Package: gex (via r-universe)

May 9, 2026

Title A Dependency-Light Hex-Logo Builder

Version 0.2.2

Description A simple interface to create hexagon-shaped logos that help promote your R package or other projects. Uses the 'grid' system.

License MIT + file LICENSE

URL https://github.com/matt-dray/gex

BugReports https://github.com/matt-dray/gex/bugs

Imports grDevices, grid, gridGeometry

Suggests magick, png, rsvg, svglite, tinysnapshot, tinytest

Config/testthat/edition 3

Encoding UTF-8

Issues https://github.com/matt-dray/gex

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Repository https://matt-dray.r-universe.dev

Date/Publication 2025-10-10 18:46:53 UTC

RemoteUrl https://github.com/matt-dray/gex

RemoteRef HEAD

RemoteSha 12236c166f66edfff748ed6e2341b6ab65aa58ef

Contents

add_border *Add a Border to the Edge of the Hexagon*

Description

Add a border of given thickness and colour to the inner edges of hexagon.

Usage

```
add_border(width = 0.05, col = "black")
```

Arguments

width	Numeric. Thickness of the border, expressed as the inverse ratio of the interior of the hex to the full extent of the hex (must be between 0 and 1).
col	Character. Named R colour or hexadecimal code for the border around the hex.

Details

Order:

When building a hex, this function should be called after open_device, add_hex and any calls to add_text and add_image (in that order), and before close_device.

Colours:

Named colour values must be listed in `grDevices::colours()`. Hexadecimal colour values must be provided with length 6 or 8 and must begin with an octothorpe (#).

Value

NULL. Adds to an existing graphics device.

See Also

Other hex elements: `add_hex()`

Examples

```
temp_path <- tempfile(fileext = ".png")
open_device(temp_path)
add_hex()
img_path <- system.file("img", "Rlogo.png", package = "png")
img_png <- png::readPNG(img_path)
add_image(img_png)
add_text()
add_border()
close_device()
```

add_hex *Add a Hexagon*

Description

Add a hexagon 'canvas' to which elements can be added.

Usage

```
add_hex(col = "grey")
```

Arguments

col	Character. Named R colour or hexadecimal code for the interior background.

Details

Order:

When building a hex, this function should be called after open_device. You can then use add_text, add_image and add_border (if desired) and finally close_device.

Colours:

Named colour values must be listed in `grDevices::colours()`. Hexadecimal colour values must be provided with length 6 or 8 and must begin with an octothorpe (#).

Value

NULL. Adds to an existing graphics device.

See Also

Other hex elements: `add_border()`

Examples

```
temp_path <- tempfile(fileext = ".png")
open_device(temp_path)
add_hex()
img_path <- system.file("img", "Rlogo.png", package = "png")
img_png <- png::readPNG(img_path)
add_image(img_png)
add_text()
add_border()
close_device()
```

add_image *Add an Image to the Hexagon*

Description

Overlay an image on the hexagon. Call this function separately for each image you want to add.

Usage

```
add_image(img, x = 0.5, y = 0.7, angle = 0, width = 0.4)
```

Arguments

img	Array. A PNG or JPEG file read in by the user, most likely using packages 'png' or 'jpeg'.
x	Numeric. Image location on the hexagon's x-axis.
y	Numeric. Image location on the hexagon's y-axis.
angle	Numeric. Text rotation in degrees.
width	Numeric. Image width.

Details

Order:
When building a hex, this function should be called after open_device and add_hex. You can then use further calls to add_image, add_text and add_border (if desired) and finally close_device.

Coordinates:
Coordinates should be provided within the x- and y-axis ranges, which are both from 0 to 1, giving the centre as x = 0.5 and y = 0.5.

Value

NULL. Adds to an existing graphics device.

See Also

Other hex content adders: add_text()

Examples

```
temp_path <- tempfile(fileext = ".png")
open_device(temp_path)
add_hex()
img_path <- system.file("img", "Rlogo.png", package = "png")
img_png <- png::readPNG(img_path)
add_image(img_png)
add_text()
add_border()
close_device()
```

`add_text` *Add Text to the Hexagon*

Description

Overlay text on the hexagon. Call this function separately for each string you want to add.

Usage

```
add_text(
  string = "example",
  x = 0.5,
  y = 0.4,
  angle = 0,
  size = 20,
  col = "black",
  family = "sans",
  face = c("plain", "bold", "italic", "bold.italic")
)
```

Arguments

`string`	Character. Text to display. `NULL` (or an empty string) if you don't want to place text.
`x`	Numeric. Text location on the hexagon's x-axis.
`y`	Numeric. Text location on the hexagon's y-axis.
`angle`	Numeric. Rotation of text string in degrees. Positive values will rotate anti-clockwise by the given angle.
`size`	Numeric. Text point-size.
`col`	Character. Text colour. A named R colour or hexadecimal code.
`family`	Character. Name of a font family available on your system.
`face`	Character. Font face for the text.

Details

Order:

When building a hex, this function should be called after open_device and add_hex. You can then use further calls to add_text, add_image and add_border (if desired) and finally close_device.

Coordinates:

Coordinates should be provided within the x- and y-axis ranges, which are both from 0 to 1, giving the centre as x = 0.5 and y = 0.5.

Colours:

Named colour values must be listed in `grDevices::colours()`. Hexadecimal colour values must be provided with length 6 or 8 and must begin with an octothorpe (#).

Value

NULL. Adds to an existing graphics device.

See Also

Other hex content adders: add_image()

Examples

```
temp_path <- tempfile(fileext = ".png")
open_device(temp_path)
add_hex()
img_path <- system.file("img", "Rlogo.png", package = "png")
img_png <- png::readPNG(img_path)
add_image(img_png)
add_text()
add_border()
close_device()
```

---

close_device *Close the Device and Write to File*

Description

Clip to the area of the outer hexagon and shut down the PNG plot device, which writes to the file_path specified in open_device.

Usage

```
close_device()
```

Details

Order:

When building a hex, this function should be called at the end, after open_device, add_hex and any calls to add_text, add_image and add_border.

Value

Named numeric. The device name and number where the hex has been written.

See Also

Other hex device handlers: open_device()

Examples

```
temp_path <- tempfile(fileext = ".png")
open_device(temp_path)
add_hex()
img_path <- system.file("img", "Rlogo.png", package = "png")
img_png <- png::readPNG(img_path)
add_image(img_png)
add_text()
add_border()
close_device()
```

---

open_device *Open a PNG Device with Sticker-Standard Dimensions*

Description

Begin a PNG plot device with dimensions matching the Stickers Standard: 4.39 cm wide by 5.08 cm high (2 by 1.73 inches).

Usage

```
open_device(file_path, resolution = 300)
```

Arguments

`file_path`	Character. File path to a .png where the output file will be saved. The containing directory must already exist.
`resolution`	Numeric. Resolution of the graphics device in pixels per inch (ppi). Higher values have better resolution but create larger file sizes.

Details

Order:

When building a hex, this function should be called first, followed by add_hex. You can then use add_text, add_image and add_border (if desired) and finally close_device.

Value

Nothing. A graphics device is opened.

See Also

Other hex device handlers: `close_device()`

Examples

```
temp_path <- tempfile(fileext = ".png")
open_device(temp_path)
add_hex()
img_path <- system.file("img", "Rlogo.png", package = "png")
img_png <- png::readPNG(img_path)
add_image(img_png)
add_text()
add_border()
close_device()
```

Index