Package 'slickR'

February 14, 2020

Title Create Interactive Carousels with the JavaScript
'Slick' Library

Version 0.4.9

Date 2020-02-13

Description Create and customize interactive carousels
using the 'Slick' JavaScript library and the
'htmlwidgets' package. The carousels can contain
plots produced in R, images, 'iframes', videos and
other 'htmlwidgets'. These carousels can be created
directly from the R console, and viewed in the 'RStudio'
internal viewer, in Shiny apps and R Markdown documents.

License MIT + file LICENSE

URL https://github.com/metrumresearchgroup/slickR

BugReports https://github.com/metrumresearchgroup/slickR/issues

Depends R (>= 3.5.0)

Imports htmltools,utils,tools,lifecycle,stats,xml2,base64enc,htmlwidgets

Suggests jsonlite,
knitr,
pdftools,
rmarkdown,
svglite,
leaflet,
testthat,
covr

VignetteBuilder knitr

RdMacros lifecycle

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.0.2

R topics documented:

---

getHelp   *getHelp* **Deprecated**

Description

return rd help file as html

Usage

```
getHelp(fn, pkg)
```

Arguments

fn	character of function name
pkg	name of package that contains fn

Details

This function has been moved to the iframes vignette

Value

character

---

nba_player_logo   *NBA player logo*

Description

NBA player logo URI

Usage

nba_player_logo

Format

A data frame with 150 rows and 4 variables:

team character, Team name
position factor, Position of the Player
name character, Player Name
uri character, Player Headshot URI
player_home character, Player Homepage URI

Details

Logos harvested from http://www.espn.com/nba/depth

See Also

Other data: nba_team_logo

---

nba_team_logo   *NBA team logo*

Description

NBA team logo URI

Usage

nba_team_logo

Format

A data frame with 30 rows and 2 variables:

team character, Team Name
uri character, Team logo URI

Details

Logos harvested from https://i.cdn.turner.com/nba/nba/.element/img/4.0/global/logos/512x512/bg.white/svg/team_short

See Also

Other data: nba_player_logo

---

settings   *Modify components of slick settings*

Description

Customize non-data components of the carousel.

Usage

```
settings(
  accessibility,
  adaptiveHeight,
  autoplay,
  autoplaySpeed,
  arrows,
  asNavFor,
  appendArrows,
  appendDots,
  prevArrow,
  nextArrow,
  centerMode,
  centerPadding,
  cssEase,
  customPaging,
  dots,
  dotsClass,
  draggable,
  fade,
  focusOnSelect,
  easing,
  edgeFriction,
  infinite,
  initialSlide,
  lazyLoad,
  mobileFirst,
  pauseOnFocus,
  pauseOnHover,
  pauseOnDotsHover,
  respondTo,
  responsive,
  rows,
  slide,
  slidesPerRow,
  slidesToShow,
  slidesToScroll,
  speed,
  swipe,
```

```
    swipeToSlide,
    touchMove,
    touchThreshold,
    useCSS,
    useTransform,
    variableWidth,
    vertical,
    verticalSwiping,
    rtl,
    waitForAnimate,
    zIndex,
    ...
)
```

Arguments

`accessibility`	logical, Enables tabbing and arrow key navigation, Default: TRUE
`adaptiveHeight`	logical, Enables adaptive height for single slide horizontal carousels., Default: FALSE
`autoplay`	logical, Enables Autoplay, Default: FALSE
`autoplaySpeed`	numeric, Autoplay Speed in milliseconds, Default: 3000
`arrows`	logical, Prev/Next Arrows, Default: TRUE
`asNavFor`	character, Set the slider to be the navigation of other slider (Class or ID Name), Default: null
`appendArrows`	character, Change where the navigation arrows are attached (Selector, htmlString, Array, Element, jQuery object), Default: $(element)
`appendDots`	character, Change where the navigation dots are attached (Selector, htmlString, Array, Element, jQuery object), Default: $(element)
`prevArrow`	character (html|jQuery selector) | object (DOM node|jQuery object), Allows you to select a node or customize the HTML for the "Previous" arrow., Default: <button type="button" class="slick-prev">Previous</button>
`nextArrow`	character (html|jQuery selector) | object (DOM node|jQuery object), Allows you to select a node or customize the HTML for the "Next" arrow., Default: <button type="button" class="slick-next">Next</button>
`centerMode`	logical, Enables centered view with partial prev/next slides. Use with odd numbered slidesToShow counts., Default: FALSE
`centerPadding`	character, Side padding when in center mode (px or \ Default: '50px'
`cssEase`	character, CSS3 Animation Easing, Default: 'ease'
`customPaging`	function, Custom paging templates. See source for use example., Default: n/a
`dots`	logical, Show dot indicators, Default: FALSE
`dotsClass`	character, Class for slide indicator dots container, Default: 'slick-dots'
`draggable`	logical, Enable mouse dragging, Default: TRUE
`fade`	logical, Enable fade, Default: FALSE
`focusOnSelect`	logical, Enable focus on selected element (click), Default: FALSE
`easing`	character, Add easing for jQuery animate. Use with easing libraries or default easing methods, Default: 'linear'
`edgeFriction`	numeric, Resistance when swiping edges of non-infinite carousels, Default: 0.15

infinite	logical, Infinite loop sliding, Default: TRUE
initialSlide	numeric, Slide to start on, Default: 0
lazyLoad	character, Set lazy loading technique. Accepts 'ondemand' or 'progressive', Default: 'ondemand'
mobileFirst	logical, Responsive settings use mobile first calculation, Default: FALSE
pauseOnFocus	logical, Pause Autoplay On Focus, Default: TRUE
pauseOnHover	logical, Pause Autoplay On Hover, Default: TRUE
pauseOnDotsHover	
	logical, Pause Autoplay when a dot is hovered, Default: FALSE
respondTo	character, Width that responsive object responds to. Can be 'window', 'slider' or 'min' (the smaller of the two), Default: 'window'
responsive	object, Object containing breakpoints and settings objects (see demo). Enables settings sets at given screen width. Set settings to "unslick" instead of an object to disable slick at a given breakpoint., Default: none
rows	numeric, Setting this to more than 1 initializes grid mode. Use slidesPerRow to set how many slides should be in each row., Default: 1
slide	element, Element query to use as slide, Default: ''
slidesPerRow	numeric, With grid mode intialized via the rows option, this sets how many slides are in each grid row. dver, Default: 1
slidesToShow	numeric, # of slides to show, Default: 1
slidesToScroll	numeric, # of slides to scroll, Default: 1
speed	numeric, Slide/Fade animation speed in milliseconds, Default: 300
swipe	logical, Enable swiping, Default: TRUE
swipeToSlide	logical, Allow users to drag or swipe directly to a slide irrespective of slidesToScroll, Default: FALSE
touchMove	logical, Enable slide motion with touch, Default: TRUE
touchThreshold	numeric, To advance slides, the user must swipe a length of (1/touchThreshold) * the width of the slider, Default: 5
useCSS	logical, Enable/Disable CSS Transitions, Default: TRUE
useTransform	logical, Enable/Disable CSS Transforms, Default: TRUE
variableWidth	logical, Variable width slides, Default: FALSE
vertical	logical, Vertical slide mode, Default: FALSE
verticalSwiping	
	logical, Vertical swipe mode, Default: FALSE
rtl	logical, Change the slider's direction to become right-to-left, Default: FALSE
waitForAnimate	logical, Ignores requests to advance the slide while animating, Default: TRUE
zIndex	numeric, Set the zIndex values for slides, useful for IE9 and lower, Default: 1000
...	unused

Details

To find further information on the attributes that can be used please refer to http://kenwheeler.github.io/slick/.

Value

list of class c('setting','slickR')

Examples

```
if(interactive()){

slick <- slickR(obj=nba_team_logo$uri)

slick

# Use settings function to manipulate the slick object

# Add dots

slick + settings(dots = TRUE)

# Autplay elements in the carousel

slick + settings(autoplay = TRUE, autoplaySpeed = 1000)

}
```

---

slickpdf   *slickpdf* **Deprecated**

Description

convert multiple pdfs or a multipage pdf to a slick carousel

Usage

```
slickpdf(obj, obj2 = NULL, synch = TRUE, img_format = "png", ...)
```

Arguments

obj	character, path to pdf(s)
obj2	character, path to pdf(s) to compare to obj, Default: NULL
synch	boolean, if TRUE then obj and obj2 images are synchronized, Default: TRUE
img_format	format of image to convert to, Default: 'png'
...	arguments passed to [slickR](#)

Details

This function has been replaced with the example in the PDF vignette.

Value

slick

---

slickR	*slick.js image carousel htmlwidget*

Description

use slick.js library in R

Usage

```
slickR(
  obj,
  slideId = "baseDiv",
  slideType = "img",
  objLinks = NULL,
  padding = 1,
  width = "95%",
  height = NULL,
  elementId = NULL,
  slickOpts = NULL,
  synchSlides = NULL,
  dotObj = NULL
)
```

Arguments

obj	character, vector of path or url to images
slideId	character, id of slide, Default: 'baseDiv'
slideType	character, type of object to put in slide, Default: 'img'
objLinks	character, links to attach to images in slide, Default: NULL
padding	numeric, percent of width between each image in the carousel for each slider, Default: 1
width	character, width of htmlwidget, Default: '95%'
height	character, height of htmlwidget, Default: NULL
elementId	character, id tag of htmlwidget, Default: NULL
slickOpts	**Deprecated**
synchSlides	**Deprecated**
dotObj	**Deprecated**

Details

slick.js is an image carousel javascript library.

To find all the attributes that can be used please refer to the link.

It is possible to stack slides through the %stack% operator.

It is possible to synchronize slides through the %synch% operator.

To replace the dots with icons use the settings to define the customPaging attribute with the appropriate JS(.) function call.

The slideType accepts the type of html DOM you want to be in the slide, eg img, iframe.

The combination of slideType = 'img-lazy' and settings(lazyLoad = 'ondemand') will inform the htmlwidget that the images are to be loaded lazily. For more information see the slickjs documentation.

Examples

```
if(interactive()){

slick <- slickR(obj=nba_team_logo$uri)

slick

# add dots to the first slick and autoplay at 1 second a slide

slick + settings(dots = TRUE, autoplay = TRUE, autoplaySpeed = 1000)


# working with multiple slicks

sets <- split(
  sample(nba_team_logo$uri, size = 28, replace = FALSE),
  rep(c(1,2,3,4),each=7)
)

slicks <- lapply(sets,FUN = function(x,...){
  slickR(obj = x,...)
},height = 100)

# independent slicks

Reduce(`%stack%`,slicks)

# 1,2 synch stacked on 3,4 synch

 (slicks[[1]] %synch% slicks[[2]]) %stack% (slicks[[3]]%synch%slicks[[4]])

}
```

---

slickR-shiny	*Shiny bindings for slickR*

Description

Output and render functions for using slickR within Shiny applications and interactive Rmd documents.

Usage

```
slickROutput(outputId, width = "100%", height = "400px")

renderSlickR(expr, env = parent.frame(), quoted = FALSE)
```

Arguments

`outputId`	output variable to read from
`width, height`	Must be a valid CSS unit (like `'100%'`, `'400px'`, `'auto'`) or a number, which will be coerced to a string and have `'px'` appended.
`expr`	An expression that generates a slickR
`env`	The environment in which to evaluate `expr`.
`quoted`	Is `expr` a quoted expression (with `quote()`)? This is useful if you want to save an expression in a variable.

---

slick_div	*Create a DOM element for slickR*

Description

Wraps an R object in slickR compatible DOM

Usage

```
slick_div(
  x,
  css = htmltools::css(marginLeft = "auto", marginRight = "auto"),
  type = "img",
  links = NULL
)
```

Arguments

x	object
css	[css](#) object, Default: htmltools::css(marginLeft='auto',marginRight='auto')
type	character, type of DOM, Default: 'img'
links	character, links to use on the input object, Default: NULL

Details

Method converts xml_document, htmlwidget, character, or list of objects to a compatible DOM

Value

list of [tag](#) objects

See Also

Other html: [slick_list](#)()

Examples

```
# image
slick_div(nba_team_logo$uri[1])

# vector of images
slick_div(nba_team_logo$uri[c(1:5)])

# text
slick_div('abc',type = 'p')
```

---

slick_list   *List of DOM elements*

Description

Convert a list of tags into a list that slickR can render

Usage

```
slick_list(...)
```

Arguments

...   Elements to convert

Value

tagList

See Also

tagList, tags

Other html: slick_div()

Examples

```
img1 <- htmltools::img(src = nba_player_logo$uri[1])
img2 <- htmltools::img(src = nba_player_logo$uri[2])

slick_list(img1,img2)
```

%synch%   *Slick pipe operators*

Description

Pipe operators to combine plots by either stacking or synching multiple sliders.

Usage

```
e1 %synch% e2

e1 %stack% e2
```

Arguments

e1   slickR htmlwidget to place on top

e2   slickR htmlwidget to place on bottom

Value

htmlwidget

Examples

```
if(interactive()){

# working with multiple slicks

sets <- split(
  sample(nba_team_logo$uri, size = 28, replace = FALSE),
  rep(c(1,2,3,4),each=7)
)

slicks <- lapply(sets,FUN = function(x,...){
  slickR(obj = x,...)
},height = 100)

# independent slicks

Reduce(`%stack%`,slicks)

# 1,2 synch stacked on 3,4 synch

 (slicks[[1]] %synch% slicks[[2]]) %stack% (slicks[[3]]%synch%slicks[[4]])

}
```

Index