

Package: flexlsx (via r-universe)

September 11, 2024

Type Package

Title Exporting 'flextable' to 'xlsx' Files

Version 0.2.2

Description Exports 'flextable' objects to 'xlsx' files, utilizing functionalities provided by 'flextable' and 'openxlsx2'.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports dplyr (>= 1.1.1), grDevices, openxlsx2 (>= 1.0.0), purrr, rlang, stringi, tibble, tidyr (>= 1.0.0)

Suggests covr, gtsummary, flextable (>= 0.9.5), testthat (>= 3.0.0)

Depends R (>= 4.1.0)

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Config/testthat/edition 3

URL <https://github.com/pteridin/flexlsx>

BugReports <https://github.com/pteridin/flexlsx/issues>

Repository <https://pteridin.r-universe.dev>

RemoteUrl <https://github.com/pteridin/flexlsx>

RemoteRef HEAD

RemoteSha 7eb31eba8e7e727206658f104449fbfed24c874d

Contents

ftpart_to_style_tibble	2
ft_to_style_tibble	3
ft_to_xlsx_border_width	3
get_dim_colwise	4
get_dim_ranges	4

get_dim_rowwise	5
handle_null_border	5
prepare_color	6
style_to_hash	6
wb_add_caption	7
wb_add_flextable	7
wb_apply_border	8
wb_apply_cell_styles	9
wb_apply_content	9
wb_apply_merge	10
wb_apply_text_styles	10
wb_change_cell_width	11
wb_change_row_height	11

Index	12
--------------	-----------

ftpart_to_style_tibble

Converts a flextable-part to a tibble styles

Description

[Experimental]

[Experimental]

Usage

```
ftpart_to_style_tibble(ft_part, part = c("header", "body", "footer"))
```

Arguments

ft_part the part of the flextable to extract the style from

part the name of the part

Value

a [tibble](#)

ft_to_style_tibble *Converts a flextable to a tibble with style information*

Description

[Experimental]

Usage

```
ft_to_style_tibble(  
  ft,  
  offset_rows = 0L,  
  offset_cols = 0L,  
  offset_caption_rows = 0L  
)
```

Arguments

ft a [flextable](#)
offset_rows offsets the start-row
offset_cols offsets the start-columns
offset_caption_rows number of rows to offset the caption by

Value

a [tibble](#)

ft_to_xlsx_border_width
 Determines the border width

Description

[Experimental]

Usage

```
ft_to_xlsx_border_width(border_width)
```

Arguments

border_width a numeric vector determining the border-width

Value

a factor of xlsx border styles

get_dim_colwise	<i>Groups each row with same style each column</i>
-----------------	--

Description**[Experimental]****Usage**

```
get_dim_colwise(df_rows)
```

Arguments

df_rows [tibble](#) of row-wise aggregates style

Value

[tibble](#) of column-wise aggregates style

get_dim_ranges	<i>Retrieves dims of same style rows within same column</i>
----------------	---

Description**[Experimental]****Usage**

```
get_dim_ranges(df_x)
```

Arguments

df_x styling information incl. col_id & row_id

Value

merged styles as a [tibble](#)

get_dim_rowwise	<i>Groups each column with same style each row</i>
-----------------	--

Description

[Experimental]

Usage

```
get_dim_rowwise(df_x, df_style_hashed)
```

Arguments

df_x styling information incl. col_id & row_id
df_style_hashed [tibble](#) of hashed style information

Value

[tibble](#) of row-wise aggregates style information

handle_null_border	<i>Where there is no border return NULL</i>
--------------------	---

Description

[Experimental]

Usage

```
handle_null_border(border_width)
```

Arguments

border_width a numeric vector determining the border-width

Value

border_width or NULL

prepare_color	<i>Prepares the color for content style</i>
---------------	---

Description

Converts a color name to the hexadecimal RGB-value Removes "transparent" color

Usage

```
prepare_color(color_name)
```

Arguments

color_name	The name of the color
------------	-----------------------

Value

The hexadecimal RGB-value

style_to_hash	<i>Retrieves hashed style information</i>
---------------	---

Description**[Experimental]**

Converts each style to an individual integer hash for easy comparison and aggregation.

Usage

```
style_to_hash(df_x)
```

Arguments

df_x	styling information incl. col_id & row_id
------	---

Value

hashed style information as a [tibble](#)

wb_add_caption	<i>Adds a caption to an excel file</i>
----------------	--

Description

[Experimental]

Usage

```
wb_add_caption(  
    wb,  
    sheet,  
    ft,  
    offset_rows = offset_rows,  
    offset_cols = offset_cols  
)
```

Arguments

wb	an openxlsx2 workbook
sheet	an openxlsx2 workbook sheet
ft	a flextable
offset_rows	zero-based row offset
offset_cols	zero-based column offset

wb_add_flextable	<i>Adds a flextable to an openxlsx2 workbook sheet</i>
------------------	--

Description

[Experimental]

Usage

```
wb_add_flextable(  
    wb,  
    sheet,  
    ft,  
    start_col = 1,  
    start_row = 1,  
    offset_caption_rows = 0L,  
    dims = NULL  
)
```

Arguments

wb	an openxlsx2 workbook
sheet	an openxlsx2 workbook sheet
ft	a flextable
start_col	a vector specifying the starting column to write to.
start_row	a vector specifying the starting row to write to.
offset_caption_rows	number of rows to offset the caption by
dims	Spreadsheet dimensions that will determine start_col and start_row: "A1", "A1:B2", "A:B"

Value

an openxlsx2 workbook

Examples

```
if(requireNamespace("flextable", quietly = TRUE)) {
  # Create a flextable
  ft <- flextable::as_flextable(table(mtcars[,c("am", "cyl")]))

  # Create a workbook
  wb <- openxlsx2::wb_workbook()$add_worksheet("mtcars")

  # Add flextable to workbook
  wb <- wb_add_flextable(wb, "mtcars", ft)

  # Workbook can now be saved wb$save(),
  # opened wb$open() - or removed
  rm(wb)
}
```

wb_apply_border

Applies the border styles

Description

[Experimental]

Usage

```
wb_apply_border(wb, sheet, df_style)
```


Arguments

<code>wb</code>	the workbook
<code>sheet</code>	the sheet of the workbook
<code>df_style</code>	the styling tibble from ft_to_style_tibble

`wb_apply_cell_styles` *Applies the cell styles*

Description

[Experimental]

Usage

```
wb_apply_cell_styles(wb, sheet, df_style)
```

Arguments

<code>wb</code>	the workbook
<code>sheet</code>	the sheet of the workbook
<code>df_style</code>	the styling tibble from ft_to_style_tibble

`wb_apply_content` *Applies the content*

Description

[Experimental]

Usage

```
wb_apply_content(wb, sheet, df_style)
```

Arguments

<code>wb</code>	the workbook
<code>sheet</code>	the sheet of the workbook
<code>df_style</code>	the styling tibble from ft_to_style_tibble

wb_apply_merge *Merges cells*

Description

[Experimental]

Usage

```
wb_apply_merge(wb, sheet, df_style)
```

Arguments

wb	the workbook
sheet	the sheet of the workbook
df_style	the styling tibble from ft_to_style_tibble

Value

df_style tibble

wb_apply_text_styles *Applies the text styles*

Description

[Experimental]

Usage

```
wb_apply_text_styles(wb, sheet, df_style)
```

Arguments

wb	the workbook
sheet	the sheet of the workbook
df_style	the styling tibble from ft_to_style_tibble

`wb_change_cell_width` *Changes the cell width*

Description

Changes the cell width

Usage

`wb_change_cell_width(wb, sheet, ft, offset_cols)`

Arguments

<code>wb</code>	an openxlsx2 workbook
<code>sheet</code>	an openxlsx2 workbook sheet
<code>ft</code>	a flextable
<code>offset_cols</code>	zero-based column offset

`wb_change_row_height` *Changes the row height*

Description

Changes the row height

Usage

`wb_change_row_height(wb, sheet, df_style)`

Arguments

<code>wb</code>	an openxlsx2 workbook
<code>sheet</code>	an openxlsx2 workbook sheet
<code>df_style</code>	the styling tibble from ft_to_style_tibble

Index

flextable, 3
ft_to_style_tibble, 3, 9–11
ft_to_xlsx_border_width, 3
ftpart_to_style_tibble, 2

get_dim_colwise, 4
get_dim_ranges, 4
get_dim_rowwise, 5

handle_null_border, 5

prepare_color, 6

style_to_hash, 6

tibble, 2–6

wb_add_caption, 7
wb_add_flextable, 7
wb_apply_border, 8
wb_apply_cell_styles, 9
wb_apply_content, 9
wb_apply_merge, 10
wb_apply_text_styles, 10
wb_change_cell_width, 11
wb_change_row_height, 11
workbook, 9, 10