

Package: juicedown (via r-universe)

August 31, 2024

Title 'juice' + 'markdown': Convert 'R Markdown' into 'HTML' with
Inline Styles

Version 0.1.1.9000

Description A convenience tool to create 'HTML' with inline styles
using 'juicyjuice' and 'markdown' packages. It is particularly
useful when working on a content management system (CMS) whose
code editor eliminates style and link tags. The main use case
of the package is the learning management system, 'Moodle'.
Additional helper functions for teaching purposes are provided.
Learn more about 'juicedown' at
<<https://kenjisato.github.io/juicedown/>>.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

URL <https://kenjisato.github.io/juicedown/>

Suggests readxl, testthat (>= 3.0.0)

Config/testthat/edition 3

Imports clipr, dplyr, glue, juicyjuice, knitr, markdown, rlang, rvest,
sass, stringr, xfun, xml2

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

RemoteUrl <https://github.com/kenjisato/juicedown>

RemoteRef HEAD

RemoteSha 52c6db2e29cf3361213e6f523a6d23d2f221591b

Contents

anki	2
anki_setup	2
convert	3
convert_html2cms	5

counter	6
includeAudio	6
includeFlickr	7
includeGraphics	8
includeQuestion	8
includeText	9
includeYT	9
juicedown_example	10
tweak_footnote_highlight	11
tweak_moodle_heading	11

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

anki	<i>Hide/Show text</i>
------	-----------------------

Description

This function inserts HTML snippet to help students memorize important concepts. To make this function to work, you need to run `anki_setup()` somewhere in the markdown document.

Usage

```
anki(text)
```

Arguments

`text` character. Text to toggle.

Value

HTML code span.

anki_setup	<i>Hide/Show text</i>
------------	-----------------------

Description

This function inserts JavaScript code for anki texts. Call this function in a code chunk somewhere in your R Markdown file. Use `anki()` function with a inline r code, to make a hidden block.

Usage

```

anki_setup(
  background = "yellow",
  color = "blue",
  border = "slateblue",
  padding = 3,
  margin = 3,
  cursor = "pointer"
)

```

Arguments

background	character. Background color of the blank box.
color	character. Font color of the answer of the blank box.
border	character. Border color of the blank box.
padding	integer. Space between the answer text and the border.
margin	integer. Left and right margin around the border.
cursor	character.

Value

character. CSS/JavaScript code for anki texts.

convert

Converts R Markdown file into HTML fragment with inline styles

Description

This function facilitates writing contents in R Markdown formats for CMS the source editor of which silently removes style tags. When run interactively, the resulting HTML code will be copied to the clipboard.

It basically does three things:

1. it converts (R) Markdown file into a HTML file with `knitr::knit() + markdown::mark()`,
2. embed CSS information in style attribute with `juicyjuice::css_inline()`, and then
3. extracts article/div element that is ready for copying and pasting into CMS's source editor.

Usage

```

convert(
  file = NULL,
  dir = NULL,
  tag = NULL,
  id = NULL,
  clip = TRUE,

```

```

    full_html = NULL,
    remove_script = NULL,
    stylesheet = NULL,
    template = NULL
  )

```

Arguments

file	character. Path to the (R)markdown file.
dir	character. Output directory.
tag	character. Surrounding tag for the HTML block, "article" and "div" are allowed.
id	character. ID attribute for the surrounding tag.
clip	logical. Whether or not copy the result to clipboard. Ignored (set to FALSE) if full_html = TRUE. Default is TRUE.
full_html	logical. Produce the complete HTML or HTML block only?
remove_script	logical. Whether or not remove script tags. Ignored (set to FALSE) if full_html = TRUE.
stylesheet	character. Paths to the CSS files used in markdown::mark()
template	character. Path to the template used in markdown::mark()

Value

Invisibly returns a character vector identical to the result file.

Details

Change default behaviors with `options()`:

Currently, three global defaults are available. You can set these defaults to alter the appearance of the result.

- `juicedown.template`: Defaults to `juicedown::pkg_file("xml", "template.html")`
- `juicedown.article.css`: Defaults to `juicedown::pkg_file("css", "article.scss")`
- `juicedown.div.css`: `juicedown::pkg_file("css", "div.scss")`

Priority order:

You can pass conversion parameters in function argument and YAML metadata (under `juicedown` key) and for some parameters, global options, with priority given in that order. For instance, if the `stylesheet` parameter exists in the function call, it is used. If not and if YAML metadata has `stylesheet` key under `juicedown` key, then that will be used. In the below example, `some.css` used.

```

---
juicedown:
  stylesheet:
    some.css
---

```

If neither the function argument nor missing YAML metadata exist, then the global option (such as `juicedown.article.css`) will be used.

Examples

```
file <- juicedown_example("markdown", "sample.md")
tdir <- tempdir()
convert(file, dir = tdir, clip = FALSE)
unlink(file.path(tdir, "sample.html"))
```

convert_html2cms	<i>Convert a HTML file into a HTML block to be copied and pasted for CMS.</i>
------------------	---

Description

Convert a HTML file into a HTML block to be copied and pasted for CMS.

Usage

```
convert_html2cms(
  in_text,
  full_html = the$full_html %||% FALSE,
  tag = the$tag %||% "body",
  id = the$id,
  remove_script = the$remove_script %||% FALSE
)
```

Arguments

in_text	character. Character vector representing the HTML file to convert.
full_html	logical. If TRUE, produce complete html output, convenient when drafting a document. When this option is enabled, tag and id and remove_script options are silently ignored.
tag	character. Outer-most tag for the resulting HTML snippet.
id	character. id attribute for the outer-most tag.
remove_script	logical. If TRUE, script tags are all stripped out.

Value

character. HTML block.

counter	<i>Counter factory.</i>
---------	-------------------------

Description

Define a counter with `cnt <- counter("type")`. Add cross-referenced label in your document with `cnt$set("ref")`. Refer to "ref" with `cnt$get("ref")`.

Usage

```
counter(label, open = NULL, close = NULL, sep = " ", link = FALSE)
```

Arguments

label	character. Default prefix for the counter.
open	character. Opening bracket such as "(" or "["
close	character. Closing bracket such as ")" or "]"
sep	character. Separator between label and counter.
link	logical. If TRUE, <code>get()</code> method returns a counter surrounded by a tag, linked to the referenced id.

Value

List of functions to set and get counters with label.

Examples

```
cnt <- counter("Quiz")
cnt$set("aaa")
cnt$set()
cnt$set("bbb", quiet = TRUE)
cnt$get("aaa")
cnt$get("bbb")
cnt$get("aaa", type = "bare")
cnt$dump()
```

includeAudio	<i>Include Audio media.</i>
--------------	-----------------------------

Description

Include Audio media.

Usage

```
includeAudio(  
  url,  
  preload = c("metadata", "auto", "none"),  
  alt = "Your browser does not support the audio element."  
)
```

Arguments

url	character. URL of the audio. The media must not be gated.
preload	character. One of metadata, auto and none.
alt	character. Message displayed when the browser does not support the audio tag.

Value

An R object of class "knit_asis" (defined in the [knitr](#) package), the data of which is the HTML audio tag to insert into the document.

includeFlickr	<i>Include Flickr album</i>
---------------	-----------------------------

Description

Include Flickr album

Usage

```
includeFlickr(albumUrl, title = NULL, width = 480, height = 360)
```

Arguments

albumUrl	character. Share URL of a public album
title	character. Default NULL. If given, overwrite the title of the album.
width	integer. Width of the widget in pixel.
height	integer. Height of the widget in pixel.

Value

Flickr embed code.

includeGraphics	<i>Include graphic media.</i>
-----------------	-------------------------------

Description

This function is a wrapper for `knitr::include_graphics()`.

Usage

```
includeGraphics(path, ...)
```

Arguments

path	character. Path to the image.
...	Parameters passed to <code>knitr::include_graphics()</code> , other than path.

Value

The result of `knitr::include_graphics()`

includeQuestion	<i>Include Summary/Details block.</i>
-----------------	---------------------------------------

Description

Include Summary/Details block.

Usage

```
includeQuestion(
  summary,
  detail,
  summary_label = "Quiz",
  detail_label = "Answer",
  ...
)
```

Arguments

summary	character. Text to show, typically, question string.
detail	character. Texts to hide, typically, answer string.
summary_label	character. Label for the open block.
detail_label	character. Label for the hidden block.
...	character. Concatenated with the detail argument.

Value

HTML5 summary/details block is inserted.

includeText	<i>Insert text into a document.</i>
-------------	-------------------------------------

Description

Insert text into a document.

Usage

```
includeText(path, data = parent.frame(), quiet = TRUE)
```

Arguments

path	Path to the child document.
data	Environment or list used to knit the child document.
quiet	Passed to <code>knitr::knit()</code> .

Value

An R object of class "knit_asis" (defined in the `knitr` package), the data of which represents the content of the child document (after `knitr::knit()`)

Examples

```
template <- tempfile()
cat("Hello, `r name`.\\n", file = template)
includeText(template, data = list(name = "Alice"))
file.remove(template)
```

includeYT	<i>Include YouTube video.</i>
-----------	-------------------------------

Description

Include YouTube video.

Usage

```
includeYT(url, .class = "includeYT")
```

Arguments

<code>url</code>	character. URL of the YouTube video.
<code>.class</code>	character. The resulting iframe tag is surrounded by div block. You can specify its class attribute to modify its styling.

Value

An embed code of a YouTube video is inserted.

`juicedown_example` *List example files*

Description

List example files for the package. Show the path when the exact name is given.

Usage

```
juicedown_example(name = NULL, file = NULL)
```

Arguments

<code>name</code>	character. A single string for example name.
<code>file</code>	character. A single string for the main file.

Details

The package sample files are organized under samples directory like so:

```
samples/ |– from-html/ | -- sample.html – include/ |– pic/ ‘– sample.Rmd
```

- `juicedown_example()` is equivalent to `ls samples`,
- `juicedown_example("from-html")` is to `ls samples/from-html`, and
- `juicedown_example("from-html", "sample.html")` shows the full path.

Value

character. A list of example files or a path string to each example file.

Examples

```
juicedown_example()
juicedown_example("from-html")
juicedown_example("from-html", "sample.html")
```

tweak_footnote_highlight
Highlight target link for footnotes

Description

Footnote highlighting is corrupt when CSS is in-lined. Fix this by calling this function in a code chunk somewhere in your R Markdown file.

Usage

```
tweak_footnote_highlight()
```

Value

Script block that makes highlighting possible.

tweak_moodle_heading *Hide H2 heading for Moodle LMS*

Description

You may want the title of the article to be different from the title of the resource. Call this function in a code chunk somewhere in your R Markdown file.

Usage

```
tweak_moodle_heading()
```

Value

Script block that makes H2 head of the content invisible.

Index

anki, 2
anki(), 2
anki_setup, 2
anki_setup(), 2

convert, 3
convert_html2cms, 5
counter, 6

includeAudio, 6
includeFlickr, 7
includeGraphics, 8
includeQuestion, 8
includeText, 9
includeYT, 9

juicedown_example, 10
juicyjuice::css_inline(), 3

knitr, 7, 9
knitr::include_graphics(), 8
knitr::knit(), 3, 9

markdown::mark(), 3

options(), 4

tweak_footnote_highlight, 11
tweak_moodle_heading, 11