

Package ‘ftExtra’

March 28, 2021

Title Extensions for 'Flextable'

Version 0.2.0

Date 2021-03-28

Maintainer Atsushi Yasumoto <atusy.rpkg@gmail.com>

Description Build display tables easily by extending the functionality of the 'flextable' package. Features include spanning header, grouping rows, parsing markdown and so on.

License MIT + file LICENSE

URL <https://ftextra.atusy.net>, <https://github.com/atusy/ftExtra>

BugReports <https://github.com/atusy/ftExtra/issues>

Imports dplyr, jsonlite, flextable (>= 0.6.4), tidyr, purrr, magrittr, rmarkdown, rlang, stringr, tibble, tidyselect (>= 1.1.0), xfun, yaml, vctrs

Suggests testthat (>= 2.1.0), knitr, qpdf

VignetteBuilder knitr

Encoding UTF-8

Language en-US

RoxygenNote 7.1.1

SystemRequirements pandoc (>= 2.0.6) - <http://pandoc.org>

NeedsCompilation no

Author Atsushi Yasumoto [aut, cph, cre]
(<<https://orcid.org/0000-0002-8335-495X>>),
David Gohel [ctb],
Romain François [ctb] (<<https://orcid.org/0000-0002-2444-4226>>)

Repository CRAN

Date/Publication 2021-03-28 12:40:02 UTC

R topics documented:

as_flextable_methods	2
as_paragraph_md	3
colformat_md	4
footnote_options	5
separate_header	6
span_header	7
with_blanks	7

Index	9
--------------	----------

as_flextable_methods *method to convert object to flextable*

Description

This is a convenient function to let users create flextable bindings from any objects.

Usage

```
## S3 method for class 'grouped_df'
as_flextable(
  x,
  groups_to = c("titles", "merged", "asis"),
  groups_pos = c("left", "asis"),
  ...
)

## S3 method for class 'data.frame'
as_flextable(x, col_keys = names(x), ...)
```

Arguments

x	object to be transformed as flextable
groups_to	One of titles, merged, or asis. See examples for the result.
groups_pos	When groups_to = "merged", grouping columns are reordered according to group_pos. Choices are left (default) or asis.
...	arguments for custom methods
col_keys	columns names/keys to display. If some column names are not in the dataset, they will be added as blank columns by default.

See Also

Other as_flextable methods: [as_flextable.gam\(\)](#), [as_flextable.glm\(\)](#), [as_flextable.grouped_data\(\)](#), [as_flextable.htest\(\)](#), [as_flextable.lm\(\)](#), [as_flextable.xtable\(\)](#)

Examples

```
# For grouped_df
grouped_df <- iris %>%
  dplyr::group_by(Species) %>%
  dplyr::slice(1, 2)

as_flextable(grouped_df, groups_to = "titles")
as_flextable(grouped_df, groups_to = "titles", hide_grouplabel = TRUE)
as_flextable(grouped_df, groups_to = "merged")
as_flextable(grouped_df, groups_to = "asis")
# For data.frame
iris %>%
  head() %>%
  as_flextable()
```

as_paragraph_md	<i>Convert a character vector into markdown paragraph(s)</i>
-----------------	--

Description

Parse markdown cells and returns the "paragraph" object.

Usage

```
as_paragraph_md(
  x,
  auto_color_link = "blue",
  md_extensions = NULL,
  pandoc_args = NULL,
  .from = "markdown+autolink_bare_uris",
  .footnote_options = NULL,
  ...
)
```

Arguments

x	A character vector.
auto_color_link	A color of the link texts.
md_extensions	Pandoc's extensions. Although it is prefixed with "md", extensions for any formats specified to .from can be used. See https://www.pandoc.org/MANUAL.html#extensions for details.
pandoc_args	Additional command line options to pass to pandoc
.from	Pandoc's --from argument (default: 'markdown+autolink_bare_uris').
.footnote_options	Options for footnotes generated by footnote_options().
...	Arguments passed to internal functions.

Examples

```

if (rmarkdown::pandoc_available("2.0.6")) {
  library(flextable)
  ft <- flextable(
    data.frame(
      x = c("**foo** bar", "***baz***", "*qux*"),
      stringsAsFactors = FALSE
    )
  )
  ft <- compose(ft, j = "x", i = 1:2, value = as_paragraph_md(x))
  autofit(ft)
}

```

colformat_md

*Format character columns as markdown text***Description**

Format character columns as markdown text

Usage

```

colformat_md(
  x,
  j = where(is.character),
  part = c("body", "header", "all"),
  auto_color_link = "blue",
  md_extensions = NULL,
  pandoc_args = NULL,
  .from = "markdown+autolink_bare_uris",
  .footnote_options = footnote_options(),
  .sep = "\n\n"
)

```

Arguments

x	A flextable object
j	Columns to be treated as markdown texts. Selection can be done by the semantics of <code>dplyr::select()</code> .
part	One of "body", "header", and "all". If "all", formatting proceeds in the order of "header" and "body".
auto_color_link	A color of the link texts.
md_extensions	Pandoc's extensions. Although it is prefixed with "md", extensions for any formats specified to <code>.from</code> can be used. See https://www.pandoc.org/MANUAL.html#extensions for details.

pandoc_args Additional command line options to pass to pandoc
 .from Pandoc's --from argument (default: 'markdown+autolink_bare_uris').
 .footnote_options Options for footnotes generated by footnote_options().
 .sep A separator of paragraphs (default: "\n\n")

Examples

```

if (rmarkdown::pandoc_available("2.0.6")) {
  d <- data.frame(
    x = c("**bold**", "*italic*"),
    y = c("^superscript^", "~subscript~"),
    z = c("***^ft^~Extra~** is*", "*Cool*")
  )
  colformat_md(flextable::flextable(d))
}

```

footnote_options *Options for footnotes*

Description

Configure options for footnotes.

Usage

```

footnote_options(
  ref = c("1", "a", "A", "i", "I", "*"),
  prefix = "",
  suffix = "",
  start = 1L,
  max = 26L,
  inline = FALSE,
  sep = "; "
)

```

Arguments

ref One of "1", "a", "A", "i", "I", or "*" to as a choice for a symbol to cross-reference footnotes.
 prefix, suffix Pre- and suf-fixes for ref (default: "").
 start A starting number of footnotes.
 max A max number of footnotes.
 inline whether to add footnote on same line as previous footnote or not
 sep inline = T, character string to use as a separator between footnotes

Value

An environment

Examples

```
o <- footnote_options("1", start = 1L)
```

separate_header	<i>Separate the header based on delimiters</i>
-----------------	--

Description

Separate the header based on delimiters

Usage

```
separate_header(  
  x,  
  sep = "[_\\.]",  
  theme_fun = flextable::theme_booktabs,  
  ...  
)
```

Arguments

x	A flextable object ⁴
sep	Separator between columns. If character, sep is interpreted as a regular expression. The default value is a regular expression that matches any sequence of non-alphanumeric values. If numeric, sep is interpreted as character positions to split at. Positive values start at 1 at the far-left of the string; negative value start at -1 at the far-right of the string. The length of sep should be one less than into.
theme_fun	deprecated, use set_flextable_defaults() instead.
...	Passed to theme_fun

Examples

```
iris %>%  
  as_flextable() %>%  
  separate_header()
```

span_header	<i>Span the header based on delimiters</i>
-------------	--

Description

Span the header based on delimiters

Usage

```
span_header(x, sep = "[_\\.]", theme_fun = flextable::theme_booktabs, ...)
```

Arguments

x	A flextable object‘
sep	Separator between columns. If character, sep is interpreted as a regular expression. The default value is a regular expression that matches any sequence of non-alphanumeric values. If numeric, sep is interpreted as character positions to split at. Positive values start at 1 at the far-left of the string; negative value start at -1 at the far-right of the string. The length of sep should be one less than into.
theme_fun	deprecated, use set_flextable_defaults() instead.
...	Passed to theme_fun

Examples

```
iris %>%
  as_flextable() %>%
  span_header()
```

with_blanks	<i>Specify blank columns easily via col_keys</i>
-------------	--

Description

Specify blank columns easily via col_keys

Usage

```
with_blanks(after = NULL, before = NULL)
```

Arguments

after, before	Blank columns are added after/before the selected columns. Selections can be done by the semantics of <code>dplyr::select</code> .
---------------	--

Examples

```
iris %>%  
  as_flextable(col_keys = with_blanks(dplyr::ends_with("Width")))
```


Index

`as_flextable.data.frame`
 (`as_flextable_methods`), 2
`as_flextable.gam`, 2
`as_flextable.glm`, 2
`as_flextable.grouped_data`, 2
`as_flextable.grouped_df`
 (`as_flextable_methods`), 2
`as_flextable.htest`, 2
`as_flextable.lm`, 2
`as_flextable.xtable`, 2
`as_flextable_methods`, 2
`as_paragraph_md`, 3

`colformat_md`, 4

`footnote_options`, 5

`separate_header`, 6
`set_flextable_defaults()`, 6, 7
`span_header`, 7

`with_blanks`, 7