

# Package ‘fusionchartsR’

May 8, 2026

**Type** Package

**Version** 1.2.0

**Title** Embedding FusionCharts in R

**Description** FusionCharts provides awesome and minimalist functions to make beautiful interactive charts <<https://www.fusioncharts.com/>>.

**License** MIT + file LICENSE

**URL** <https://alexym1.github.io/fusionchartsR/>,  
<https://github.com/alexym1/fusionchartsR>

**BugReports** <https://github.com/alexym1/fusionchartsR/issues>

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**Imports** ggplot2, htmlwidgets, jsonlite, magrittr, scales, shiny

**Suggests** rmarkdown, knitr

**NeedsCompilation** no

**Author** Alex Yahiaoui Martinez [aut, cre] (ORCID:  
<<https://orcid.org/0000-0002-5315-675X>>)

**Maintainer** Alex Yahiaoui Martinez <yahiaoui-martinez.alex@outlook.com>

**Repository** CRAN

**Date/Publication** 2026-01-12 18:50:15 UTC

## Contents

fusionAnchors . . . . .	2
fusionAxis . . . . .	3
fusionBackground . . . . .	4
fusionCanvas . . . . .	5
fusionCaption . . . . .	7
fusionCustomAxis . . . . .	8
fusionCustomBoxplot . . . . .	10
fusionCustomLegend . . . . .	12

fusionDiv	14
fusionGrid	16
fusionGrid-shiny	17
fusionLegend	18
fusionLogo	19
fusionMultiPlot	20
fusionMultiPlot-shiny	21
fusionPalette	21
fusionPlot	23
fusionPlotOutput	24
fusionSubcaption	25
fusionTheme	26
fusionTooltip	26
fusionTrendline	27
ggfusionPlot	29
runDemo	30

## Index 31

---

fusionAnchors	<i>Adding FusionCharts anchors</i>
---------------	------------------------------------

---

### Description

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/anchors-and-lines>

### Usage

```
fusionAnchors(
  fusionPlot,
  drawAnchors = TRUE,
  showvalues = FALSE,
  anchorSides = "0",
  anchorRadius = "3",
  anchorAlpha = "100",
  anchorBorderThickness = "1",
  anchorBorderColor = "#5a5a5a",
  anchorBgColor = "#ffffff",
  anchorBgAlpha = "100",
  anchorImageAlpha = "100",
  anchorImageScale = "150"
)
```

### Arguments

fusionPlot	fusionPlot object got by fusionPlot()
drawAnchors	Show the anchors
showvalues	Show the values of the anchors

anchorSides	Specify the number of sides to define the shape of the anchor
anchorRadius	Set the radius of the anchor
anchorAlpha	Set the transparency of the anchor
anchorBorderThickness	Set the thickness of the anchor border
anchorBorderColor	Set the hex code for anchor border color
anchorBgColor	Set the hex code for anchor background color
anchorBgAlpha	Set the transparency of the anchor background
anchorImageAlpha	Set the transparency of the image
anchorImageScale	Set the scale of the image

### Examples

```
library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
df %>%
  fusionPlot(x = "label", y = "value", type = "line") %>%
  fusionAnchors(anchorRadius = "6", anchorBorderThickness = "2") %>%
  fusionTheme(theme = "fusion")
```

---

fusionAxis

*Adding FusionCharts axis*

---

### Description

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/axes>

### Usage

```
fusionAxis(
  fusionPlot,
  xAxisName = "Change X axis",
  yAxisName = "Change Y axis",
  AxisNameFont = "Arial",
  AxisNameFontSize = "12",
  AxisNameFontColor = "#999999",
  AxisNameFontBold = TRUE,
  AxisNameFontItalic = FALSE
)
```

**Arguments**

**fusionPlot** fusionPlot object got by fusionPlot()  
**xAxisName** Specify the title of the X-axis of the chart  
**yAxisName** Specify the title of the Y-axis of the chart  
**AxisNameFont** Set the font family of axis  
**AxisNameFontSize**  
                   Set the font size (between 0 and 72) of axis  
**AxisNameFontColor**  
                   Set the font color of axis in hex code  
**AxisNameFontBold**  
                   Set the font style to bold  
**AxisNameFontItalic**  
                   Set the font style to italic

**Examples**

```

library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
df %>%
  fusionPlot(x = "label", y = "value", type = "column3d") %>%
  fusionAxis(xAxisName = "Countries", yAxisName = "Numbers", AxisNameFontSize = "20") %>%
  fusionCustomAxis(xAxisPosition = "top", yAxisPosition = "right") %>%
  fusionTheme(theme = "gammel")

```

---

fusionBackground	<i>Adding FusionCharts borders &amp; background</i>
------------------	---

---

**Description**

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/border-and-background>

**Usage**

```

fusionBackground(
  fusionPlot,
  showBorder = FALSE,
  borderColor = "#666666",
  borderThickness = "4",
  borderAlpha = "80",
  bgColorStart = "#ffffff",
  bgColorEnd = NULL,
  bgAlphaStart = "50",
  bgAlphaEnd = NULL,
  bgratioStart = "60",
  bgratioEnd = "40",
  bgAngle = "180"
)

```

**Arguments**

fusionPlot	fusionPlot object got by fusionPlot()
showBorder	Show the chart border
borderColor	Specify the color of the border
borderThickness	Set the thickness of the border
borderAlpha	Set the transparency of the border
bgColorStart, bgColorEnd	Set the hex codes of the starting and ending gradient colors
bgAlphaStart, bgAlphaEnd	Set the transparency of the starting ending gradient colors
bgratioStart, bgratioEnd	Set the radius of gradient colors
bgAngle	Set the angle in degrees of gradient colors

**Examples**

```
library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
df %>%
  fusionPlot(x = "label", y = "value", type = "column3d") %>%
  fusionBackground(showBorder = TRUE, bgColorStart = "#DDDDDD") %>%
  fusionTheme(theme = "fusion")
```

fusionCanvas

*Adding FusionCharts canvas***Description**

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/canvas>

**Usage**

```
fusionCanvas(
  fusionPlot,
  showCanvasBg = FALSE,
  canvasbgColorFirst = "#5a5a5a",
  canvasbgColorSecond = NULL,
  canvasBgDepth = "0",
  canvasbgAlpha = "100",
  canvasBgRatioStart = "40",
  canvasBgRatioEnd = "60",
  canvasBgAngle = "0",
  showCanvasBorder = FALSE,
```

```

    canvasBorderColor = "#666666",
    canvasBorderAlpha = "80",
    canvasBorderThickness = "1",
    showCanvasBase = FALSE,
    canvasBaseDepth = "5",
    canvasBaseColor = "#aaaaaa"
  )

```

### Arguments

```

fusionPlot      fusionPlot object got by fusionPlot()
showCanvasBg    Show the canvas background
canvasbgColorFirst
                Specify the hex code of the first canvas background color
canvasbgColorSecond
                Specify the hex code of the second canvas background color
canvasBgDepth   Set the depth of the canvas background
canvasBgAlpha   Set the transparency of the background color
canvasBgRatioStart
                Set the first value of the canvas background ratio (in percentage)
canvasBgRatioEnd
                Set the second value of the canvas background ratio (in percentage)
canvasBgAngle   Specify canvas background angle (in degrees)
showCanvasBorder
                Show the canvas border
canvasBorderColor
                Set the border color
canvasBorderAlpha
                Set the transparency of the border
canvasBorderThickness
                Set the thickness of the border
showCanvasBase  Show the canvas base
canvasBaseDepth
                Set the height of the canvas base
canvasBaseColor
                Specify the hex code of the base color

```

### Examples

```

library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
df %>%
  fusionPlot(x = "label", y = "value", type = "column2d") %>%
  fusionCanvas(showCanvasBorder = TRUE, canvasBorderThickness = "4", canvasBorderAlpha = "80") %>%
  fusionTheme(theme = "fusion")

```

---

fusionCaption	<i>Adding FusionCharts caption</i>
---------------	------------------------------------

---

## Description

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/caption-and-sub-caption>

## Usage

```
fusionCaption(  
  fusionPlot,  
  caption = "Add a caption here",  
  captionFont = "Arial",  
  captionFontSize = "18",  
  captionFontColor = "#5A5A5A",  
  captionFontBold = TRUE,  
  captionOnTop = TRUE,  
  captionAlignment = c("center", "left", "right")  
)
```

## Arguments

fusionPlot	fusionPlot object got by fusionPlot()
caption	Specify the caption of the chart
captionFont	Set the caption font family
captionFontSize	Set the caption font size (between 0 and 72)
captionFontColor	Set the caption font color
captionFontBold	Enable caption font to bold
captionOnTop	Display the caption at the top of the chart
captionAlignment	Specify the horizontal alignment of the caption

## Examples

```
library(fusionchartsR)  
  
mtcars %>%  
  fusionPlot(  
    x = "cyl",  
    y = "mpg",  
    type = "boxandwhisker2d"  
  ) %>%  
  fusionCaption(caption = "Caption on the left", captionAlignment = "left") %>%
```

```
fusionSubcaption(subcaption = "subcaption too") %>%
fusionPalette(palettecolors = c("#5D62B5", "#979AD0")) %>%
fusionTheme(theme = "fusion")
```

---

fusionCustomAxis	<i>Customing FusionCharts axis</i>
------------------	------------------------------------

---

## Description

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/axes>

## Usage

```
fusionCustomAxis(
  fusionPlot,
  showlabels = TRUE,
  xAxisPosition = c("bottom", "top", "left", "right"),
  yAxisPosition = c("left", "right", "top", "bottom"),
  yAxisMinValue = NULL,
  yAxisMaxValue = NULL,
  AxisNameBorderColor = NULL,
  AxisNameBorderAlpha = "0",
  AxisNameBorderPadding = "6",
  AxisNameBorderRadius = "3",
  AxisNameBorderThickness = "2",
  AxisNameBorderDashed = FALSE,
  AxisNameBorderDashLen = "4",
  AxisNameBorderDashGap = "2",
  AxisNameBgColor = NULL,
  AxisNameBgAlpha = "0",
  AxisNameFontAlpha = "100",
  AxisValueFont = "Arial",
  AxisValueFontSize = "1px",
  AxisValueFontColor = NULL,
  AxisValueFontBold = FALSE,
  AxisValueFontItalic = FALSE,
  AxisValueAlpha = "100",
  AxisValueBgColor = NULL,
  AxisValueBgAlpha = "50",
  AxisValueBorderColor = "#ffffff",
  AxisValueBorderAlpha = "0",
  AxisValueBorderPadding = "5",
  AxisValueBorderRadius = "2",
  AxisValueBorderThickness = "3",
  AxisValueBorderDashed = FALSE,
  AxisValueBorderDashLen = "2",
```

```

    AxisValueBorderDashGap = "2"
)

```

### Arguments

fusionPlot	fusionPlot object got by fusionPlot()
showlabels	Display the data labels
xAxisPosition	change the position of the x-axis
yAxisPosition	change the position of the y-axis
yAxisMinValue	Set the lower limit of the x-axis
yAxisMaxValue	Set the upper limit of the y-axis
AxisNameBorderColor	Set the border color of the name of the axis
AxisNameBorderAlpha	Set the transparency of the border around the name of axis
AxisNameBorderPadding	Set the padding of the border around the name of the axis
AxisNameBorderRadius	Set the radius of the border around the name of the axis
AxisNameBorderThickness	Set the thickness of the border around the name of the axis
AxisNameBorderDashed	Make the border around the name of the axis dashed
AxisNameBorderDashLen	Set the length of each dash in the dashed border around the name of the axis
AxisNameBorderDashGap	Set the gap between two consecutive dashes in the dashed border around the name of the axis
AxisNameBgColor	Set the background color of the name of the axis
AxisNameBgAlpha	Set the transparency of the background of the name of the axis
AxisNameFontAlpha	Set the transparency of the name of the axis
AxisValueFont	Set the font of the axis values
AxisValueFontSize	Set the font size (between 0 to 72) of the axis values
AxisValueFontColor	Set the font color of the axis
AxisValueFontBold	Set the font of the axis values to bold
AxisValueFontItalic	Set the font for the axis values to italics
AxisValueAlpha	Set the degree of transparency of the axis values

**AxisValueBgColor**  
 Set the background color of the axis values

**AxisValueBgAlpha**  
 Set the background color transparency of the axis values

**AxisValueBorderColor**  
 Set the border color of the axis values

**AxisValueBorderAlpha**  
 Set the transparency of the border of the axis values

**AxisValueBorderPadding**  
 Set the padding of the axis values border

**AxisValueBorderRadius**  
 Set the border radius of the axis values

**AxisValueBorderThickness**  
 Set the border thickness of the axis values

**AxisValueBorderDashed**  
 Make the axis values border dashed

**AxisValueBorderDashLen**  
 Set the length of each dash for the dashed borders around axis values

**AxisValueBorderDashGap**  
 Set the gap between two consecutive dashes for the dashed borders around the axis values

### Examples

```

library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
df %>%
  fusionPlot(x = "label", y = "value", type = "column3d") %>%
  fusionAxis(xAxisName = "Countries", yAxisName = "Numbers", AxisNameFontSize = "20") %>%
  fusionCustomAxis(xAxisPosition = "top", yAxisPosition = "right") %>%
  fusionTheme(theme = "gammel")

```

---

fusionCustomBoxplot    *Customing "boxandwhisker2d" chart*

---

### Description

Customing "boxandwhisker2d" chart

### Usage

```

fusionCustomBoxplot(
  fusionPlot,
  showmean = TRUE,
  drawmeanconnector = FALSE,
  lowerboxcolor = "#29C3BE",

```

```

upperboxcolor = "#5D62B5",
mediancolor = "#FFFFFF",
meaniconshape = "polygon",
meaniconsides = "2",
meaniconradius = "2",
showalloutliers = TRUE,
outliericonsides = "20",
outliericonalpha = "40",
outliericonshape = "triangle",
outliericonsradius = "4"
)

```

### Arguments

fusionPlot	fusionPlot object got by fusionPlot()
showmean	Show means
drawmeanconnector	Connect all means
lowerboxcolor	Set the color of the lower box
upperboxcolor	Set the color of the upper box
mediancolor	Set the color of the median line
meaniconshape	Set the shape of the mean icon
meaniconsides	Set the sides of the mean icon
meaniconradius	Set the radius of the mean icon
showalloutliers	Show outliers
outliericonsides	Set the sides of the outliers
outliericonalpha	Set the background color transparency of the outliers
outliericonshape	Set the shape of the outliers
outliericonsradius	Set the radius of the outliers

### Examples

```

library(fusionchartsR)
mtcars %>%
  fusionPlot(x = "cyl", y = "mpg", type = "boxandwhisker2d") %>%
  fusionCustomBoxplot(drawmeanconnector = TRUE)

```

---

fusionCustomLegend      *Customizing FusionCharts legend*

---

### Description

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/legend>

### Usage

```
fusionCustomLegend(
    fusionPlot,
    plotHighlightEffect = FALSE,
    plotHighlightEffectColor = "#7f7f7f",
    plotHighlightEffectAlpha = "60",
    drawCustomLegendIcon = TRUE,
    legendIconBgColor = NULL,
    legendIconAlpha = "100",
    legendIconBgAlpha = "100",
    legendIconBorderColor = "#123456",
    legendIconBorderThickness = "0",
    legendIconSides = "1",
    legendIconStartAngle = "45",
    legendScrollBgColor = "#5A5A5A",
    legendBgColor = "#CCCCCC",
    legendBgAlpha = "0",
    legendBorderColor = "#666666",
    legendBorderThickness = "0",
    legendBorderAlpha = "40",
    legendCaptionAlignment = c("center", "left", "right"),
    legendShadow = FALSE,
    legendItemFontBold = FALSE,
    legendItemFont = "Arial",
    legendItemFontSize = "14",
    legendItemFontColor = "#5A5A5A",
    legendItemHover = FALSE,
    legendItemHoverFontColor = "#cccccc"
)
```

### Arguments

fusionPlot	fusionPlot object got by fusionPlot()
plotHighlightEffect	Enable highlighting of corresponding data series after hover over a legend text
plotHighlightEffectColor	Specify the color
plotHighlightEffectAlpha	Specify the opacity

<code>drawCustomLegendIcon</code>	Enable drawing of a custom legend icon
<code>legendIconBgColor</code>	Specify the hex color code for the background of the legend icon
<code>legendIconAlpha</code>	Set the legend icon transparency (0 to 100)
<code>legendIconBgAlpha</code>	Set the legend icon background transparency
<code>legendIconBorderColor</code>	Specify the hex color code for the border of the legend icon
<code>legendIconBorderThickness</code>	Set the thickness of the legend icon border
<code>legendIconSides</code>	Set the number of sides for the legend icon
<code>legendIconStartAngle</code>	Set the starting angle for drawing the legend icon
<code>legendScrollBgColor</code>	Specify the background color of the scroll bar
<code>legendBgColor</code>	Specify the background color for the legend
<code>legendBgAlpha</code>	Specify the background transparency for the legend
<code>legendBorderColor</code>	Specify the border color for the legend
<code>legendBorderThickness</code>	Specify the border thickness for the legend
<code>legendBorderAlpha</code>	Specify the border transparency for the legend
<code>legendCaptionAlignment</code>	Specify the horizontal alignment of the legend caption
<code>legendShadow</code>	Enable the legend shadow
<code>legendItemFontBold</code>	Display legend keys in bold
<code>legendItemFont</code>	Specify the legend item font
<code>legendItemFontSize</code>	Specify the legend item font size (0 to 72)
<code>legendItemFontColor</code>	Specify the legend item font color
<code>legendItemHover</code>	Enable hover effect to legend item
<code>legendItemHoverFontColor</code>	Specify the legend item font color on hover

**Examples**

```
library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
df %>%
  fusionPlot(x = "label", y = "value", type = "doughnut2d") %>%
  fusionCustomLegend(plotHighlightEffect = TRUE) %>%
  fusionTheme(theme = "fusion")
```

---

fusionDiv

*Adding FusionCharts Div & Grid*


---

**Description**

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/div-lines-and-grids>  
& <https://www.fusioncharts.com/dev/chart-guide/chart-configurations/vertical-div-lines>

**Usage**

```
fusionDiv(
  fusionPlot,
  adjustDiv = FALSE,
  numDivLines = "5",
  divLineColor = "#5a5a5a",
  divLineAlpha = "10",
  divLineDashed = FALSE,
  divLineDashLen = "5",
  divLineDashGap = "6",
  numVDivLines = "5",
  vDivLineColor = "#F2F2F2",
  vDivLineThickness = "1",
  vDivLineAlpha = "100",
  vDivLineDashed = FALSE,
  vDivLineDashLen = "5",
  vDivLineDashGap = "3",
  showAlternateHGridColor = FALSE,
  alternateHGridColor = "#5a5a5a",
  alternateHGridAlpha = "1",
  showAlternateVGridColor = FALSE,
  alternateVGridColor = "#5a5a5a",
  alternateVGridAlpha = "3"
)
```

**Arguments**

fusionPlot	fusionPlot object got by fusionPlot()
adjustDiv	Enable the automatic adjustment of horizontal lines

numDivLines	Set the number of hozitontal lines
divLineColor	Specify the hex code for the color of the hozitontal lines
divLineAlpha	Set the transparency of the hozitontal lines
divLineDashed	Display the hozitontal lines as dashed
divLineDashLen	Set the length of each dashed hozitontal lines
divLineDashGap	Set the gap between the dashed hozitontal lines
numVDivLines	Specify the number of vertical lines
vDivLineColor	Set the color of the vertical lines
vDivLineThickness	Set the thickness of the vertical lines
vDivLineAlpha	Set the transparency of the vertical lines
vDivLineDashed	Display the vertical lines as dashed
vDivLineDashLen	Set the length of each dashed vertical lines
vDivLineDashGap	Set the gap between the dashed vertical lines
showAlternateHGridColor	Display the horizontal grid bands
alternateHGridColor	Specify the hex code for the color of the horizontal grid
alternateHGridAlpha	Set the transparency of the horizontal grid
showAlternateVGridColor	Display the vertical grid bands
alternateVGridColor	Specify the hex code for the color of the vertical grid
alternateVGridAlpha	Set the transparency of the vertical grid

### Examples

```
library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
df %>%
  fusionPlot(x = "label", y = "value", type = "column2d") %>%
  fusionDiv(divLineColor = "#6699cc", divLineAlpha = "60", divLineDashed = TRUE) %>%
  fusionTheme(theme = "fusion")
```

---

`fusionGrid`*Create interactive data tables with FusionGrid*

---

### Description

Create interactive, feature-rich data tables using FusionGrid. FusionGrid provides sorting, filtering, pagination, and many other features for displaying tabular data.

### Usage

```
fusionGrid(  
  data,  
  schema = NULL,  
  config = NULL,  
  dataTableOptions = list(enableIndex = FALSE),  
  width = NULL,  
  height = "500px",  
  elementId = NULL  
)
```

### Arguments

<code>data</code>	A data frame to display in the grid
<code>schema</code>	Optional list defining column schema. Each element should be a list with 'name' and optionally 'type' (e.g., 'number', 'string', 'date')
<code>config</code>	Optional list of configuration options for the grid (layout, rowOptions, etc.)
<code>dataTableOptions</code>	Optional list of options for DataTable creation (default: list(enableIndex = FALSE))
<code>width</code>	Width of the grid (default: NULL for auto-sizing)
<code>height</code>	Height of the grid (default: "500px")
<code>elementId</code>	Optional element ID for the widget

### Value

An htmlwidget object

### Examples

```
## Not run:  
# Basic usage  
fusionGrid(mtcars)  
  
# With schema definition  
schema <- list(  
  list(name = "mpg", type = "number"),  
  list(name = "cyl", type = "number"),
```

```

    list(name = "disp", type = "number"),
    list(name = "hp", type = "number")
  )
fusionGrid(mtcars[, 1:4], schema = schema)

# With configuration
# Some example may be found at:
# https://www.fusioncharts.com/dev/fusiongrid/tutorials-fusiongrid/column-filter
config <- list(
  layout = list(density = "compact", autoHeight = TRUE),
  rowOptions = list(
    style = list("background-color" = "oldlace"),
    hover = list(
      enable = TRUE,
      style = list("background-color" = "white")
    )
  )
)
fusionGrid(iris, config = config)

## End(Not run)

```

---

fusionGrid-shiny

*Shiny bindings for fusionGrid*


---

## Description

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

## Usage

```
fusionGridOutput(outputId, width = "100%", height = "500px")
```

```
renderFusionGrid(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 fusionGrid
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.

fusionLegend

*Adding FusionCharts legend***Description**

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/legend>

**Usage**

```
fusionLegend(
  fusionPlot,
  showLegend = TRUE,
  interactiveLegend = TRUE,
  legendPosition = c("bottom", "left", "right"),
  legendAllowDrag = FALSE,
  legendIconScale = "1",
  reverseLegend = FALSE,
  legendCaption = "",
  legendCaptionBold = TRUE,
  legendCaptionFont = "Arial",
  legendCaptionFontSize = "14",
  legendCaptionFontColor = "#333333"
)
```

**Arguments**

fusionPlot	fusionPlot object got by fusionPlot()
showLegend	Show the legend
interactiveLegend	Enable interactive legend
legendPosition	Specify the position of the legend
legendAllowDrag	Make the legend draggable
legendIconScale	Specify the legend icon size (values from "1" to "5")
reverseLegend	Reverse the order of datasets
legendCaption	Specify the legend caption value
legendCaptionBold	Set the legend caption font style to bold
legendCaptionFont	Specify the legend caption font
legendCaptionFontSize	Specify the legend caption font size
legendCaptionFontColor	Specify the hex color code for the caption font legend

**Examples**

```
library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
df %>%
  fusionPlot(x = "label", y = "value", type = "doughnut2d") %>%
  fusionLegend(legendCaption = "LegendCaption", legendCaptionFontSize = "24") %>%
  fusionTheme(theme = "fusion")
```

fusionLogo

*Adding FusionCharts logo***Description**

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/loading-external-logo>

**Usage**

```
fusionLogo(
  fusionPlot,
  logoURL = "NULL",
  logoAlpha = "40",
  logoScale = "80",
  logoPosition = c("TL", "TR", "BL", "BR", "CC"),
  logoLink = NULL
)
```

**Arguments**

fusionPlot	fusionPlot object got by fusionPlot()
logoURL	Specify the URL of the external logo
logoAlpha	Set the transparency of the external logo
logoScale	Set the scale of the external logo (0 to 300)
logoPosition	Specify the position of the external logo
logoLink	Add an external link to the external logo

**Examples**

```
library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
logoURL <- "https://static.fusioncharts.com/sampledData/images/Logo-HM-72x72.png"
df %>%
  fusionPlot(x = "label", y = "value", type = "doughnut2d") %>%
  fusionLogo(logoURL = logoURL) %>%
  fusionTheme(theme = "fusion")
```

---

fusionMultiPlot      *Create new multiple charts*

---

## Description

Main function to make interactive charts. Check all charts at <https://www.fusioncharts.com/charts>

## Usage

```
fusionMultiPlot(data, x, y, col, type = "msstepline", numberSuffix = NULL)
```

## Arguments

data	Default dataset to use
x, y	character name of variable
col	define seriesname variable
type	type of chart
numberSuffix	Specify the suffix for all the Y-axis values on the chart

## Examples

```
library(fusionchartsR)

# Multiple charts
new.data <- data.frame(
  label = rep(x = c(2012:2016), times = 2),
  seriesname = c(rep("iOS App Store", 5), rep("Google Play Store", 5)),
  values = c(1:10)
)

new.data %>%
  fusionMultiPlot(
    x = "label",
    y = "values",
    col = "seriesname",
    type = "mscolumn2d",
  ) %>%
  fusionTheme(theme = "fusion")
```

---

fusionMultiPlot-shiny *Shiny bindings for fusionMultiPlot*

---

### Description

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

### Usage

```
fusionMultiPlotOutput(outputId, width = "100%", height = "400px")
renderFusionMultiPlot(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 fusionMultiPlot
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.

---

fusionPalette *Adding FusionCharts palette*

---

### Description

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/data-plot>

### Usage

```
fusionPalette(
  fusionPlot,
  palettecolors = NULL,
  usePlotGradientColor = FALSE,
  plotGradientColor = "#003366",
  plotFillAngle = "0",
  plotFillRatioStart = "90",
  plotFillRatioEnd = "100",
  plotFillAlpha = "100",
  showPlotBorder = FALSE,
  drawFullAreaBorder = FALSE,
```

```

inheritPlotBorderColor = FALSE,
plotBorderDashed = FALSE,
plotBorderDashLen = "4",
plotBorderDashGap = "4",
plotBorderThickness = "1",
plotBorderColor = "#666666",
useRoundEdges = FALSE,
plotHoverEffect = FALSE,
plotFillHoverColor = "#5D62B5",
plotFillHoverAlpha = "100",
plotBorderHoverColor = "#000000",
plotBorderHoverAlpha = "100",
plotBorderHoverThickness = "1",
plotBorderHoverDashed = TRUE,
plotBorderHoverDashLen = "6",
plotBorderHoverDashGap = "2"
)

```

### Arguments

**fusionPlot** fusionPlot object got by fusionPlot()

**palettecolors** Specify your custom palette for data plots

**usePlotGradientColor**  
Use the gradient effect

**plotGradientColor**  
Specify the hex code of the gradient color

**plotFillAngle** Set the fill angle for the gradient (0 to 360)

**plotFillRatioStart**  
Specify the start of the gradient effect

**plotFillRatioEnd**  
Specify the end of the gradient effect

**plotFillAlpha** Set the transparency of the gradient fill

**showPlotBorder** Show the plot border

**drawFullAreaBorder**  
To set the top border of the area chart (only works if showPlotBorder = TRUE)

**inheritPlotBorderColor**  
Enable the plot border to inherit the color of an area plot

**plotBorderDashed**  
Make the border dashed

**plotBorderDashLen**  
Set the length of each dash in plot-border (in pixels)

**plotBorderDashGap**  
Set the gap between two consecutive dashes in plot border (in pixels)

**plotBorderThickness**  
Set the thickness of the plot border

plotBorderColor	Set the color of the plot border
useRoundEdges	Enable rounded edges (2D Column or Bar charts only)
plotHoverEffect	Enable hover effects for the data plots
plotFillHoverColor	Set the hover color for data plots in hex code format
plotFillHoverAlpha	Set the transparency for hover color for data plots
plotBorderHoverColor	Set the hover border color
plotBorderHoverAlpha	Set the transparency of hover border for data plots
plotBorderHoverThickness	Set the hover border thickness (in pixels)
plotBorderHoverDashed	Make dashed borders on hover
plotBorderHoverDashLen	Set the length of each dash for all data plots on hover
plotBorderHoverDashGap	Set the gap between two consecutive dashes for all data plots on hover(in pixels)

### Examples

```
library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))

df %>%
  fusionPlot(x = "label", y = "value", type = "pie2d") %>%
  fusionPalette(palettecolors = c("5d62b5", "29c3be", "f2726f")) %>%
  fusionTheme(theme = "gammel")
```

---

fusionPlot

*Create new charts*

---

### Description

Main function to make interactive charts. Check all charts at <https://www.fusioncharts.com/charts>

### Usage

```
fusionPlot(data, x, y, type = "column2d", numberSuffix = NULL)
```

**Arguments**

data	Default dataset to use
x, y	character name of variable
type	type of chart
numberSuffix	Specify the suffix for all the Y-axis values on the chart

**Details**

A 2x2 confusion matrix can be displayed using `type = "confusionMatrix"`.

**Examples**

```
library(fusionchartsR)

# Single
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
df %>%
  fusionPlot(x = "label", y = "value", type = "pie2d") %>%
  fusionTheme(theme = "fusion")
```

---

fusionPlotOutput

*Shiny bindings for fusionPlot*


---

**Description**

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

**Usage**

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

renderfusionPlot(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 fusionPlot
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.

---

fusionSubcaption	<i>Adding FusionCharts subcaption</i>
------------------	---------------------------------------

---

## Description

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/caption-and-sub-caption>

## Usage

```
fusionSubcaption(  
  fusionPlot,  
  subcaption = "Add a subCaption here",  
  subcaptionFont = "Arial",  
  subcaptionFontSize = "14",  
  subcaptionFontColor = "#999999",  
  subcaptionFontBold = FALSE  
)
```

## Arguments

fusionPlot	fusionPlot object got by fusionPlot()
subcaption	Specify the subcaption of the chart
subcaptionFont	Set the subcaption font family
subcaptionFontSize	Set the subcaption font size (between 0 and 72)
subcaptionFontColor	Set the subcaption font color in hex code
subcaptionFontBold	Enable subcaption font to bold

## Examples

```
library(fusionchartsR)  
  
mtcars %>%  
  fusionPlot(  
    x = "cyl",  
    y = "mpg",  
    type = "boxandwhisker2d"  
  ) %>%  
  fusionCaption(caption = "Caption on the left", captionAlignment = "left") %>%  
  fusionSubcaption(subcaption = "subcaption too") %>%  
  fusionPalette(palettecolors = c("#5D62B5", "#979AD0")) %>%  
  fusionTheme(theme = "fusion")
```

---

fusionTheme	<i>Adding FusionCharts theme</i>
-------------	----------------------------------

---

**Description**

<https://www.fusioncharts.com/dev/themes/introduction-to-themes>

**Usage**

```
fusionTheme(
  fusionPlot,
  theme = c("fusion", "gammel", "candy", "zune", "ocean", "carbon", "umber", "fint",
    "access", "boilerplate-basic", "boilerplate-extended")
)
```

**Arguments**

fusionPlot	fusionPlot object got by fusionPlot()
theme	Chart theme

**Examples**

```
library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
df %>%
  fusionPlot(x = "label", y = "value", type = "pie2d") %>%
  fusionTheme(theme = "gammel")
```

---

fusionTooltip	<i>Adding FusionCharts tooltip</i>
---------------	------------------------------------

---

**Description**

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/tool-tips>

**Usage**

```
fusionTooltip(
  fusionPlot,
  showToolTip = TRUE,
  tooltipBorderColor = "#666666",
  tooltipBgColor = "#ffffff",
  tooltipBgAlpha = "100",
  showToolTipShadow = TRUE
)
```

**Arguments**

**fusionPlot** fusionPlot object got by fusionPlot()  
**showToolTip** Display tooltip  
**toolTipBorderColor**  
                   Specify the color of the tooltip border  
**toolTipBgColor** Specify the hex code for the tooltip background color  
**toolTipBgAlpha** Set the tooltip background color transparency  
**showToolTipShadow**  
                   Enable tooltip shadow

**Examples**

```

library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
df %>%
  fusionPlot(x = "label", y = "value", type = "doughnut2d") %>%
  fusionTooltip(toolTipBgColor = "#3526ad", toolTipBgAlpha = "50", showToolTipShadow = FALSE) %>%
  fusionTheme(theme = "fusion")

```

---

 fusionTrendline

*Adding FusionCharts trend-line/trend-zone*


---

**Description**

<https://www.fusioncharts.com/dev/chart-guide/chart-configurations/trend-lines-and-zones>

**Usage**

```

fusionTrendline(
  fusionPlot,
  isTrendZone = FALSE,
  displayValue = "",
  startValue = "",
  endValue = "",
  color = "#FFFFFF",
  thickness = 2,
  alpha = "60",
  valueOnRight = TRUE,
  trendValueFont = "Arial",
  trendValueFontSize = "12",
  trendValueFontBold = TRUE,
  trendValueFontItalic = FALSE,
  trendValueAlpha = "80",
  trendValueBgColor = "#000000",
  trendValueBgAlpha = "10",

```

```

trendValueBorderColor = "#000000",
trendValueBorderAlpha = "80",
trendValueBorderPadding = "4",
trendValueBorderRadius = "5",
trendValueBorderThickness = "2",
trendValueBorderDashed = FALSE,
trendValueBorderDashLen = "#5A5A5A",
trendValueBorderDashGap = "1"
)

```

### Arguments

fusionPlot	fusionPlot object got by fusionPlot()
isTrendZone	Render a trend zone on a chart
displayValue	Add text next to the trend-line
startValue	Specify the data value of the starting point of the trend-line
endValue	Specify the data value of the ending point of the trend-line
color	Specify the hex code for the color of the trend-line
thickness	Specify the thickness of the trend-line(in pixels)
alpha	Specify the transparency of the trend-line
valueOnRight	Enable right position
trendValueFont	Set the font family for the trend-line display values
trendValueFontSize	Set the font size for the trend-line display values
trendValueFontBold	Make trend-line display values appear in bold
trendValueFontItalic	Make trend-line display values appear in italic
trendValueAlpha	Set the transparency for the trend-line display values
trendValueBgColor	Set the color for the background of the trend-line display values
trendValueBgAlpha	Set the transparency for the background of trend-line display values
trendValueBorderColor	Set the color for the border around the trend-line display values
trendValueBorderAlpha	Set the transparency for the border around the trend-line display values (0 to 100)
trendValueBorderPadding	Set the padding for the border around the trend-line display values
trendValueBorderRadius	Set the radius for the border around the trend-line display values

`trendValueBorderThickness`  
 Set the thickness of the border around the trend-line display values  
`trendValueBorderDashed`  
 Specify whether the border around the trend-line display value will be drawn as a dashed line  
`trendValueBorderDashLen`  
 Set the length of each dash  
`trendValueBorderDashGap`  
 Set the gap between each dash

### Examples

```

library(fusionchartsR)
df <- data.frame(label = c("Venezuela", "Saudi", "Canada", "Russia"), value = c(290, 260, 180, 115))
df %>%
  fusionPlot(x = "label", y = "value", type = "column2d") %>%
  fusionTrendline(displayValue = "Help", startValue = "100") %>%
  fusionTheme(theme = "candy")
  
```

---

ggfusionPlot

*Convert a ggplot2 object to a fusionchartsR object.*

---

### Description

[Experimental]

### Usage

```
ggfusionPlot(object)
```

### Arguments

object            a ggplot object

### Examples

```

library(ggplot2)
library(fusionchartsR)

df <- data.frame(
  label = rep(x = c(2012:2016), times = 2),
  seriesname = c(rep("iOS App Store", 5), rep("Google Play Store", 5)),
  values = c(1:10)
)

object <- ggplot(df, aes(label, values, fill = seriesname)) +
  geom_col()

ggfusionPlot(object)
  
```

---

runDemo

*Fusioncharts Demo*

---

**Description**

Running Shiny App

**Usage**

```
runDemo()
```

**Examples**

```
if (interactive()) {  
  library(shiny)  
  library(fusionchartsR)  
  runDemo()  
}
```

# Index

- fusionAnchors, [2](#)
- fusionAxis, [3](#)
- fusionBackground, [4](#)
- fusionCanvas, [5](#)
- fusionCaption, [7](#)
- fusionCustomAxis, [8](#)
- fusionCustomBoxplot, [10](#)
- fusionCustomLegend, [12](#)
- fusionDiv, [14](#)
- fusionGrid, [16](#)
- fusionGrid-shiny, [17](#)
- fusionGridOutput (fusionGrid-shiny), [17](#)
- fusionLegend, [18](#)
- fusionLogo, [19](#)
- fusionMultiPlot, [20](#)
- fusionMultiPlot-shiny, [21](#)
- fusionMultiPlotOutput
  - (fusionMultiPlot-shiny), [21](#)
- fusionPalette, [21](#)
- fusionPlot, [23](#)
- fusionPlotOutput, [24](#)
- fusionSubcaption, [25](#)
- fusionTheme, [26](#)
- fusionTooltip, [26](#)
- fusionTrendline, [27](#)
  
- ggfusionPlot, [29](#)
  
- renderFusionGrid (fusionGrid-shiny), [17](#)
- renderFusionMultiPlot
  - (fusionMultiPlot-shiny), [21](#)
- renderFusionPlot (fusionPlotOutput), [24](#)
- runDemo, [30](#)