

Package ‘flextable’

June 7, 2017

Type Package

Title Functions for Tabular Reporting

Version 0.2.0

Description Create pretty tables for 'Microsoft Word', 'Microsoft PowerPoint' and 'HTML' documents. Functions are provided to let users create tables, modify and format their content. It extends package 'officer' that does not contain any feature for customized tabular reporting. Function `tabwid()` produces an 'htmlwidget' ready to be used in 'Shiny' or 'R Markdown (*.Rmd)' documents.

License GPL-3

LazyData TRUE

Depends officer

Imports stats,utils,grDevices,
dplyr,lazyeval,purrr,tibble,magrittr,tidyr,R6,stringr,
htmltools,htmlwidgets, gdttools

RoxygenNote 6.0.1.9000

Suggests rmarkdown, knitr, xml2, testthat

URL <https://davidgohel.github.io/flextable>

BugReports <https://github.com/davidgohel/flextable/issues>

VignetteBuilder knitr

NeedsCompilation no

Author David Gohel [aut, cre]

Maintainer David Gohel <david.gohel@ardata.fr>

Repository CRAN

Date/Publication 2017-06-07 11:49:05 UTC

R topics documented:

<code>add_header</code>	2
<code>align</code>	3

as_image	4
autofit	5
bg	5
body_add_flextable	6
bold	6
border	7
color	8
dim.flextable	8
dim_pretty	9
display	9
empty_blanks	10
flextable	11
fontsize	12
height	12
italic	13
merge_at	13
merge_h	14
merge_none	14
merge_v	15
minibar	16
padding	16
ph_with_flextable	17
rotate	18
set_header_df	18
set_header_labels	19
style	20
tabwid	20
tabwid-shiny	21
theme_booktabs	22
theme_box	22
theme_tron	23
theme_tron_legacy	23
theme_vanilla	24
theme_zebra	24
void	25
width	25

Index 27

add_header	<i>Add a row of labels in headers</i>
------------	---------------------------------------

Description

Add a single row of labels in the flextable's header part. It can be inserted at the top or the bottom of header part.

Usage

```
add_header(x, top = TRUE, ...)
```

Arguments

`x` a flextable object

`top` should the row be inserted at the top or the bottom.

`...` a named list (names are data colnames) of strings specifying corresponding labels to add.

Details

An horizontal and a vertical merge are performed automatically. Use [merge_none](#) to drop these merging instructions.

Examples

```
ft <- flextable( head( iris ),
  col_keys = c("Species", "Sepal.Length", "Petal.Length", "Sepal.Width", "Petal.Width") )
ft <- add_header(x = ft, Sepal.Length = "length",
  Sepal.Width = "width", Petal.Length = "length",
  Petal.Width = "width", Species = "Species", top = FALSE )
ft <- add_header(ft, Sepal.Length = "Inches",
  Sepal.Width = "Inches", Petal.Length = "Inches",
  Petal.Width = "Inches", Species = "Species", top = TRUE )
ft <- merge_h(ft, part = "header")
ft
```

<code>align</code>	<i>Set text alignment</i>
--------------------	---------------------------

Description

change text alignment of selected rows and columns of a flextable.

Usage

```
align(x, i = NULL, j = NULL, align = "left", part = "body")
```

Arguments

`x` a flextable object

`i` rows selection

`j` columns selection

`align` text alignment - a single character value, expected value is one of 'left', 'right', 'center', 'justify'.

`part` partname of the table (one of 'all', 'body', 'header')

Examples

```
ft <- flextable(mtcars)
ft <- align(ft, align = "center")
```

as_image

image wrapper

Description

The function has to be used with function `display()`. It lets add images within flextable.

Usage

```
as_image(x, src, width = 1, height = 0.2)
```

Arguments

x	image will be repeated while iterating this variable. The variable should be one of the original data frame. Note its values are not used, only its size.
src	image filename
width, height	size of the png file in inches

See Also

[display](#)

Examples

```
library(magrittr)
img.file <- file.path( Sys.getenv("R_HOME"), "doc", "html", "logo.jpg" )
myft <- flextable(head( mtcars, n = 10))
myft <- myft %>%
  display(
    i = ~ qsec > 18, col_key = "qsec", pattern = "{{r_logo}}",
    formatters = list( r_logo ~ as_image(qsec,
      src = img.file, width = .20, height = .15)),
    fprops = list(qsec = fp_text(color = "orange")) )
myft
```

autofit	<i>Adjusts cell widths and heights</i>
---------	--

Description

compute and apply optimized widths and heights.

Usage

```
autofit(x, add_w = 0.1, add_h = 0.1)
```

Arguments

x	flextable object
add_w	extra width to add in inches
add_h	extra height to add in inches

Examples

```
ft <- flextable(mtcars)
ft <- autofit(ft)
ft
```

bg	<i>Set background color</i>
----	-----------------------------

Description

change background color of selected rows and columns of a flextable.

Usage

```
bg(x, i = NULL, j = NULL, bg, part = "body")
```

Arguments

x	a flextable object
i	rows selection
j	columns selection
bg	color to use as background color
part	partname of the table (one of 'all', 'body', 'header')

Examples

```
ft <- flextable(mtcars)
ft <- bg(ft, bg = "#DDDDDD", part = "header")
```

body_add_flextable *wml table code*

Description

produces the wml of a flextable

Usage

```
body_add_flextable(x, value, align = "center", pos = "after")
```

Arguments

x	a docx object
value	flextable object
align	left (default), center or right.
pos	where to add the flextable relative to the cursor, one of "after", "before", "on" (end of line).

Examples

```
library(officer)
ft <- flextable(head(mtcars))
ft <- theme_zebra(ft)
ft <- autofit(ft)
doc <- read_docx()
doc <- body_add_flextable(doc, value = ft)
print(doc, target = "test.docx")
```

bold *Set bold font*

Description

change font weight of selected rows and columns of a flextable.

Usage

```
bold(x, i = NULL, j = NULL, bold = TRUE, part = "body")
```

Arguments

x	a flextable object
i	rows selection
j	columns selection
bold	boolean value
part	partname of the table (one of 'all', 'body', 'header')

Examples

```
ft <- flextable(mtcars)
ft <- bold(ft, bold = TRUE, part = "header")
```

border	<i>Set cell borders</i>
--------	-------------------------

Description

change borders of selected rows and columns of a flextable.

Usage

```
border(x, i = NULL, j = NULL, border = NULL, border.top = NULL,
       border.bottom = NULL, border.left = NULL, border.right = NULL,
       part = "body")
```

Arguments

x	a flextable object
i	rows selection
j	columns selection
border	border (shortcut for top, bottom, left and right)
border.top	border top
border.bottom	border bottom
border.left	border left
border.right	border right
part	partname of the table (one of 'all', 'body', 'header')

Examples

```
ft <- flextable(mtcars)
ft <- border(ft, border.top = fp_border(color = "orange") )
```

color *Set font color*

Description

change font color of selected rows and columns of a flextable.

Usage

```
color(x, i = NULL, j = NULL, color, part = "body")
```

Arguments

x	a flextable object
i	rows selection
j	columns selection
color	color to use as font color
part	partname of the table (one of 'all', 'body', 'header')

Examples

```
ft <- flextable(mtcars)
ft <- color(ft, color = "orange", part = "header")
```

dim.flextable *Get flextable dimensions*

Description

returns widths and heights for each table columns and rows.

Usage

```
## S3 method for class 'flextable'
dim(x)
```

Arguments

x	flextable object
---	------------------

dim_pretty	<i>Calculate pretty dimensions</i>
------------	------------------------------------

Description

return minimum estimated widths and heights for each table columns and rows.

Usage

```
dim_pretty(x)
```

Arguments

x flextable object

Examples

```
ft <- flextable(mtcars)
dim_pretty(ft)
```

display	<i>Define flextable displayed values</i>
---------	--

Description

Modify flextable displayed values by specifying a string expression. Function is handling complex formatting as well as image insertion.

Usage

```
display(x, i = NULL, col_key, pattern, formatters = list(),
        fprops = list(), part = "body")
```

Arguments

x a flextable object
i rows selection
col_key column to modify, a single character
pattern string to format
formatters a list of formula, left side for the name, right side for the content.
fprops a named list of [fp_text](#)
part partname of the table (one of 'all', 'body', 'header')

Examples

```
# Formatting data values example -----
library(magrittr)
ft <- flextable(head( mtcars, n = 10))
ft <- display(ft, col_key = "carb",
  i = ~ drat > 3.5, pattern = "# {{carb}}",
  formatters = list(carb ~ sprintf("%.1f", carb)),
  fprops = list(carb = fp_text(color="orange") ) )
ft <- autofit(ft)
```

empty_blanks

make blank columns as transparent

Description

blank columns are set as transparent. This is a shortcut function that will delete top and bottom borders, change background color to transparent and display empty content.

Usage

```
empty_blanks(x)
```

Arguments

x a flextable object

Examples

```
library(magrittr)

typology <- data.frame(
  col_keys = c( "Sepal.Length", "Sepal.Width", "Petal.Length",
               "Petal.Width", "Species" ),
  what = c("Sepal", "Sepal", "Petal", "Petal", " " ),
  measure = c("Length", "Width", "Length", "Width", "Species"),
  stringsAsFactors = FALSE )
typology

head(iris) %>%
  flextable(
    col_keys = c("Species",
                "break1", "Sepal.Length", "Sepal.Width",
                "break2", "Petal.Length", "Petal.Width") ) %>%
  set_header_df(mapping = typology, key = "col_keys" ) %>%
  merge_h(part = "header") %>%
  theme_vanilla() %>%
  empty_blanks() %>%
  width(j = c(2, 5), width = .1 )
```

`flextable`*flextable creation*

Description

Create a flextable object with function `flextable`.

`flextable` are designed to make tabular reporting easier for R users. Functions are available to let you format text, paragraphs and cells; table cells can be merge vertically or horizontally, row headers can easily be defined, rows heights and columns widths can be manually set or automatically computed.

Usage

```
flextable(data, col_keys = names(data), cwidth = 0.75, cheight = 0.25)
```

```
## S3 method for class 'flextable'  
print(x, ...)
```

Arguments

<code>data</code>	dataset
<code>col_keys</code>	columns names/keys to display. If some column names are not in the dataset, they will be added as blank columns by default.
<code>cwidth, cheight</code>	initial width and height to use for cell sizes.
<code>x</code>	flextable object
<code>...</code>	unused argument

Details

A flextable is made of 2 parts: header and body.

Most functions have an argument named `part` that will be used to specify what part of the table should be modified.

Examples

```
ft <- flextable(mtcars)  
ft
```

fontsize	<i>Set font size</i>
----------	----------------------

Description

change font size of selected rows and columns of a flextable.

Usage

```
fontsize(x, i = NULL, j = NULL, size = 11, part = "body")
```

Arguments

x	a flextable object
i	rows selection
j	columns selection
size	integer value (points)
part	partname of the table (one of 'all', 'body', 'header')

Examples

```
ft <- flextable(mtcars)
ft <- fontsize(ft, size = 14, part = "header")
```

height	<i>Set flextable rows height</i>
--------	----------------------------------

Description

control rows height

Usage

```
height(x, i = NULL, height, part = "body")
```

Arguments

x	flextable object
i	rows selection
height	height in inches
part	partname of the table

Examples

```
ft <- flextable(iris)
ft <- height(ft, height = .3)
```

italic	<i>Set italic font</i>
--------	------------------------

Description

change font decoration of selected rows and columns of a flextable.

Usage

```
italic(x, i = NULL, j = NULL, italic = TRUE, part = "body")
```

Arguments

x	a flextable object
i	rows selection
j	columns selection
italic	boolean value
part	partname of the table (one of 'all', 'body', 'header')

Examples

```
ft <- flextable(mtcars)
ft <- italic(ft, italic = TRUE, part = "header")
```

merge_at	<i>Merge flextable cells</i>
----------	------------------------------

Description

Merge flextable cells

Usage

```
merge_at(x, i = NULL, j = NULL, part = "body")
```

Arguments

x	flextable object
i, j	columns and rows to merge
part	partname of the table where merge has to be done.

Examples

```
library(officer)
library(magrittr)

ft_merge <- head( mtcars ) %>%
  flextable( cwidth = .5 ) %>%
  merge_at( i = 1:2, j = 1:3 )
ft_merge
```

merge_h	<i>Merge flextable cells horizontally</i>
---------	---

Description

Merge flextable cells horizontally when consecutive cells have identical values.

Usage

```
merge_h(x, i = NULL, part = "body")
```

Arguments

x	flextable object
i	rows where cells have to be merged.
part	partname of the table where merge has to be done.

Examples

```
dummy_df <- data.frame( col1 = letters,
  col2 = letters, stringsAsFactors = FALSE )
ft_merge <- flextable(dummy_df)
ft_merge <- merge_h(x = ft_merge)
ft_merge
```

merge_none	<i>Delete flextable merging informations</i>
------------	--

Description

Delete all merging informations from a flextable.

Usage

```
merge_none(x, part = "all")
```

Arguments

x flextable object
 part partname of the table where merge has to be done.

Examples

```
typology <- data.frame(
  col_keys = c( "Sepal.Length", "Sepal.Width", "Petal.Length", "Petal.Width", "Species" ),
  what = c("Sepal", "Sepal", "Petal", "Petal", "Species"),
  measure = c("Length", "Width", "Length", "Width", "Species"),
  stringsAsFactors = FALSE )

ft <- flextable( head( iris ) )
ft <- set_header_df(ft, mapping = typology, key = "col_keys" )
ft <- merge_v(ft, j = c("Species"))

ft <- theme_tron_legacy( merge_none( ft ) )
ft
```

merge_v *Merge flextable cells vertically*

Description

Merge flextable cells vertically when consecutive cells have identical values.

Usage

```
merge_v(x, j = NULL, part = "body")
```

Arguments

x flextable object
 j columns names/keys where cells have to be merged.
 part partname of the table where merge has to be done.

Examples

```
ft_merge <- flextable(mtcars)
ft_merge <- merge_v(ft_merge, j = c("gear", "carb"))
ft_merge
```

minibar	<i>draw a single bar</i>
---------	--------------------------

Description

This function is used to insert bars into flextable with function display

Usage

```
minibar(value, max, barcol = "#CCCCCC", bg = "transparent", width = 1,
        height = 0.2)
```

Arguments

value	bar height
max	max bar height
barcol	bar color
bg	background color
width, height	size of the resulting png file in inches

padding	<i>Set paragraph paddings</i>
---------	-------------------------------

Description

change paddings of selected rows and columns of a flextable.

Usage

```
padding(x, i = NULL, j = NULL, padding = NULL, padding.top = NULL,
        padding.bottom = NULL, padding.left = NULL, padding.right = NULL,
        part = "body")
```

Arguments

x	a flextable object
i	rows selection
j	columns selection
padding	padding (shortcut for top, bottom, left and right)
padding.top	padding top
padding.bottom	padding bottom
padding.left	padding left
padding.right	padding right
part	partname of the table (one of 'all', 'body', 'header')

Examples

```
ft <- flextable(mtcars)
ft <- padding(ft, padding.top = 4)
```

ph_with_flextable *add flextable into a PowerPoint slide*

Description

add a flextable as a new shape in the current slide.

Usage

```
ph_with_flextable(x, value, type, index = 1)
ph_with_flextable_at(x, value, left, top)
```

Arguments

x	a pptx device
value	flextable object
type	placeholder type
index	placeholder index (integer). This is to be used when a placeholder type is not unique in the current slide, e.g. two placeholders with type 'body'.
left, top	location of flextable on the slide

Examples

```
library(officer)
ft <- flextable(head(mtcars))

doc <- read_pptx()
doc <- add_slide(doc, layout = "Title and Content",
                master = "Office Theme")
doc <- ph_with_flextable(doc, value = ft, type = "body")
doc <- ph_with_flextable_at(doc, value = ft, left = 4, top = 5)
print(doc, target = "test.pptx" )
```

rotate	<i>rotate cell text</i>
--------	-------------------------

Description

apply a rotation to cell text

Usage

```
rotate(x, i = NULL, j = NULL, rotation, align = "center", part = "body")
```

Arguments

x	a flextable object
i	rows selection
j	columns selection
rotation	one of "lrtb", "tblr", "btlr"
align	one of "center" or "top" or "bottom"
part	partname of the table (one of 'all', 'body', 'header')

Examples

```
ft <- flextable(mtcars)
ft <- rotate(ft, rotation = "lrtb", align = "top", part = "header")
```

set_header_df	<i>Set flextable's header rows</i>
---------------	------------------------------------

Description

Use a data.frame to specify flextable's header rows.

The data.frame must contain a column whose values match flextable col_keys argument, this column will be used as join key. The other columns will be displayed as header rows. The leftmost column is used as the top header row and the rightmost column is used as the bottom header row.

Usage

```
set_header_df(x, mapping = NULL, key = "col_keys")
```

Arguments

x	a flextable object
mapping	a data.frame specifying for each colname content of the column.
key	column to use as key when joining data_mapping.

Examples

```
typology <- data.frame(
  col_keys = c( "Sepal.Length", "Sepal.Width", "Petal.Length",
               "Petal.Width", "Species" ),
  what = c("Sepal", "Sepal", "Petal", "Petal", "Species"),
  measure = c("Length", "Width", "Length", "Width", "Species"),
  stringsAsFactors = FALSE )

ft <- flextable( head( iris ))
ft <- set_header_df(ft, mapping = typology, key = "col_keys" )
ft <- merge_h(ft, part = "header")
ft <- merge_v(ft, j = "Species", part = "header")
ft <- theme_vanilla(ft)
ft
```

set_header_labels *Set flextable's headers labels*

Description

This function set labels for specified columns in a single row header of a flextable.

Usage

```
set_header_labels(x, ...)
```

Arguments

x a flextable object

... a named list (names are data colnames), each element is a single character value specifying label to use.

Examples

```
ft_1 <- flextable( head( iris ))
ft_1 <- set_header_labels(ft_1, Sepal.Length = "Sepal length",
  Sepal.Width = "Sepal width", Petal.Length = "Petal length",
  Petal.Width = "Petal width"
)
ft_1
```

style	<i>Set flextable style</i>
-------	----------------------------

Description

Modify flextable text, paragraphs and cells formatting properties.

Usage

```
style(x, i = NULL, j = NULL, pr_t = NULL, pr_p = NULL, pr_c = NULL,
      part = "body")
```

Arguments

x	a flextable object
i	rows selection
j	columns selection
pr_t	object(s) of class fp_text
pr_p	object(s) of class fp_par
pr_c	object(s) of class fp_cell
part	partname of the table (one of 'all', 'body', 'header')

Examples

```
def_cell <- fp_cell(border = fp_border(color="#00FFFF"))

def_par <- fp_par(text.align = "center")

ft <- flextable(mtcars)

ft <- style( ft, pr_c = def_cell, pr_p = def_par, part = "all")
ft <- style(ft, ~ drat > 3.5, ~ vs + am + gear + carb,
           pr_t = fp_text(color="red", italic = TRUE) )

ft
```

tabwid	<i>htmlwidget for flextable</i>
--------	---------------------------------

Description

htmlwidget for flextable

Usage

```
tabwid(x, width = NULL, height = NULL)
```

Arguments

x	flextable object
width	widget width. Unused argument, columns widths are managed by function width.
height	widget height Unused argument, rows heights are managed by function height.

 tabwid-shiny

Shiny bindings for tabwid

Description

Output and render functions for using tabwid within Shiny applications and interactive Rmd documents.

Usage

```
tabwidOutput(outputId, width = "100%", height = "400px")
```

```
rendertabwid(expr, env = parent.frame(), quoted = FALSE)
```

Arguments

outputId	output variable to read from
width, height	Must be a valid CSS unit (like '100%', '400px', 'auto') or a number, which will be coerced to a string and have 'px' appended.
expr	An expression that generates a tabwid
env	The environment in which to evaluate expr.
quoted	Is expr a quoted expression (with quote())? This is useful if you want to save an expression in a variable.

theme_booktabs	<i>Apply booktabs theme</i>
----------------	-----------------------------

Description

Apply theme tron to a flextable

Usage

```
theme_booktabs(x)
```

Arguments

x a flextable object

Examples

```
ft <- flextable(iris)
ft <- theme_booktabs(ft)
```

theme_box	<i>Apply box theme</i>
-----------	------------------------

Description

Apply theme box to a flextable

Usage

```
theme_box(x)
```

Arguments

x a flextable object

Examples

```
ft <- flextable(iris)
ft <- theme_box(ft)
```

theme_tron	<i>Apply tron theme</i>
------------	-------------------------

Description

Apply theme tron to a flextable

Usage

```
theme_tron(x)
```

Arguments

x a flextable object

Examples

```
ft <- flextable(iris)
ft <- theme_tron(ft)
```

theme_tron_legacy	<i>Apply tron legacy theme</i>
-------------------	--------------------------------

Description

Apply theme tron legacy to a flextable

Usage

```
theme_tron_legacy(x)
```

Arguments

x a flextable object

Examples

```
ft <- flextable(iris)
ft <- theme_tron_legacy(ft)
```

theme_vanilla	<i>Apply vanilla theme</i>
---------------	----------------------------

Description

Apply theme vanilla to a flextable

Usage

```
theme_vanilla(x)
```

Arguments

x a flextable object

Examples

```
ft <- flextable(iris)
ft <- theme_vanilla(ft)
```

theme_zebra	<i>Apply zebra theme</i>
-------------	--------------------------

Description

Apply theme zebra to a flextable

Usage

```
theme_zebra(x, odd_header = "#CFCFCF", odd_body = "#EFEFEF",
  even_header = "transparent", even_body = "transparent")
```

Arguments

x a flextable object
odd_header, odd_body, even_header, even_body
 odd/even colors for table header and body

Examples

```
ft <- flextable(iris)
ft <- theme_zebra(ft)
```

void	<i>Delete flextable content</i>
------	---------------------------------

Description

Deletion of flextable content. Values will be replaced with NA values.

Usage

```
void(x, i = NULL, j = NULL, part = "body")
```

Arguments

x	flextable object
i	rows selection
j	columns selection
part	partname of the table

Examples

```
ft <- flextable(mtcars)
ft <- void(ft, ~ drat > 3.5, ~ vs + am + gear + carb )
```

width	<i>Set flextable columns width</i>
-------	------------------------------------

Description

control columns width

Usage

```
width(x, j = NULL, width)
```

Arguments

x	flextable object
j	columns selection
width	width in inches

Details

Heights are not used when flextable is been rendered into HTML.

See Also[flextable](#)**Examples**

```
ft <- flextable(iris)
ft <- width(ft, width = 1)
```

Index

add_header, 2
align, 3
as_image, 4
autofit, 5

bg, 5
body_add_flextable, 6
bold, 6
border, 7

color, 8

dim.flextable, 8
dim_pretty, 9
display, 4, 9

empty_blanks, 10

flextable, 11, 26
fontsize, 12
fp_text, 9

height, 12

italic, 13

merge_at, 13
merge_h, 14
merge_none, 3, 14
merge_v, 15
minibar, 16

padding, 16
ph_with_flextable, 17
ph_with_flextable_at
 (ph_with_flextable), 17
print.flextable (flextable), 11

rendertabwid (tabwid-shiny), 21
rotate, 18

set_header_df, 18

set_header_labels, 19
style, 20

tabwid, 20
tabwid-shiny, 21
tabwidOutput (tabwid-shiny), 21
theme_booktabs, 22
theme_box, 22
theme_tron, 23
theme_tron_legacy, 23
theme_vanilla, 24
theme_zebra, 24

void, 25

width, 25