

# Package ‘DOYPAColors’

November 4, 2023

**Type** Package

**Title** Don't Overthink Your Palette of Colors

**Version** 0.0.1

**Description** Access diverse 'ggplot2'-compatible color palettes for simplified data visualization.

**License** MIT + file LICENSE

**URL** <https://github.com/jmestret/DOYPAColors>

**BugReports** <https://github.com/jmestret/DOYPAColors/issues>

**Depends** R (>= 3.5.0)

**Imports** ggplot2, grDevices

**Suggests** knitr, rmarkdown, testthat

**Encoding** UTF-8

**Language** en-US

**RoxygenNote** 7.2.3

**NeedsCompilation** no

**Author** Jorge Mestre [aut, cre] (<<https://orcid.org/0000-0002-8983-3417>>)

**Maintainer** Jorge Mestre <jormart2@alumni.uv.es>

**Repository** CRAN

**Date/Publication** 2023-11-04 07:50:02 UTC

## R topics documented:

doypa . . . . .	2
doypa_palette . . . . .	3
list_doypa_pals . . . . .	3
preview_doypa_pals . . . . .	4
scale_color_doypa . . . . .	5
scale_colour_doypa . . . . .	6
scale_fill_doypa . . . . .	7
<b>Index</b>	<b>9</b>

---

doypa	<i>Get a DOYPAColors color palette</i>
-------	----------------------------------------

---

### Description

This function generates a color vector of n colors for a specified palette.

### Usage

```
doypa(palette = NULL, n = NULL, reverse = FALSE, ramp = FALSE)
```

### Arguments

palette	Character string specifying the desired palette name. Available palettes: <code>list_doypa_pals()</code> .
n	Number of colors needed.
reverse	Logical. If 'TRUE', reverses the order of colors in the palette.
ramp	Logical. If 'TRUE', returns a color ramp (interpolation) instead of a vector of colors.

### Value

A vector of 'n' colors.

### See Also

'list\_doypa\_pals' to list available palettes. 'preview\_doypa\_pals' to preview color palettes.

### Examples

```
# Generate a random color palette
color_palette <- doypa()
preview_doypa_pals(colors = color_palette)

# Generate a color palette of 5 colors using the "buzz" palette
color_palette <- doypa(palette = "buzz", n = 5)
preview_doypa_pals(colors = color_palette, palette = "buzz")
```

---

doypa_palette	<i>Create a DOYPAColors color palette function</i>
---------------	----------------------------------------------------

---

**Description**

This function creates a function that generates a color vector or ramp of n colors for a specified DOYPAColors palette.

**Usage**

```
doypa_palette(palette = NULL, reverse = FALSE, ramp = FALSE)
```

**Arguments**

palette	Character string specifying the desired palette name. Available palettes: list_doypa_pals().
reverse	Logical. If 'TRUE', reverses the order of colors in the palette.
ramp	Logical. If 'TRUE', the returned function generates a color ramp (interpolation) instead of a vector of colors.

**Value**

A function that generates a vector of 'n' colors or a color ramp.

**Examples**

```
# Create a function for generating colors from the "buzz" palette
get_buzz_palette <- doypa_palette(palette = "buzz")
colors <- get_buzz_palette(5)
preview_doypa_pals(colors = colors, palette = "buzz")
```

---

list_doypa_pals	<i>List available DOYPAColors color palettes</i>
-----------------	--------------------------------------------------

---

**Description**

This function provides a list of all available DOYPAColors color palettes.

**Usage**

```
list_doypa_pals()
```

**Value**

A character vector containing the names of available color palettes.

## Examples

```
# List available DOYPAColors color palettes
palette_names <- list_doypa_pals()
print(palette_names)
```

---

preview\_doypa\_pals      *Preview DOYPAColors palettes*

---

## Description

This function generates a visual preview of DOYPAColors color palettes, allowing you to explore and select color schemes for your data visualizations.

## Usage

```
preview_doypa_pals(palette = NULL, colors = NULL)
```

## Arguments

palette	Character string specifying the name of a specific DOYPAColors palette to preview. If provided, this overrides 'colors'.
colors	A vector of colors to preview.

## Value

A 'ggplot' visualization of DOYPAColors color palettes for exploring and selecting color schemes in your data visualizations.

## Examples

```
# Preview a specific DOYPAColors palette by name
preview_doypa_pals(palette = "buzz")

# Preview a custom vector of colors
custom_colors <- c("#FF5733", "#33FF57", "#5733FF")
preview_doypa_pals(colors = custom_colors)

# Preview all available DOYPAColors palettes
preview_doypa_pals()
```

---

scale\_color\_doypa      *Color scale for ggplot2 with DOYPAColors color palettes*

---

### Description

This function sets the color scale for ggplot2 using DOYPAColors color palettes.

### Usage

```
scale_color_doypa(  
  palette = NULL,  
  n = NULL,  
  reverse = FALSE,  
  ramp = FALSE,  
  discrete = FALSE,  
  ...  
)
```

### Arguments

palette	Character string specifying the desired palette name. Available palettes: list_doypa_pals().
n	Number of colors needed.
reverse	Boolean indicating whether to reverse the order of colors in the palette (default: FALSE).
ramp	Boolean indicating whether to use a color ramp (gradient) or a vector of colors (default: FALSE).
discrete	Boolean indicating whether to generate a discrete or continuous palette (default: continuous).
...	Additional parameters to pass to ggplot2's scale_color functions.

### Value

A 'ggplot2' color scale suitable for adding to a 'ggplot2' object to control color aesthetics.

### Examples

```
library(ggplot2)  
  
# Discrete data  
data(iris)  
disc <- ggplot(iris, aes(x = Petal.Width, y = Petal.Length, color = Species)) +  
  geom_point() + theme_classic()  
disc <- disc + scale_color_doypa(palette = "buzz", discrete = TRUE)  
print(disc)  
  
# Continuous data  
cont <- ggplot(iris, aes(x = Petal.Width, y = Petal.Length, color = Sepal.Length)) +
```

```
geom_point() + theme_classic()
cont <- cont + scale_color_doypa(palette = "buzz")
print(cont)
```

---

scale\_colour\_doypa      *Color scale for ggplot2 with DOYPAColors color palettes*

---

### Description

This function sets the color scale for ggplot2 using DOYPAColors color palettes.

### Usage

```
scale_colour_doypa(
  palette = NULL,
  n = NULL,
  reverse = FALSE,
  ramp = FALSE,
  discrete = FALSE,
  ...
)
```

### Arguments

palette	Character string specifying the desired palette name. Available palettes: list_doypa_pals().
n	Number of colors needed.
reverse	Boolean indicating whether to reverse the order of colors in the palette (default: FALSE).
ramp	Boolean indicating whether to use a color ramp (gradient) or a vector of colors (default: FALSE).
discrete	Boolean indicating whether to generate a discrete or continuous palette (default: continuous).
...	Additional parameters to pass to ggplot2's scale_color functions.

### Value

A 'ggplot2' color scale suitable for adding to a 'ggplot2' object to control color aesthetics.

---

scale\_fill\_doypa      *Fill scale for ggplot2 with DOYPAColors color palettes*

---

### Description

This function sets the fill scale for ggplot2 using DOYPAColors color palettes.

### Usage

```
scale_fill_doypa(  
  palette = NULL,  
  n = NULL,  
  reverse = FALSE,  
  ramp = FALSE,  
  discrete = FALSE,  
  ...  
)
```

### Arguments

palette	Character string specifying the desired palette name. Available palettes: list_doypa_pals().
n	Number of colors needed.
reverse	Boolean indicating whether to reverse the order of colors in the palette (default: FALSE).
ramp	Boolean indicating whether to use a color ramp (gradient) or a vector of colors (default: FALSE).
discrete	Boolean indicating whether to generate a discrete or continuous palette (default: continuous).
...	Additional parameters to pass to ggplot2's scale_fill functions.

### Value

A 'ggplot2' fill scale suitable for adding to a 'ggplot2' object to control fill aesthetics.

### Examples

```
library(ggplot2)  
  
# Discrete data  
data(iris)  
disc <- ggplot(iris, aes(x = Species, y = Petal.Length, fill = Species)) +  
  geom_boxplot() + theme_classic()  
disc <- disc + scale_fill_doypa(palette = "buzz", discrete = TRUE)  
print(disc)  
  
# Continuous data  
cont <- ggplot(faithfuld, aes(waiting, eruptions, fill = density)) +
```

```
  geom_tile() + theme_classic()
cont <- cont + scale_fill_doypa(palette = "buzz")
print(cont)
```



# Index

`doypa`, [2](#)

`doypa_palette`, [3](#)

`list_doypa_pals`, [3](#)

`preview_doypa_pals`, [4](#)

`scale_color_doypa`, [5](#)

`scale_colour_doypa`, [6](#)

`scale_fill_doypa`, [7](#)