

Package: slideutils (via r-universe)

March 5, 2025

Type Package

Title R Markdown Slide Utilities

Version 0.1.4

Date 2023-03-09

BugReports <https://github.com/ucanr-igis/slideutils/issues>

URL <https://github.com/ucanr-igis/slideutils>

Description Utilities for creating R Markdown Slides.

License GPL (>= 3)

Depends R (>= 3.6)

Imports magrittr, dplyr, htmltools, crayon, knitr, ggplot2, magick,
googlesheets4, rmarkdown

Suggests leaflet, urlshorteneR, testthat (>= 3.0.0)

Encoding UTF-8

LazyData true

Roxygen list(markdown = FALSE)

RoxygenNote 7.2.3

VignetteBuilder knitr

Config/testthat/edition 3

Config/pak/sysreqs make libmagick++-dev gsfonts libssl-dev

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

RemoteUrl <https://github.com/ucanr-igis/slideutils>

RemoteRef HEAD

RemoteSha 51ea111ab32bb3ebeb3f14ffc909ce755ac49d2

Contents

su_gfr_elems	2
su_gfr_report	3
su_gfr_snippet	3
su_gfr_taglist	4
su_img_build	4
su_img_maketrans	6
su_meme	7
su_tips	8
Index	10

su_gfr_elems	<i>Generate elements for a Google Form Summary</i>
--------------	--

Description

Generate elements for a Google Form Summary

Usage

```
su_gfr_elems(resp_tbl, rpt_tbl, show_msg = TRUE)
```

Arguments

resp_tbl	Responses tibble
rpt_tbl	Report template
show_msg	Show additional messages

Details

This function takes two tibbles: i) the responses from a Google Form Survey (or any kind of survey, where the format of the responses mirrors that of Google Forms), and ii) a tibble that specifies how the questions should be summarized and presented in the output report.

This function does not download any data from Google Sheets. It is presumed you have already done that. It also doesn't do anything terribly useful with the summaries. It is presumed you'll pass the object returned by this function to su_gfr_report(), or a similar function that renders or analyzes the results.

For an example of rpt_tbl, please see [here](#).

`su_gfr_report`*Generate a complete Google Form Summary HTML Report*

Description

Generate a complete Google Form Summary HTML Report

Usage

```
su_gfr_report(  
  gfr_elems,  
  output_fn,  
  output_dir = ".",  
  rpt_title = "Google Form Summary",  
  rpt_rmd = NULL  
)
```

Arguments

<code>gfr_elems</code>	Google Form Summary Elements object
<code>output_fn</code>	Report filename
<code>output_dir</code>	Output directory
<code>rpt_title</code>	The title for the report
<code>rpt_rmd</code>	The Rmd file for the report

See Also

[su_gfr_elems](#)

`su_gfr_snippet`*Generate a snippet of HTML code for summary report*

Description

Generate a snippet of HTML code for summary report

Usage

```
su_gfr_snippet(gfr_elems, output_fn, output_dir = ".", overwrite = TRUE)
```

Arguments

<code>gfr_elems</code>	Google Form Summary Elements object
<code>output_fn</code>	Report filename
<code>output_dir</code>	Output directory
<code>overwrite</code>	Overwrite an existing file, Logical

Details

This generates the same HTML content as [su_gfr_report](#), however only the report elements are saved to the HTML file. In other words this function doesn't produce a full HTML report

See Also

[su_gfr_elems](#), [su_gfr_report](#)

su_gfr_taglist	<i>Generate a snippet of HTML code for summary report</i>
----------------	---

Description

Generate a snippet of HTML code for summary report

Usage

```
su_gfr_taglist(gfr_elems)
```

Arguments

gfr_elems	Google Form Summary Elements object
-----------	-------------------------------------

Details

This will return a list of HTML tags for the elements in gfr_elems

See Also

[su_gfr_elems](#)

su_img_build	<i>Generate a list of HTML tags to show stacked PNGs</i>
--------------	--

Description

Generate a list of HTML tags to show stacked PNGs

Usage

```
su_img_build(
  img_fns,
  display_first = TRUE,
  img_class = "stackme",
  center = FALSE,
  border = c("none", "1px solid gray")[1],
  include_css = FALSE,
  base_dir = NULL
)
```

Arguments

img_fns	A vector of image files names
display_first	Display the first image initially
img_class	The class that image tags will be assigned
center	Center the DIV on the document
border	A valid CSS style for the DIV border
include_css	Include the CSS for the images
base_dir	Where to look for the files

Details

This will return a list of HTML tags that when inserted into a R markdown HTML document will display the PNG files as a build. You can use this function within code chunks in R markdown to create a build effect from a set of PNG files. The PNG files will be added to the DIV in the order received. All images should have the same dimensions, and images after the first one should either have transparent backgrounds or contain the earlier elements in them.

To use this function, add the following style definition to your Rmarkdown document. Alternately, you can pass `include_css = TRUE`, but note that this will insert the style definition into your HTML document each time you insert an image build, so you really only need to add `include_css = TRUE` on the first build in the slide deck.

```
```{css echo = FALSE}
img.stackme {
 position:absolute;
 top:0;
 left:0;
}
```
```

Insert the build into your Markdown with something like:

```
```{r echo = FALSE, results = "asis"}
paste0("./images/myslide_0", 1:6, ".png") %>%
 slideutils::su_img_build(display_first = TRUE, center = FALSE) %>%
 htmltools::tagList()
```
```

su_img_maketrans

Make PNG images transparent backgrounds

Description

Make PNG images transparent backgrounds

Usage

```
su_img_maketrans(  
    x,  
    pattern = ".png$",  
    out_dir = NULL,  
    out_fn = "img_%03d.png",  
    overwrite = FALSE  
)
```

Arguments

| | |
|-----------|---|
| x | Vector of image file names or a directory |
| pattern | Pattern to match if x is a directory |
| out_dir | Output directory |
| out_fn | A filename template |
| overwrite | Overwrite existing files |

Details

This will take one or more PNG files (e.g., saved from PowerPoint slides) and create copies where white becomes transparent. This can be useful if you want to use the images as a build.

TIP: if the source of your PNG files are a series of PowerPoint slides (that you convert to PNG with File » Save As...), set the size of your PowerPoint slides to produce the desired output dimensions. When you save PowerPoint slides to raster formats, it saves them as 96 dpi (unless you've tweaked that). So for example if you want to create PNG files that are 1000 x 562 (roughly 16:9 that will display fine on a single slide), set the size of your PowerPoint presentation to 10.4" x 5.9".

x can be a directory that contains image files, or a vector of filenames. If x is a directory, you can use pattern to pass a regular expression.

out_dir is the output directory (optional). out_fn is a file name template that will be passed to sprintf for evaluation. Example: out_fn = \"slide_%02d.png\" will result in output files named slide_01.png, slide_02.png, slide_03.png, etc. If out_fn is NULL, the original file names appended with '_trnsbg.png' will be used for the output files.

See Also

[su_img_build](#)

su_meme

*Insert a meme***Description**

Insert a meme

Usage

```

su_meme(
  meme = "yoda-sage",
  height = c("small", "medium", "large", "default")[2],
  quote_main = NULL,
  data_uri = FALSE,
  img_path = "./images/",
  copy_to = NULL,
  tbl_width_px = 640,
  tbl_border_css = "none",
  tbl_margin_css = "0 auto",
  font_size_css = "100%",
  font_color_css = "black",
  img_boxshadow_css = c("none", "0px 0px 5px 1px #555")[1],
  img_borderradius_css = c("0", "5px")[1],
  debug = FALSE
)

```

Arguments

| | |
|----------------------|---|
| meme | The name of a meme (character) |
| height | The image height (keyword or number) |
| quote_main | The main quote (character) |
| data_uri | Return a URI for the image instead of a file name (logical) |
| img_path | The relative path from the HTML file to the image (ignored if data_uri is TRUE) |
| copy_to | A directory to copy the image to if needed |
| tbl_width_px | Table width in pixels |
| tbl_border_css | CSS for the table border property |
| tbl_margin_css | CSS for the table margin property |
| font_size_css | CSS for the font-size property |
| font_color_css | CSS for the font color property |
| img_boxshadow_css | CSS for the image box-shadow property |
| img_borderradius_css | CSS for the image border-radius property |
| debug | Print debugging messages to the console, Logical |

Details

This will return an HTML object that can be printed in a RMarkdown.

height can be a keyword `small|medium|large`, which will make the meme image appear 160, 240, or 320 pixels tall respectively. Or you can pass an integer. The width will be scaled automatically.

Note that `img_path` is not the path to the image from the active directory, but rather the relative path from the HTML file being rendered to the image. It will be added to the `src` attribute of the `img` tag.

Value

html object

Examples

```
## Not run:
su_meme("yoda-sage", "medium", "Organize your work in RStudio Projects,<br/>and peace with your files you will have")

## End(Not run)
```

su_tips

Generate small HTML files

Description

Generate small HTML files based on content from a spreadsheet

Usage

```
su_tips(
  gsheet_id,
  gsheet_sheet,
  tips_dir,
  url_base,
  templates_dir = system.file("tips_templates", package = "slideutils"),
  bitly = TRUE,
  open_html = TRUE,
  quiet = FALSE
)
```

Arguments

| | |
|---------------------------|--|
| <code>gsheet_id</code> | A Google Spreadsheet ID |
| <code>gsheet_sheet</code> | The name of the sheet in the Google Spreadsheet |
| <code>tips_dir</code> | Directory where to save the generated HTML files |
| <code>url_base</code> | The base of the URL |

| | |
|---------------|---|
| templates_dir | Directory where the HTML template is saved |
| bitly | Create a bitly link, Logical |
| open_html | Open the local HTML file |
| quiet | Suppress messages |
| | <p>This will generate small HTML files that you can use as hints, tips, or solutions for R exercises. In addition to generating the local HTML file, it will also generate the URL, and optionally a bitly URL, that you can embed in your .Rmd or .R file.</p> <p>The way this works is you have a spreadsheet with columns that contain the content of the questions. Currently, only Google Sheets are supported. You can model your spreadsheet after the following:</p> <p>https://docs.google.com/spreadsheets/d/1iEOGHh4hC4KbiD5pn4kBTBMt3zrJgarvSilR98ei00/copy</p> <p>The spreadsheet does not have to be open to anyone with the link. You'll be prompted to for your credentials using the googlesheets4 package.</p> <p>This script goes through the sheet and i) renders (creates HTML file) for all rows where 'needs_rendering' is blank or TRUE ii) msg_html will be injected into a <div> iii) updates the Google Sheet needs_rendering - set to FALSE url - populated bitly - populated</p> |

Value

A vector of HTML files rendered

Index

su_gfr_elems, [2](#), [3](#), [4](#)
su_gfr_report, [3](#), [4](#)
su_gfr_snippet, [3](#)
su_gfr_taglist, [4](#)
su_img_build, [4](#), [6](#)
su_img_maketrans, [6](#)
su_meme, [7](#)
su_tips, [8](#)