

# Package ‘flexlsx’

May 7, 2024

**Type** Package

**Title** Exporting 'flextable' to 'xlsx' Files

**Version** 0.2.1

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

**License** MIT + file LICENSE

**Encoding** UTF-8

**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)

**RoxygenNote** 7.3.1

**Config/testthat/edition** 3

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

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

**NeedsCompilation** no

**Author** Tobias Heidler [aut, cre, cph]  
(<https://orcid.org/0000-0001-9193-0980>)

**Maintainer** Tobias Heidler <tobias.heidler@googlemail.com>

**Repository** CRAN

**Date/Publication** 2024-05-07 07:50:02 UTC

## R topics documented:

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

<b>Index</b>	<b>12</b>
--------------	-----------

---

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 <a href="#">workbook</a>
<code>sheet</code>	the sheet of the workbook
<code>df_style</code>	the styling tibble from <a href="#">ft_to_style_tibble</a>

---

`wb_apply_cell_styles` *Applies the cell styles*

---

### Description

**[Experimental]**

### Usage

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

### Arguments

<code>wb</code>	the <a href="#">workbook</a>
<code>sheet</code>	the sheet of the workbook
<code>df_style</code>	the styling tibble from <a href="#">ft_to_style_tibble</a>

---

`wb_apply_content` *Applies the content*

---

### Description

**[Experimental]**

### Usage

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

### Arguments

<code>wb</code>	the <a href="#">workbook</a>
<code>sheet</code>	the sheet of the workbook
<code>df_style</code>	the styling tibble from <a href="#">ft_to_style_tibble</a>

---

wb\_apply\_merge      *Merges cells*

---

### Description

**[Experimental]**

### Usage

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

### Arguments

wb	the <a href="#">workbook</a>
sheet	the sheet of the workbook
df_style	the styling tibble from <a href="#">ft_to_style_tibble</a>

### 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 <a href="#">workbook</a>
sheet	the sheet of the workbook
df_style	the styling tibble from <a href="#">ft_to_style_tibble</a>

---

wb\_change\_cell\_width *Changes the cell width*

---

**Description**

Changes the cell width

**Usage**

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

**Arguments**

wb	an openxlsx2 workbook
sheet	an openxlsx2 workbook sheet
ft	a flextable
offset_cols	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**

wb	an openxlsx2 workbook
sheet	an openxlsx2 workbook sheet
df_style	the styling tibble from <a href="#">ft_to_style_tibble</a>

# 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