeric scalar, nonzero positive integer to specify the number of significant digits in the output coefficient.
<code>...</code>	Not used, force later arguments to be used by name.
<code>format</code>	Character. Possible values are "enr" (engineering notation) and "sci" (scientific notation). Use argument by name.
<code>omit_power</code>	Numeric vector $c(p, q)$ specifying the range of exponents over which power of ten notation is omitted, where $p \leq q$. If NULL all numbers are formatted in powers of ten notation. Use argument by name.

Details

Given a number, a numerical vector, or a numerical column from a data frame, `format_power()` converts the numbers to character strings of the form, " $a \times 10^n$ ", where a is the coefficient and n is the exponent. The string includes markup delimiters `$. . .$` for rendering as an inline equation in R Markdown or Quarto Markdown document. The user can specify the number of significant digits and scientific or engineering format.

Powers-of-ten notation is omitted over a range of exponents via `omit_power` such that numbers are converted to character strings of the form, " a ", where a is the number in decimal notation. The default `omit_power = c(-1, 2)` formats numbers such as 0.123, 1.23, 12.3, and 123 in decimal form. To cancel these exceptions and convert all numbers to powers-of-ten notation, set the `omit_power` argument to NULL.

Value

A character vector with the following properties:

- Numbers represented in powers of ten notation except for those with exponents in the range specified in `omit_power`
- Elements delimited with `$. . .$` for rendering as inline math in an R Markdown or Quarto Markdown document.

Examples

```
# Scalar value
format_power(101100)

# Vector value
x <- c(1.2222e-6, 2.3333e-5, 3.4444e-4, 4.1111e-3, 5.2222e-2, 6.3333e-1,
      7.4444e+0, 8.1111e+1, 9.2222e+2, 1.3333e+3, 2.4444e+4, 3.1111e+5, 4.2222e+6)
format_power(x)

# Compare significant digits
format_power(x[1], 3)
format_power(x[1], 4)

# Compare format type
```

```
format_power(x[3], format = "engr")
format_power(x[3], format = "sci")

# Compare omit_power range
format_power(x[6], omit_power = c(-1, 2))
format_power(x[6], omit_power = c(0, 2))
format_power(x[8])
format_power(x[8], omit_power = NULL)

# Apply to columns of a data frame (data.table syntax)
y <- x[1:6]
z <- x[8:13]
DT <- data.table::data.table(y, z)
DT[, lapply(.SD, function(x) format_power(x))]
```

Index

- * **datasets**

- density, [2](#)

- * **format_***

- format_power, [2](#)

density, [2](#)

format_power, [2](#)