

# Package ‘monochromeR’

May 9, 2026

**Title** Easily Create, View and Use Monochrome Colour Palettes

**Version** 0.2.0

**Description** Generate a monochrome palette from a starting colour for a specified number of colours. The package can also be used to display colour palettes in the plot window, with or without hex codes and colour labels.

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.1.2

**URL** <https://github.com/cararthompson/monochromeR>

**BugReports** <https://github.com/cararthompson/monochromeR/issues>

**Maintainer** Cara Thompson <packages@cararthompson.com>

**Imports** ggplot2, magrittr

**NeedsCompilation** no

**Author** Cara Thompson [aut, cre] (ORCID:  
<<https://orcid.org/0000-0002-6472-5073>>)

**Repository** CRAN

**Date/Publication** 2023-09-06 01:20:15 UTC

## Contents

check_colour_return_rgb . . . . .	2
generate_palette . . . . .	2
hex_to_rgb . . . . .	4
rgba_to_hex . . . . .	4
rgba_to_rgb . . . . .	5
rgb_to_hex . . . . .	6
view_palette . . . . .	6
<b>Index</b>	<b>8</b>

check\_colour\_return\_rgb

*Checks colour variables are either RGB values, hex colour codes or a recognised colour name and converts to rgb (helper funct)*

---

### **Description**

Checks colour variables are either RGB values, hex colour codes or a recognised colour name and converts to rgb (helper funct)

### **Usage**

```
check_colour_return_rgb(colour, colour_variable_name)
```

### **Arguments**

colour            The colour string / rgb vector to check  
colour\_variable\_name    The name of the variable, for readability of error messages

### **Value**

An error message if the colour value can't be interpreted

### **Examples**

```
check_colour_return_rgb("White", "test_colour")  
## Not run: check_colour_return_rgb("foo", "test_colour")
```

---

generate\_palette

*Generate a monochrome palette*

---

### **Description**

This function allows users generate a monochrome colour palette containing any number of colours, starting from the colour they specify. The modification parameter can be set to make the palette go darker, lighter, or both ways from the starting colour. The function also allows users to create a palette that goes from one colour to another, by providing a blend\_colour.

**Usage**

```
generate_palette(
  colour,
  modification,
  n_colours,
  blend_colour = NULL,
  view_palette = FALSE,
  view_labels = TRUE,
  ...
)
```

**Arguments**

colour	The starting colour for the palette, which must be either be a recognised colour name (e.g. "white"), a hex colour code (e.g. "#ffffff") or vector of length 3 (red value, green value, blue value, e.g. c(15, 75, 99)), with all values between 0 and 255.
modification	One of the following: "go_darker", "go_lighter", "go_both_ways", or "blend". If a blend_colour is supplied, modification is automatically set to "blend".
n_colours	Number of colours (levels) required in the palette
blend_colour	Optional. Can be either be a recognised colour name (e.g. "white"), a hex colour code (e.g. "#ffffff") or vector of length 3 (red value, green value, blue value, e.g. c(15, 75, 99)), with all values between 0 and 255.
view_palette	Logical. view_palette = TRUE displays the palette in the plot window.
view_labels	Logical. If view_palette is set to TRUE, view_labels = FALSE determines whether or not the hex colour codes are shown on the palette displayed in the plot window.
...	Allows for US spelling of color/colour.

**Value**

A vector of hex colour codes making up the generated palette

**Examples**

```
generate_palette("red", modification = "go_lighter",
  n_colours = 5, view_palette = TRUE, view_labels = TRUE)

generate_palette(c(15, 75, 99), modification = "go_both_ways",
  n_colours = 12, view_palette = TRUE, view_labels = FALSE)

generate_palette("red", blend_colour = "blue",
  n_colours = 6, view_palette = TRUE)
```

---

hex_to_rgb	<i>Converts Hex codes values to RGB vectors</i>
------------	---

---

**Description**

Converts Hex codes values to RGB vectors

**Usage**

```
hex_to_rgb(x)
```

**Arguments**

x	A hex colour code
---	-------------------

**Value**

A corresponding matrix of red, blue and green values

**Examples**

```
hex_to_rgb("purple")
hex_to_rgb("#fafafa")
```

---

rgba_to_hex	<i>rgba_to_hex</i>
-------------	--------------------

---

**Description**

rgba\_to\_hex

**Usage**

```
rgba_to_hex(colour_rgba, background_colour = "#ffffff", ...)
```

**Arguments**

colour_rgba	A vector of length 4: c(red value, green value, blue value, alpha). All colour values must be between 0 and 255. Alpha must be between 0 and 1.
background_colour	Defaults to white. Users can specify a different colour to get the hex code for their original colour blended with a specified background colour. background_colour must either be a recognised colour name (e.g. "white"), a hex colour code (e.g. "#ffffff") or vector of length 3 (red value, green value, blue value), with all values between 0 and 255. The default value is white ("#ffffff").
...	Allows for US spelling of color/colour.

**Value**

Returns the corresponding hex colour code

**Examples**

```
rgba_to_hex(c(52, 46, 39, 0.8))
rgba_to_hex(c(52, 46, 39, 0.8), "blue")
rgba_to_hex(c(52, 46, 39, 0.8), "#032cfc")
```

rgba\_to\_rgb *Converts RGBA to RGB (helper function)*

**Description**

Converts RGBA to RGB (helper function)

**Usage**

```
rgba_to_rgb(colour_rgba, background_colour = "#ffffff", ...)
```

**Arguments**

- colour\_rgba     A vector of length 4: c(red value, green value, blue value, alpha). All colour values must be between 0 and 255. Alpha must be between 0 and 1.
- background\_colour     Defaults to white. Users can specify a different colour to get the hex code for their original colour blended with a specified background colour. background\_colour must either be a recognised colour name (e.g. "white"), a hex colour code (e.g. "#ffffff") or vector of length 3 (red value, green value, blue value), with all values between 0 and 255. The default value is white ("#ffffff").
- ...             Allows for US spelling of color/colour.

**Value**

A matrix of red, green and blue values

**Examples**

```
rgba_to_rgb(c(52, 46, 39, 0.8))
rgba_to_rgb(c(52, 46, 39, 0.8), "blue")
rgba_to_rgb(c(52, 46, 39, 0.8), "#032cfc")
```

---

rgb_to_hex	<i>Converts RGB values to hex colour code</i>
------------	---

---

**Description**

Converts RGB values to hex colour code

**Usage**

```
rgb_to_hex(x)
```

**Arguments**

x                    A matrix of red, blue and green values

**Value**

A corresponding hex colour code

**Examples**

```
temp_rgb_matrix <- rgba_to_rgb(c(52, 46, 39, 0.8))
rgb_to_hex(temp_rgb_matrix)
```

---

view_palette	<i>Easy way to view the created palette</i>
--------------	---

---

**Description**

Easy way to view the created palette

**Usage**

```
view_palette(monochrome_palette, view_labels = TRUE)
```

**Arguments**

monochrome\_palette

Vector of hex colour codes, or a generate\_palette() call

view\_labels

Logical. If view\_palette is set to TRUE, view\_labels determines whether or not the hex colour codes are shown on the palette displayed in the plot window.

**Value**

A plot showing all the colours in the palette on the same row

**Examples**

```
view_palette(c("#464E69", "#8C90A1", "#D1D2D9"))  
view_palette(c("unripe" = "#89973d", "ripe" = "#e8b92f", "overripe" = "#a45e41"))  
view_palette(generate_palette("pink", "go_darker", n_colours = 3))
```

# Index

`check_colour_return_rgb`, 2

`generate_palette`, 2

`hex_to_rgb`, 4

`rgb_to_hex`, 6

`rgba_to_hex`, 4

`rgba_to_rgb`, 5

`view_palette`, 6