

Package ‘sparkline’

May 9, 2026

Type Package

Title 'jQuery' Sparkline 'htmlwidget'

Version 2.0

Date 2016-11-10

Maintainer Ramnath Vaidyanathan <ramnath.vaidya@gmail.com>

Description Include interactive sparkline charts
<<http://omnipotent.net/jquery.sparkline>> in
all R contexts with the convenience of 'htmlwidgets'.

Imports htmltools, htmlwidgets (>= 0.8)

Suggests formattable, knitr, rmarkdown, shiny

License MIT + file LICENSE

RoxygenNote 5.0.1

VignetteBuilder knitr

NeedsCompilation no

Author Ramnath Vaidyanathan [aut, cre],
Kent Russell [aut, ctb],
Gareth Watts [aut, cph] (jquery.sparkline library in htmlwidgets/lib,
<https://github.com/gwatts/jquery.sparkline>),
jQuery Foundation [cph] (jQuery library),
jQuery contributors [ctb, cph] (jQuery library; authors listed in
inst/htmlwidgets/lib/jquery/jquery-AUTHORS.txt)

Repository CRAN

Date/Publication 2016-11-12 01:06:40

Contents

sparkline	2
sparkline-shiny	2
spk_add_deps	3
spk_chr	4
spk_composite	5

Index	7
--------------	----------

sparkline *Interactive Sparklines*

Description

Create interactive sparklines for inline visualization.

Usage

```
sparkline(values, ..., width = 60, height = 20, elementId = NULL,
  renderSelector = NULL)
```

Arguments

values	vector of values to visualize
...	additional options passed to <code>jquery.sparkline</code> ; see jquery.sparkline docs
height, width	height and width of sparkline htmlwidget specified in any valid CSS size unit.
elementId	string id as a valid CSS element id.
renderSelector	string as CSS selector to render in a DOM element other than the htmlwidget container

sparkline-shiny *Shiny bindings for sparkline*

Description

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

Usage

```
sparklineOutput(outputId, width = "60px", height = "20px")

renderSparkline(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 sparkline
env	The environment in which to evaluate expr.
quoted	Is expr a quoted expression (with <code>quote()</code>)? This is useful if you want to save an expression in a variable.

`spk_add_deps`*Add Sparkline Dependencies to Tag or 'htmlwidget'*

Description

Add Sparkline Dependencies to Tag or 'htmlwidget'

Usage

```
spk_add_deps(tag_htmlwidget = NULL)
```

Arguments

`tag_htmlwidget` shiny.tag or htmlwidget to which you would like to add sparkline dependencies

Value

shiny.tag or htmlwidget

Examples

```
# use spk_add_deps with other htmlwidgets

library(sparkline)
library(formattable)

fw <- as.htmlwidget(
  formattable(
    data.frame(
      id = c("a", "b"),
      sparkline = c(
        spk_chr(runif(10,0,10), type="bar"),
        spk_chr(runif(10,0,5), type="bar")
      ),
      stringsAsFactors = FALSE
    )
  )
)

spk_add_deps(fw)

# use spk_add_deps with htmltools/shiny tags

library(sparkline)
library(htmltools)

div <- tags$div(
  spk_chr(1:10, type="bar")
)
```

```
spk_add_deps(div)
```

```
spk_chr Character Version of Sparklines
```

Description

Create a character version of interactive sparklines for use with other 'htmlwidgets' or tags.

Usage

```
spk_chr(values, ..., width = 60, height = 20, elementId = NULL,
        renderSelector = NULL)
```

Arguments

values	vector of values to visualize
...	additional options passed to <code>jquery.sparkline</code> ; see jquery.sparkline docs
width	height and width of sparkline htmlwidget specified in any valid CSS size unit.
height	height and width of sparkline htmlwidget specified in any valid CSS size unit.
elementId	string id as a valid CSS element id.
renderSelector	string as CSS selector to render in a DOM element other than the htmlwidget container

Examples

```
## Not run:
#spk_chr works well with dplyr summarise

library(dplyr)
library(sparkline)
library(formattable)

mtcars %>%
  group_by(cyl) %>%
  summarise(
    hp = spk_chr(
      hp, type="box",
      chartRangeMin=0, chartRangeMax=max(mtcars$hp)
    ),
    mpg = spk_chr(
      mpg, type="box",
      chartRangeMin=0, chartRangeMax=max(mtcars$mpg)
    )
  ) %>%
  formattable() %>%
  formattable::as.htmlwidget() %>%
```

```
    spk_add_deps()
## End(Not run)
```

spk_composite	<i>Add a Composite to an Existing Sparkline</i>
---------------	---

Description

Add a Composite to an Existing Sparkline

Usage

```
spk_composite(sparkline = NULL, sparklineToAdd = NULL, ...)
```

Arguments

`sparkline` [sparkline](#) to which we would like to add a composite.
`sparklineToAdd` [sparkline](#) to add to another [sparkline](#)
`...` extra arguments to modify `sparklineToAdd` or define a [sparkline](#).

Value

[sparkline](#) object

Examples

```
library(sparkline)

s11 <- sparkline(
  c(5,4,5,-2,0,3),
  type='bar',
  barColor="#aaf",
  chartRangeMin=-5,
  chartRangeMax=10,
  # set an id that will make it easier to refer
  # in the next sparkline
  elementId="sparkline-for-composite"
)

s12 <- sparkline(
  c(4,1,5,7,9,9,8,7,6,6,4,7,8,4,3,2,2,5,6,7),
  type="line",
  fillColor = FALSE,
  lineColor = 'red',
  chartRangeMin = -5,
  chartRangeMax = 10
)
```

```
# add sparkline as a composite
spk_composite(s11, s12)

# add values and options as a composite
spk_composite(
  s11,
  values=c(4,1,5,7,9,9,8,7,6,6,4,7,8,4,3,2,2,5,6,7),
  options = list(
    type="line",
    fillColor = FALSE,
    lineColor = 'red',
    chartRangeMin = -5,
    chartRangeMax = 10
  )
)

# add combination of sparkline and options as a composite
spk_composite(
  s11,
  s12,
  options = list(
    type="box"
  )
)
```

Index

`renderSparkline` (`sparkline-shiny`), 2

`sparkline`, 2, 5

`sparkline-shiny`, 2

`sparklineOutput` (`sparkline-shiny`), 2

`spk_add_deps`, 3

`spk_chr`, 4

`spk_composite`, 5