

# Package ‘tvthemes’

September 3, 2019

**Type** Package

**Title** TV Show Themes and Color Palettes for 'ggplot2' Graphics

**Version** 1.0.0

**Maintainer** Ryo Nakagawara <ryonakagawara@gmail.com>

**Description** Contains various 'ggplot2' themes and color palettes based on TV shows such as 'Game of Thrones', 'Brooklyn Nine-Nine', 'Avatar: The Last Airbender', 'Spongebob Squarepants', and more.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.1

**Imports** ggplot2 (>= 3.1.0), extrafont (>= 0.17), scales (>= 1.0.0), magick (>= 2.0), glue (>= 1.3.1), grDevices (>= 3.5.3)

**Suggests** testthat (>= 2.1.1), gapminder (>= 0.3.0), dplyr (>= 0.8.0.1), spelling (>= 2.0), cowplot (>= 0.9.4), png (>= 0.1-7)

**URL** <https://github.com/Ryo-N7/tvthemes>

**BugReports** <https://github.com/Ryo-N7/tvthemes/issues>

**Language** en-US

**NeedsCompilation** no

**Author** Ryo Nakagawara [aut, cre]

**Repository** CRAN

**Date/Publication** 2019-09-03 07:30:02 UTC

## R topics documented:

attackOnTitan_pal . . . . .	2
avatarTLA_pal . . . . .	4
bigHero6_pal . . . . .	6
brooklyn99_pal . . . . .	8

hilda_pal . . . . .	10
import_ChelseaMarket . . . . .	12
import_cinzel . . . . .	12
import_rickAndMorty . . . . .	13
import_roboto_condensed . . . . .	13
import_simpsons . . . . .	14
import_spongeBob . . . . .	14
import_theLastAirbender . . . . .	15
import_titillium_web . . . . .	15
kimPossible_pal . . . . .	16
paintBikiniBottom . . . . .	17
parksAndRec_pal . . . . .	18
rickAndMorty_pal . . . . .	20
simpsons_pal . . . . .	22
spongeBob_pal . . . . .	23
sponge_images . . . . .	25
theme_brooklyn99 . . . . .	26
theme_hildaDay . . . . .	27
theme_hildaDusk . . . . .	29
theme_hildaNight . . . . .	30
theme_parksAndRec . . . . .	31
theme_parksAndRecLight . . . . .	33
theme_parksAndRec_light . . . . .	34
theme_rickAndMorty . . . . .	36
theme_simpsons . . . . .	37
theme_spongeBob . . . . .	39
theme_theLastAirbender . . . . .	40
westeros_pal . . . . .	42

## Index 44

---

attackOnTitan_pal	<i>Attack On Titan palette</i>
-------------------	--------------------------------

---

### Description

Attack On Titan palette

### Usage

```
attackOnTitan_pal(n, type = c("discrete", "continuous"),
  reverse = FALSE)
```

```
scale_color_attackOnTitan(n, type = "discrete", reverse = FALSE, ...)
```

```
scale_colour_attackOnTitan(n, type = "discrete", reverse = FALSE, ...)
```

```
scale_fill_attackOnTitan(n, type = "discrete", reverse = FALSE, ...)
```

**Arguments**

n	number of colors
type	discrete or continuous
reverse	reverse order, Default: FALSE
...	Arguments passed on to <code>ggplot2::discrete_scale</code>
<b>aesthetics</b>	The names of the aesthetics that this scale works with
<b>scale_name</b>	The name of the scale
<b>palette</b>	A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take.
<b>name</b>	The name of the scale. Used as the axis or legend title. If <code>waiver()</code> , the default, the name of the scale is taken from the first mapping used for that aesthetic. If <code>NULL</code> , the legend title will be omitted.
<b>breaks</b>	One of: <ul style="list-style-type: none"> <li>• <code>NULL</code> for no breaks</li> <li>• <code>waiver()</code> for the default breaks computed by the transformation object</li> <li>• A character vector of breaks</li> <li>• A function that takes the limits as input and returns breaks as output</li> </ul>
<b>labels</b>	One of: <ul style="list-style-type: none"> <li>• <code>NULL</code> for no labels</li> <li>• <code>waiver()</code> for the default labels computed by the transformation object</li> <li>• A character vector giving labels (must be same length as breaks)</li> <li>• A function that takes the breaks as input and returns labels as output</li> </ul>
<b>limits</b>	A character vector that defines possible values of the scale and their order.
<b>expand</b>	Vector of range expansion constants used to add some padding around the data, to ensure that they are placed some distance away from the axes. Use the convenience function <code>expand_scale()</code> to generate the values for the <code>expand</code> argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.
<b>na.translate</b>	Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify <code>na.translate = FALSE</code> .
<b>na.value</b>	If <code>na.translate = TRUE</code> , what value aesthetic value should missing be displayed as? Does not apply to position scales where NA is always placed at the far right.
<b>drop</b>	Should unused factor levels be omitted from the scale? The default, <code>TRUE</code> , uses the levels that appear in the data; <code>FALSE</code> uses all the levels in the factor.
<b>guide</b>	A function used to create a guide or its name. See <code>guides()</code> for more info.
<b>position</b>	The position of the axis. "left" or "right" for vertical scales, "top" or "bottom" for horizontal scales
<b>super</b>	The super class to use for the constructed scale

**Examples**

```

library(scales)
show_col(attackOnTitan_pal()(5))

library(ggplot2)
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_attackOnTitan()

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_colour_attackOnTitan()

ggplot(mpg, aes(displ)) +
  geom_histogram(aes(fill = class),
    col = "black", size = 0.1) +
  scale_fill_attackOnTitan()

```

---

avatarTLA\_pal

*Avatar: The Last Airbender palette*


---

**Description**

Avatar: The Last Airbender palette

**Usage**

```

avatarTLA_pal(palette = "FireNation", n, type = c("discrete",
  "continuous"), reverse = FALSE)

scale_color_avatarTLA(palette = "FireNation", n, type = "discrete",
  reverse = FALSE, ...)

scale_colour_avatarTLA(palette = "FireNation", n, type = "discrete",
  reverse = FALSE, ...)

scale_fill_avatarTLA(palette = "FireNation", n, type = "discrete",
  reverse = FALSE, ...)

```

**Arguments**

palette	name of palette (FireNation, EarthKingdom, WaterTribe, AirNomads), Default: "FireNation"
n	number of colors
type	discrete or continuous

reverse reverse order, Default: FALSE

... Arguments passed on to `ggplot2::discrete_scale`

**aesthetics** The names of the aesthetics that this scale works with

**scale\_name** The name of the scale

**palette** A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take.

**name** The name of the scale. Used as the axis or legend title. If `waiver()`, the default, the name of the scale is taken from the first mapping used for that aesthetic. If `NULL`, the legend title will be omitted.

**breaks** One of:

- `NULL` for no breaks
- `waiver()` for the default breaks computed by the transformation object
- A character vector of breaks
- A function that takes the limits as input and returns breaks as output

**labels** One of:

- `NULL` for no labels
- `waiver()` for the default labels computed by the transformation object
- A character vector giving labels (must be same length as breaks)
- A function that takes the breaks as input and returns labels as output

**limits** A character vector that defines possible values of the scale and their order.

**expand** Vector of range expansion constants used to add some padding around the data, to ensure that they are placed some distance away from the axes. Use the convenience function `expand_scale()` to generate the values for the `expand` argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.

**na.translate** Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.

**na.value** If `na.translate = TRUE`, what value aesthetic value should missing be displayed as? Does not apply to position scales where `NA` is always placed at the far right.

**drop** Should unused factor levels be omitted from the scale? The default, `TRUE`, uses the levels that appear in the data; `FALSE` uses all the levels in the factor.

**guide** A function used to create a guide or its name. See `guides()` for more info.

**position** The position of the axis. "left" or "right" for vertical scales, "top" or "bottom" for horizontal scales

**super** The super class to use for the constructed scale

## Examples

```
library(scales)
show_col(avatarTLA_pal()(5))
```

```

library(ggplot2)
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_avatarTLA()

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_colour_avatarTLA()

ggplot(mpg, aes(displ)) +
  geom_histogram(aes(fill = class), col = "black", size = 0.1) +
  scale_fill_avatarTLA()

```

---

bigHero6\_pal

*Big Hero 6 palette*


---

## Description

Big Hero 6 palette

## Usage

```

bigHero6_pal(n, type = c("discrete", "continuous"), reverse = FALSE)

scale_color_bigHero6(n, type = "discrete", reverse = FALSE, ...)

scale_colour_bigHero6(n, type = "discrete", reverse = FALSE, ...)

scale_fill_bigHero6(n, type = "discrete", reverse = FALSE, ...)

```

## Arguments

n	number of colors
type	discrete or continuous
reverse	reverse order, Default: FALSE
...	Arguments passed on to <code>ggplot2::discrete_scale</code>

**aesthetics** The names of the aesthetics that this scale works with

**scale\_name** The name of the scale

**palette** A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take.

**name** The name of the scale. Used as the axis or legend title. If `waiver()`, the default, the name of the scale is taken from the first mapping used for that aesthetic. If `NULL`, the legend title will be omitted.

**breaks** One of:

- NULL for no breaks
- `waiver()` for the default breaks computed by the transformation object
- A character vector of breaks
- A function that takes the limits as input and returns breaks as output

**labels** One of:

- NULL for no labels
- `waiver()` for the default labels computed by the transformation object
- A character vector giving labels (must be same length as breaks)
- A function that takes the breaks as input and returns labels as output

**limits** A character vector that defines possible values of the scale and their order.

**expand** Vector of range expansion constants used to add some padding around the data, to ensure that they are placed some distance away from the axes. Use the convenience function `expand_scale()` to generate the values for the `expand` argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.

**na.translate** Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.

**na.value** If `na.translate = TRUE`, what value aesthetic value should missing be displayed as? Does not apply to position scales where NA is always placed at the far right.

**drop** Should unused factor levels be omitted from the scale? The default, TRUE, uses the levels that appear in the data; FALSE uses all the levels in the factor.

**guide** A function used to create a guide or its name. See `guides()` for more info.

**position** The position of the axis. "left" or "right" for vertical scales, "top" or "bottom" for horizontal scales

**super** The super class to use for the constructed scale

## Examples

```
library(scales)
show_col(bigHero6_pal()(5))
```

```
library(ggplot2)
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_bigHero6()
```

```
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_colour_bigHero6()
```

```
ggplot(mpg, aes(displ)) +
  geom_histogram(aes(fill = class), col = "black", size = 0.1) +
  scale_fill_bigHero6()
```

---

brooklyn99\_pal

*Brooklyn Nine Nine Color and Fill Scales*


---

## Description

Brooklyn Nine Nine Color and Fill Scales

## Usage

```
brooklyn99_pal(palette = "Regular", n = n, type = c("discrete",
  "continuous"), reverse = FALSE)
```

```
scale_color_brooklyn99(palette = "Regular", n = n, type = "discrete",
  reverse = FALSE, ...)
```

```
scale_colour_brooklyn99(palette = "Regular", n = n,
  type = "discrete", reverse = FALSE, ...)
```

```
scale_fill_brooklyn99(palette = "Regular", n = n, type = "discrete",
  reverse = FALSE, ...)
```

## Arguments

**palette** name of palette, Regular or Dark Default: "Regular"

**n** number of colors

**type** discrete or continuous

**reverse** reverse order, Default: FALSE

**...** Arguments passed on to `ggplot2::discrete_scale`

**aesthetics** The names of the aesthetics that this scale works with

**scale\_name** The name of the scale

**palette** A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take.

**name** The name of the scale. Used as the axis or legend title. If `waiver()`, the default, the name of the scale is taken from the first mapping used for that aesthetic. If NULL, the legend title will be omitted.

**breaks** One of:

- NULL for no breaks
- `waiver()` for the default breaks computed by the transformation object
- A character vector of breaks
- A function that takes the limits as input and returns breaks as output



**labels** One of:

- NULL for no labels
- `waiver()` for the default labels computed by the transformation object
- A character vector giving labels (must be same length as breaks)
- A function that takes the breaks as input and returns labels as output

**limits** A character vector that defines possible values of the scale and their order.

**expand** Vector of range expansion constants used to add some padding around the data, to ensure that they are placed some distance away from the axes. Use the convenience function `expand_scale()` to generate the values for the `expand` argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.

**na.translate** Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.

**na.value** If `na.translate = TRUE`, what value aesthetic value should missing be displayed as? Does not apply to position scales where NA is always placed at the far right.

**drop** Should unused factor levels be omitted from the scale? The default, TRUE, uses the levels that appear in the data; FALSE uses all the levels in the factor.

**guide** A function used to create a guide or its name. See `guides()` for more info.

**position** The position of the axis. "left" or "right" for vertical scales, "top" or "bottom" for horizontal scales

**super** The super class to use for the constructed scale

## Details

Colors that work well with the blue background!

## Examples

```
library(scales)
show_col(brooklyn99_pal()(5))
show_col(brooklyn99_pal(palette = "Dark")(5))

library(ggplot2)
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_brooklyn99()

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_brooklyn99(palette = "Dark")
```

```

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_colour_brooklyn99(palette = "Dark")

ggplot(mpg, aes(displ)) +
  geom_histogram(aes(fill = class), col = "black", size = 0.1) +
  scale_fill_brooklyn99()

```

---

hilda\_pal

*Hilda palette*


---

## Description

Hilda palette

## Usage

```
hilda_pal(palette, n, type = c("discrete", "continuous"),
  reverse = FALSE)
```

```
scale_color_hilda(palette = "Day", n, type = "discrete",
  reverse = FALSE, ...)
```

```
scale_colour_hilda(palette = "Day", n, type = "discrete",
  reverse = FALSE, ...)
```

```
scale_fill_hilda(palette = "Day", n, type = "discrete",
  reverse = FALSE, ...)
```

## Arguments

**palette** name of palette (Day, Dusk, Night), Default: "Day"

**n** number of colors

**type** discrete or continuous

**reverse** reverse order, Default: FALSE

**...** Arguments passed on to `ggplot2::discrete_scale`

**aesthetics** The names of the aesthetics that this scale works with

**scale\_name** The name of the scale

**palette** A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take.

**name** The name of the scale. Used as the axis or legend title. If `waiver()`, the default, the name of the scale is taken from the first mapping used for that aesthetic. If `NULL`, the legend title will be omitted.

**breaks** One of:

- NULL for no breaks
- `waiver()` for the default breaks computed by the transformation object
- A character vector of breaks
- A function that takes the limits as input and returns breaks as output

**labels** One of:

- NULL for no labels
- `waiver()` for the default labels computed by the transformation object
- A character vector giving labels (must be same length as breaks)
- A function that takes the breaks as input and returns labels as output

**limits** A character vector that defines possible values of the scale and their order.

**expand** Vector of range expansion constants used to add some padding around the data, to ensure that they are placed some distance away from the axes. Use the convenience function `expand_scale()` to generate the values for the `expand` argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.

**na.translate** Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.

**na.value** If `na.translate = TRUE`, what value aesthetic value should missing be displayed as? Does not apply to position scales where NA is always placed at the far right.

**drop** Should unused factor levels be omitted from the scale? The default, `TRUE`, uses the levels that appear in the data; `FALSE` uses all the levels in the factor.

**guide** A function used to create a guide or its name. See `guides()` for more info.

**position** The position of the axis. "left" or "right" for vertical scales, "top" or "bottom" for horizontal scales

**super** The super class to use for the constructed scale

## Details

Color set from Matt Shanks & '@ChevyRay'

## Examples

```
library(scales)
show_col(hilda_pal(palette = "Dusk")(5))

library(ggplot2)
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_hilda(palette = "Day")

ggplot(airquality, aes(x = Day, y = Temp,
```

```
group = as.factor(Month), color = as.factor(Month))) +
geom_point(size = 2.5) +
scale_color_hilda(palette = "Night")

ggplot(airquality, aes(x = Day, y = Temp,
group = as.factor(Month), color = as.factor(Month))) +
geom_point(size = 2.5) +
scale_colour_hilda(palette = "Day")

ggplot(mpg, aes(displ)) +
geom_histogram(aes(fill = class), col = "black", size = 0.1) +
scale_fill_hilda(palette = "Night")
```

---

import\_ChelseaMarket    *import\_chelseaMarket*

---

### Description

Imports Chelsea Market (Hilda)

### Usage

```
import_ChelseaMarket()
```

### Details

import\_\*() functions taken from hrbrthemes.

### See Also

[font\\_import](#)

---

import\_cinzel    *import\_cinzel*

---

### Description

Game of Thrones font ("Cinzel" font)

### Usage

```
import_cinzel()
```

### Details

import\_\*() functions taken from hrbrthemes, actual font is "Trajan Pro". You may still need to install each font on your system directly by finding the .ttf file and clicking "Install".

**See Also**

[font\\_import](#)

---

*import\_rickAndMorty*    *import\_rickAndMorty*

---

**Description**

Rick & Morty font ("Get Schwifty")

**Usage**

`import_rickAndMorty()`

**Details**

Actual font is ... well, Justin Roiland's actual handwriting. `import_*`() functions taken from `hrbrthemes`. You may still need to install each font on your system directly by finding the `.ttf` file and clicking "Install".

---

*import\_roboto\_condensed*  
*import\_roboto\_condensed*

---

**Description**

taken from `hrbrthemes`

**Usage**

`import_roboto_condensed()`

**Details**

`import_*`() functions taken from `hrbrthemes`. You may still need to install each font on your system directly by finding the `.ttf` file and clicking "Install".

**See Also**

[font\\_import](#)

---

`import_simpsons`      *import\_simpsons*

---

### **Description**

The Simpsons Font ("Akbar" font)

### **Usage**

```
import_simpsons()
```

### **Details**

`import_*`() functions taken from `hrbrthemes`. You may still need to install each font on your system directly by finding the `.ttf` file and clicking "Install".

### **See Also**

[font\\_import](#)

---

`import_spongeBob`      *import\_spongeBob*

---

### **Description**

spongeBob SquarePants font ("Some-Time-Later")

### **Usage**

```
import_spongeBob()
```

### **Details**

`import_*`() functions taken from `hrbrthemes`. You may still need to install each font on your system directly by finding the `.ttf` file and clicking "Install".

---

```
import_theLastAirbender  
    import_theLastAirbender
```

---

### **Description**

The Last Airbender font ("Slayer")

### **Usage**

```
import_theLastAirbender()
```

### **Details**

Actual font is Herculanum. `import_*`() functions taken from `hrbrthemes`. You may still need to install each font on your system directly by finding the `.ttf` file and clicking "Install".

---

```
import_titillium_web  import_titillium_web
```

---

### **Description**

Imports Titillium Web

### **Usage**

```
import_titillium_web()
```

### **Details**

`import_*`() functions taken from `hrbrthemes`.

### **See Also**

[font\\_import](#)

---

kimPossible_pal	<i>Kim Possible palette</i>
-----------------	-----------------------------

---

## Description

Kim Possible palette

## Usage

```
kimPossible_pal(n, type = c("discrete", "continuous"), reverse = FALSE)
```

```
scale_color_kimPossible(n, type = "discrete", reverse = FALSE, ...)
```

```
scale_colour_kimPossible(n, type = "discrete", reverse = FALSE, ...)
```

```
scale_fill_kimPossible(n, type = "discrete", reverse = FALSE, ...)
```

## Arguments

n	number of colors
type	discrete or continuous
reverse	reverse order, Default: FALSE
...	Arguments passed on to <code>ggplot2::discrete_scale</code>

**aesthetics** The names of the aesthetics that this scale works with

**scale\_name** The name of the scale

**palette** A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take.

**name** The name of the scale. Used as the axis or legend title. If `waiver()`, the default, the name of the scale is taken from the first mapping used for that aesthetic. If `NULL`, the legend title will be omitted.

**breaks** One of:

- `NULL` for no breaks
- `waiver()` for the default breaks computed by the transformation object
- A character vector of breaks
- A function that takes the limits as input and returns breaks as output

**labels** One of:

- `NULL` for no labels
- `waiver()` for the default labels computed by the transformation object
- A character vector giving labels (must be same length as breaks)
- A function that takes the breaks as input and returns labels as output

**limits** A character vector that defines possible values of the scale and their order.



**expand** Vector of range expansion constants used to add some padding around the data, to ensure that they are placed some distance away from the axes. Use the convenience function `expand_scale()` to generate the values for the `expand` argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.

**na.translate** Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.

**na.value** If `na.translate = TRUE`, what value aesthetic value should missing be displayed as? Does not apply to position scales where NA is always placed at the far right.

**drop** Should unused factor levels be omitted from the scale? The default, TRUE, uses the levels that appear in the data; FALSE uses all the levels in the factor.

**guide** A function used to create a guide or its name. See `guides()` for more info.

**position** The position of the axis. "left" or "right" for vertical scales, "top" or "bottom" for horizontal scales

**super** The super class to use for the constructed scale

## Examples

```
library(scales)
show_col(kimPossible_pal()(5))

library(ggplot2)
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_kimPossible()

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_colour_kimPossible()

ggplot(mpg, aes(displ)) +
  geom_histogram(aes(fill = class), col = "black", size = 0.1) +
  scale_fill_kimPossible()
```

---

paintBikiniBottom      *Add SpongeBob background*

---

## Description

Add SpongeBob background

**Usage**

```
paintBikiniBottom(plot, width = 800, height = 500,
  output.file = NULL, background = "background", ...)
```

**Arguments**

plot	the ggplot object you want to Spongobify!
width	width, Default: 800
height	height, Default: 500
output.file	File path to save image, Default: NULL
background	"background" or "floral", Default: "background"
...	Other options, see <code>?magick::image_graph()</code>

**Details**

Adapted from `ggpomological`'s `paint_pomological()` function!

**Value**

Your plot with a Spongebob themed background!

---

parksAndRec_pal	<i>Parks &amp; Recreation palette</i>
-----------------	---------------------------------------

---

**Description**

Parks & Recreation palette

**Usage**

```
parksAndRec_pal(n, type = c("discrete", "continuous"), reverse = FALSE)
scale_color_parksAndRec(n, type = "discrete", reverse = FALSE, ...)
scale_colour_parksAndRec(n, type = "discrete", reverse = FALSE, ...)
scale_fill_parksAndRec(n, type = "discrete", reverse = FALSE, ...)
```

**Arguments**

n	number of colors
type	discrete or continuous
reverse	reverse order, Default: FALSE
...	Arguments passed on to <code>ggplot2::discrete_scale</code>

- aesthetics** The names of the aesthetics that this scale works with
- scale\_name** The name of the scale
- palette** A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take.
- name** The name of the scale. Used as the axis or legend title. If `waiver()`, the default, the name of the scale is taken from the first mapping used for that aesthetic. If `NULL`, the legend title will be omitted.
- breaks** One of:
- `NULL` for no breaks
  - `waiver()` for the default breaks computed by the transformation object
  - A character vector of breaks
  - A function that takes the limits as input and returns breaks as output
- labels** One of:
- `NULL` for no labels
  - `waiver()` for the default labels computed by the transformation object
  - A character vector giving labels (must be same length as breaks)
  - A function that takes the breaks as input and returns labels as output
- limits** A character vector that defines possible values of the scale and their order.
- expand** Vector of range expansion constants used to add some padding around the data, to ensure that they are placed some distance away from the axes. Use the convenience function `expand_scale()` to generate the values for the `expand` argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.
- na.translate** Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.
- na.value** If `na.translate = TRUE`, what value aesthetic value should missing be displayed as? Does not apply to position scales where `NA` is always placed at the far right.
- drop** Should unused factor levels be omitted from the scale? The default, `TRUE`, uses the levels that appear in the data; `FALSE` uses all the levels in the factor.
- guide** A function used to create a guide or its name. See `guides()` for more info.
- position** The position of the axis. "left" or "right" for vertical scales, "top" or "bottom" for horizontal scales
- super** The super class to use for the constructed scale

## Examples

```
library(scales)
show_col(parksAndRec_pal()(5))

library(ggplot2)
ggplot(airquality, aes(x = Day, y = Temp,
```

```

group = as.factor(Month), color = as.factor(Month))) +
geom_point(size = 2.5) +
scale_color_parksAndRec()

ggplot(airquality, aes(x = Day, y = Temp,
group = as.factor(Month), color = as.factor(Month))) +
geom_point(size = 2.5) +
scale_colour_parksAndRec()

ggplot(mpg, aes(displ)) +
geom_histogram(aes(fill = class), col = "black", size = 0.1) +
scale_fill_parksAndRec()

```

---

rickAndMorty\_pal      *Rick & Morty color palette*

---

## Description

Rick & Morty color palette

## Usage

```

rickAndMorty_pal(n, type = c("discrete", "continuous"),
reverse = FALSE)

scale_color_rickAndMorty(n, type = "discrete", reverse = FALSE, ...)

scale_colour_rickAndMorty(n, type = "discrete", reverse = FALSE, ...)

scale_fill_rickAndMorty(n, type = "discrete", reverse = FALSE, ...)

```

## Arguments

n	number of colors
type	discrete or continuous
reverse	reverse order, Default: FALSE
...	Arguments passed on to <code>ggplot2::discrete_scale</code>

**aesthetics** The names of the aesthetics that this scale works with

**scale\_name** The name of the scale

**palette** A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take.

**name** The name of the scale. Used as the axis or legend title. If `waiver()`, the default, the name of the scale is taken from the first mapping used for that aesthetic. If `NULL`, the legend title will be omitted.

**breaks** One of:

- `NULL` for no breaks

- `waiver()` for the default breaks computed by the transformation object
- A character vector of breaks
- A function that takes the limits as input and returns breaks as output

**labels** One of:

- NULL for no labels
- `waiver()` for the default labels computed by the transformation object
- A character vector giving labels (must be same length as breaks)
- A function that takes the breaks as input and returns labels as output

**limits** A character vector that defines possible values of the scale and their order.

**expand** Vector of range expansion constants used to add some padding around the data, to ensure that they are placed some distance away from the axes. Use the convenience function `expand_scale()` to generate the values for the `expand` argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.

**na.translate** Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.

**na.value** If `na.translate = TRUE`, what value aesthetic value should missing be displayed as? Does not apply to position scales where NA is always placed at the far right.

**drop** Should unused factor levels be omitted from the scale? The default, TRUE, uses the levels that appear in the data; FALSE uses all the levels in the factor.

**guide** A function used to create a guide or its name. See `guides()` for more info.

**position** The position of the axis. "left" or "right" for vertical scales, "top" or "bottom" for horizontal scales

**super** The super class to use for the constructed scale

## Examples

```
library(scales)
show_col(rickAndMorty_pal()(5))
```

```
library(ggplot2)
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_rickAndMorty()
```

```
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_colour_rickAndMorty()
```

```
ggplot(mpg, aes(displ)) +
  geom_histogram(aes(fill = class), col = "black", size = 0.1) +
  scale_fill_rickAndMorty()
```

---

simpsons\_pal

*The Simpsons palette*


---

### Description

The Simpsons palette

### Usage

```
simpsons_pal(n, type = c("discrete", "continuous"), reverse = FALSE)
```

```
scale_color_simpsons(n, type = "discrete", reverse = FALSE, ...)
```

```
scale_colour_simpsons(n, type = "discrete", reverse = FALSE, ...)
```

```
scale_fill_simpsons(n, type = "discrete", reverse = FALSE, ...)
```

### Arguments

**n** number of colors

**type** discrete or continuous

**reverse** reverse order, Default: FALSE

**...** Arguments passed on to `ggplot2::discrete_scale`

**aesthetics** The names of the aesthetics that this scale works with

**scale\_name** The name of the scale

**palette** A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take.

**name** The name of the scale. Used as the axis or legend title. If `waiver()`, the default, the name of the scale is taken from the first mapping used for that aesthetic. If `NULL`, the legend title will be omitted.

**breaks** One of:

- `NULL` for no breaks
- `waiver()` for the default breaks computed by the transformation object
- A character vector of breaks
- A function that takes the limits as input and returns breaks as output

**labels** One of:

- `NULL` for no labels
- `waiver()` for the default labels computed by the transformation object
- A character vector giving labels (must be same length as breaks)
- A function that takes the breaks as input and returns labels as output

**limits** A character vector that defines possible values of the scale and their order.

**expand** Vector of range expansion constants used to add some padding around the data, to ensure that they are placed some distance away from the axes. Use the convenience function `expand_scale()` to generate the values for the `expand` argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.

**na.translate** Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.

**na.value** If `na.translate = TRUE`, what value aesthetic value should missing be displayed as? Does not apply to position scales where NA is always placed at the far right.

**drop** Should unused factor levels be omitted from the scale? The default, TRUE, uses the levels that appear in the data; FALSE uses all the levels in the factor.

**guide** A function used to create a guide or its name. See `guides()` for more info.

**position** The position of the axis. "left" or "right" for vertical scales, "top" or "bottom" for horizontal scales

**super** The super class to use for the constructed scale

## Examples

```
library(scales)
show_col(simpsons_pal()(5))
```

```
library(ggplot2)
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_colorsimpsons()
```

```
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_colorsimpsons()
```

```
ggplot(mpg, aes(displ)) +
  geom_histogram(aes(fill = class), col = "black", size = 0.1) +
  scale_fillsimpsons()
```

---

spongeBob\_pal

*Spongebob Squarepants palette*

---

## Description

Spongebob Squarepants palette

**Usage**

```
spongeBob_pal(n, type = c("discrete", "continuous"), reverse = FALSE)

scale_color_spongeBob(n, type = "discrete", reverse = FALSE, ...)

scale_colour_spongeBob(n, type = "discrete", reverse = FALSE, ...)

scale_fill_spongeBob(n, type = "discrete", reverse = FALSE, ...)
```

**Arguments**

n	number of colors
type	discrete or continuous
reverse	reverse order, Default: FALSE
...	Arguments passed on to <code>ggplot2::discrete_scale</code>

**aesthetics** The names of the aesthetics that this scale works with

**scale\_name** The name of the scale

**palette** A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take.

**name** The name of the scale. Used as the axis or legend title. If `waiver()`, the default, the name of the scale is taken from the first mapping used for that aesthetic. If `NULL`, the legend title will be omitted.

**breaks** One of:

- `NULL` for no breaks
- `waiver()` for the default breaks computed by the transformation object
- A character vector of breaks
- A function that takes the limits as input and returns breaks as output

**labels** One of:

- `NULL` for no labels
- `waiver()` for the default labels computed by the transformation object
- A character vector giving labels (must be same length as breaks)
- A function that takes the breaks as input and returns labels as output

**limits** A character vector that defines possible values of the scale and their order.

**expand** Vector of range expansion constants used to add some padding around the data, to ensure that they are placed some distance away from the axes. Use the convenience function `expand_scale()` to generate the values for the `expand` argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.

**na.translate** Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.



**na.value** If `na.translate = TRUE`, what value aesthetic value should missing be displayed as? Does not apply to position scales where NA is always placed at the far right.

**drop** Should unused factor levels be omitted from the scale? The default, `TRUE`, uses the levels that appear in the data; `FALSE` uses all the levels in the factor.

**guide** A function used to create a guide or its name. See `guides()` for more info.

**position** The position of the axis. "left" or "right" for vertical scales, "top" or "bottom" for horizontal scales

**super** The super class to use for the constructed scale

## Examples

```
library(scales)
show_col(spongeBob_pal()(5))

library(ggplot2)
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_spongeBob()

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_spongeBob()

ggplot(mpg, aes(displ)) +
  geom_histogram(aes(fill = class), col = "black", size = 0.1) +
  scale_fill_spongeBob()
```

---

sponge\_images

*sponge\_images*

---

## Description

find SpongeBob background images

## Usage

```
sponge_images(which = c("background", "floral"))
```

## Arguments

which           PARAM\_DESCRIPTION, Default: `c("background", "floral")`

---

theme_brooklyn99	<i>theme_brooklyn99</i>
------------------	-------------------------

---

### Description

Brooklyn Nine-Nine theme, Recommended font: "Roboto Condensed" (title), "Calibri Light" (other text)

### Usage

```
theme_brooklyn99(text.font = NULL, title.font = NULL,
  legend.font = NULL, title.size = 18, text.size = 14,
  subtitle.size = 12, axis.title.size = 14, axis.text.size = 12,
  legend.title.size = 10, legend.text.size = 9,
  title.color = "#F9FEFF", subtitle.color = "#F9FEFF",
  text.color = "#F9FEFF", axis.title.color = "#F9FEFF",
  axis.text.color = "#F9FEFF", legend.title.color = "#F9FEFF",
  legend.text.color = "#F9FEFF", legend.position = "bottom",
  ticks = FALSE)
```

### Arguments

text.font	text font, Default: NULL
title.font	title font, Default: NULL
legend.font	legend font, Default: NULL
title.size	title font size, Default: 18
text.size	text font size, Default: 14
subtitle.size	subtitle font size, Default: 12
axis.title.size	axis title font size, Default: 14
axis.text.size	axis text font size, Default: 12
legend.title.size	legend title font size, Default: 10
legend.text.size	legend text font size, Default: 9
title.color	title color, Default: "F9FEFF"
subtitle.color	subtitle.color, Default: "F9FEFF"
text.color	text color, Default: "F9FEFF"
axis.title.color	axis title color, Default: "F9FEFF"
axis.text.color	axis text color, Default: "F9FEFF"
legend.title.color	legend title color, Default: "F9FEFF"

legend.text.color	legend text color, Default: "F9FEFF"
legend.position	legend position, Default: "bottom"
ticks	add axis ticks, Default: FALSE

### Details

Actual font: Variants of 'Univers'

### See Also

[ggplot2::theme]

### Examples

```
library(ggplot2)

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_brooklyn99() +
  theme_brooklyn99()
```

---

theme_hildaDay	<i>theme_hildaDay</i>
----------------	-----------------------

---

### Description

Hilda Day theme

### Usage

```
theme_hildaDay(text.font = "Chelsea Market",
  title.font = "Chelsea Market", legend.font = "Chelsea Market",
  title.size = 18, text.size = 14, subtitle.size = 12,
  axis.title.size = 14, axis.text.size = 12, legend.title.size = 10,
  legend.text.size = 9, title.color = "#659794",
  subtitle.color = "#659794", text.color = "#659794",
  axis.title.color = "#659794", axis.text.color = "#93a1a1",
  legend.title.color = "#659794", legend.text.color = "#93a1a1",
  legend.position = "bottom", ticks = FALSE)
```

**Arguments**

<code>text.font</code>	text font, Default: "Chelsea Market"
<code>title.font</code>	title font, Default: "Chelsea Market"
<code>legend.font</code>	legend font, Default: "Chelsea Market"
<code>title.size</code>	title font size, Default: 18
<code>text.size</code>	text font size, Default: 14
<code>subtitle.size</code>	subtitle font size, Default: 12
<code>axis.title.size</code>	axis title font size, Default: 14
<code>axis.text.size</code>	axis text font size, Default: 12
<code>legend.title.size</code>	legend title font size, Default: 10
<code>legend.text.size</code>	legend text font size, Default: 9
<code>title.color</code>	title color, Default: '#F9FEFF'
<code>subtitle.color</code>	subtitle color, Default: '#F9FEFF'
<code>text.color</code>	text color, Default: '#F9FEFF'
<code>axis.title.color</code>	axis title color, Default: '#F9FEFF'
<code>axis.text.color</code>	axis text color, Default: '#F9FEFF'
<code>legend.title.color</code>	legend title color, Default: '#F9FEFF'
<code>legend.text.color</code>	legend text color, Default: '#F9FEFF'
<code>legend.position</code>	legend position, Default: 'bottom'
<code>ticks</code>	add axis ticks, Default: FALSE

**Examples**

```
library(ggplot2)

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_hilda(palette = "Day") +
  theme_hildaDay(text.font = "Times", title.font = "Times",
    legend.font = "Times")
```

---

theme_hildaDusk	<i>theme_hildaDusk</i>
-----------------	------------------------

---

## Description

Hilda theme

## Usage

```
theme_hildaDusk(text.font = "Chelsea Market",
  title.font = "Chelsea Market", legend.font = "Chelsea Market",
  title.size = 18, text.size = 14, subtitle.size = 12,
  axis.title.size = 14, axis.text.size = 12, legend.title.size = 10,
  legend.text.size = 9, title.color = "#F9FEFF",
  subtitle.color = "#F9FEFF", text.color = "#F9FEFF",
  axis.title.color = "#F9FEFF", axis.text.color = "#F9FEFF",
  legend.title.color = "#F9FEFF", legend.text.color = "#F9FEFF",
  legend.position = "bottom", ticks = FALSE)
```

## Arguments

text.font	text font, Default: "Chelsea Market"
title.font	title font, Default: "Chelsea Market"
legend.font	legend font, Default: "Chelsea Market"
title.size	title font size, Default: 18
text.size	text font size, Default: 14
subtitle.size	subtitle font size, Default: 12
axis.title.size	axis title font size, Default: 14
axis.text.size	axis text font size, Default: 12
legend.title.size	legend title font size, Default: 10
legend.text.size	legend text font size, Default: 9
title.color	title color, Default: '#F9FEFF'
subtitle.color	subtitle color, Default: '#F9FEFF'
text.color	text color, Default: '#F9FEFF'
axis.title.color	axis title color, Default: '#F9FEFF'
axis.text.color	axis text color, Default: '#F9FEFF'
legend.title.color	legend title color, Default: '#F9FEFF'

```

legend.text.color      legend text color, Default: '#F9FEFF'
legend.position        legend position, Default: 'bottom'
ticks                  add axis ticks, Default: FALSE

```

### Examples

```

library(ggplot2)

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_hilda(palette = "Dusk") +
  theme_hildaDusk(text.font = "Times", title.font = "Times",
    legend.font = "Times")

```

---

theme_hildaNight	<i>theme_hildaNight</i>
------------------	-------------------------

---

### Description

Hilda theme

### Usage

```

theme_hildaNight(text.font = "Chelsea Market",
  title.font = "Chelsea Market", legend.font = "Chelsea Market",
  title.size = 18, text.size = 14, subtitle.size = 12,
  axis.title.size = 14, axis.text.size = 12, legend.title.size = 10,
  legend.text.size = 9, title.color = "#F9FEFF",
  subtitle.color = "#F9FEFF", text.color = "#F9FEFF",
  axis.title.color = "#F9FEFF", axis.text.color = "#F9FEFF",
  legend.title.color = "#F9FEFF", legend.text.color = "#F9FEFF",
  legend.position = "bottom", ticks = FALSE)

```

### Arguments

```

text.font      text font, Default: "Chelsea Market"
title.font     title font, Default: "Chelsea Market"
legend.font    legend font, Default: "Chelsea Market"
title.size     title font size, Default: 18
text.size      text font size, Default: 14
subtitle.size  subtitle font size, Default: 12
axis.title.size axis title font size, Default: 14

```

```

axis.text.size axis text font size, Default: 12
legend.title.size
                legend title font size, Default: 10
legend.text.size
                legend text font size, Default: 9
title.color     title color, Default: '#F9FEFF'
subtitle.color  subtitle color, Default: '#F9FEFF'
text.color      text color, Default: '#F9FEFF'
axis.title.color
                axis title color, Default: '#F9FEFF'
axis.text.color
                axis text color, Default: '#F9FEFF'
legend.title.color
                legend title color, Default: '#F9FEFF'
legend.text.color
                legend text color, Default: '#F9FEFF'
legend.position
                legend position, Default: 'bottom'
ticks           add axis ticks, Default: FALSE

```

### Examples

```

library(ggplot2)

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_hilda(palette = "Night") +
  theme_hildaNight(text.font = "Times", title.font = "Times",
    legend.font = "Times")

```

---

```

theme_parksAndRec      theme_parksAndRec

```

---

### Description

Parks & Recreation theme, Recommended font: "Titillium Web"

### Usage

```

theme_parksAndRec(text.font = NULL, title.font = NULL,
  legend.font = NULL, title.size = 20, text.size = 16,
  subtitle.size = 14, axis.title.size = 14, axis.text.size = 12,
  legend.title.size = 14, legend.text.size = 12, title.color = NULL,
  subtitle.color = NULL, text.color = NULL,
  axis.title.color = "black", axis.text.color = "black",
  legend.title.color = NULL, legend.text.color = NULL,
  legend.position = "bottom", ticks = FALSE)

```

**Arguments**

<code>text.font</code>	text font, Default: NULL
<code>title.font</code>	title font, Default: NULL
<code>legend.font</code>	legend font, Default: NULL
<code>title.size</code>	title font size, Default: 20
<code>text.size</code>	text font size, Default: 16
<code>subtitle.size</code>	subtitle font size, Default: 14
<code>axis.title.size</code>	axis title font size, Default: 14
<code>axis.text.size</code>	axis text font size, Default: 12
<code>legend.title.size</code>	legend title font size, Default: 14
<code>legend.text.size</code>	legend text font size, Default: 12
<code>title.color</code>	title color, Default: NULL
<code>subtitle.color</code>	subtitle color, Default: NULL
<code>text.color</code>	text color, Default: NULL
<code>axis.title.color</code>	axis title color, Default: NULL
<code>axis.text.color</code>	axis text color, Default: NULL
<code>legend.title.color</code>	legend title color, Default: NULL
<code>legend.text.color</code>	legend text color, Default: NULL
<code>legend.position</code>	legend position, Default: "bottom"
<code>ticks</code>	add axis ticks, Default: FALSE

**Details**

Actual font: 'Champion HTF-Heavyweight'

**See Also**

[[ggplot2::theme](#)]

**Examples**

```
library(ggplot2)

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_parksAndRec() +
  theme_parksAndRec()
```



---

theme\_parksAndRecLight  
*theme\_parksAndRecLight*

---

### Description

Parks & Recreation light theme, Recommended font: "Titillium Web"

### Usage

```
theme_parksAndRecLight(text.font = NULL, title.font = NULL,  
  legend.font = NULL, title.size = 20, text.size = 16,  
  subtitle.size = 14, axis.title.size = 14, axis.text.size = 12,  
  legend.title.size = 14, legend.text.size = 12,  
  title.color = "grey20", subtitle.color = "grey20",  
  text.color = "grey20", axis.title.color = "grey20",  
  axis.text.color = "grey20", legend.title.color = "grey20",  
  legend.text.color = "grey20", legend.position = "bottom",  
  ticks = FALSE)
```

### Arguments

text.font	text font, Default: NULL
title.font	title font, Default: NULL
legend.font	legend font, Default: NULL
title.size	title font size, Default: 20
text.size	text font size, Default: 16
subtitle.size	subtitle font size, Default: 14
axis.title.size	axis title font size, Default: 14
axis.text.size	axis text font size, Default: 12
legend.title.size	legend title font size, Default: 14
legend.text.size	legend text font size, Default: 12
title.color	title color, Default: "grey20"
subtitle.color	subtitle color, Default: "grey20"
text.color	text color, Default: "grey20"
axis.title.color	axis title color, Default: "grey20"
axis.text.color	axis text color, Default: "grey20"

```

legend.title.color      legend title color, Default: "grey20"
legend.text.color      legend text color, Default: "grey20"
legend.position        legend position, Default: "bottom"
ticks                  add axis ticks, Default: FALSE

```

**Details**

Actual font: 'Champion HTF-Heavyweight'

**See Also**

[ggplot2::theme]

**Examples**

```

library(ggplot2)

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_parksAndRec() +
  theme_parksAndRecLight()

```

---

```
theme_parksAndRec_light
```

```
theme_parksAndRec_light
```

---

**Description**

Parks & Recreation light theme, Recommended font: "Titillium Web"

**Usage**

```

theme_parksAndRec_light(text.font = NULL, title.font = NULL,
  legend.font = NULL, title.size = 20, text.size = 16,
  subtitle.size = 14, axis.title.size = 14, axis.text.size = 12,
  legend.title.size = 14, legend.text.size = 12,
  title.color = "grey20", subtitle.color = "grey20",
  text.color = "grey20", axis.title.color = "grey20",
  axis.text.color = "grey20", legend.title.color = "grey20",
  legend.text.color = "grey20", legend.position = "bottom",
  ticks = FALSE)

```

**Arguments**

text.font	text font, Default: NULL
title.font	title font, Default: NULL
legend.font	legend font, Default: NULL
title.size	title font size, Default: 20
text.size	text font size, Default: 16
subtitle.size	subtitle font size, Default: 14
axis.title.size	axis title font size, Default: 14
axis.text.size	axis text font size, Default: 12
legend.title.size	legend title font size, Default: 14
legend.text.size	legend text font size, Default: 12
title.color	title color, Default: "grey20"
subtitle.color	subtitle color, Default: "grey20"
text.color	text color, Default: "grey20"
axis.title.color	axis title color, Default: "grey20"
axis.text.color	axis text color, Default: "grey20"
legend.title.color	legend title color, Default: "grey20"
legend.text.color	legend text color, Default: "grey20"
legend.position	legend position, Default: "bottom"
ticks	add axis ticks, Default: FALSE

**Details**

Actual font: 'Champion HTF-Heavyweight' This function has been deprecated in favor of 'theme\_parksAndRecLight' to follow the naming conventions of the package.

**See Also**

[ggplot2::theme]

---

theme\_rickAndMorty      *theme\_rickAndMorty*

---

## Description

Rick & Morty theme, Recommended font: "Get Schwifty"

## Usage

```
theme_rickAndMorty(text.font = NULL, title.font = NULL,
  legend.font = NULL, title.size = 20, text.size = 12,
  subtitle.size = 14, axis.title.size = 14, axis.text.size = 10,
  legend.title.size = 10, legend.text.size = 9, title.color = NULL,
  subtitle.color = NULL, text.color = NULL, axis.title.color = NULL,
  axis.text.color = "black", legend.title.color = NULL,
  legend.text.color = NULL, legend.position = "bottom",
  ticks = FALSE)
```

## Arguments

text.font	text font, Default: NULL
title.font	title font, Default: NULL
legend.font	legend font, Default: NULL
title.size	title size, Default: 20
text.size	text font size, Default: 12
subtitle.size	subtitle font size, Default: 14
axis.title.size	axis title font size, Default: 14
axis.text.size	axis text font size, Default: 10
legend.title.size	legend title font size, Default: 10
legend.text.size	legend text font size, Default: 9
title.color	title color, Default: NULL
subtitle.color	subtitle.color, Default: NULL
text.color	text color, Default: NULL
axis.title.color	axis title color, Default: NULL
axis.text.color	axis text color, Default: "black"
legend.title.color	legend title color, Default: NULL

legend.text.color	legend text color, Default: NULL
legend.position	legend position, Default: "bottom"
ticks	add axis ticks, Default: FALSE

### Details

Actual font is based on Justin Roiland's handwriting!

### See Also

[ggplot2::theme]

### Examples

```
library(ggplot2)

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_rickAndMorty() +
  theme_rickAndMorty()
```

---

theme_simpsons	<i>theme_simpsons</i>
----------------	-----------------------

---

### Description

The Simpsons theme, Recommended font: "Akbar"

### Usage

```
theme_simpsons(text.font = NULL, title.font = NULL,
  legend.font = NULL, title.size = 18, text.size = 14,
  subtitle.size = 12, axis.title.size = 14, axis.text.size = 10,
  legend.title.size = 10, legend.text.size = 9,
  title.color = "#FFD235", subtitle.color = "#fee8c8",
  text.color = "#fee8c8", axis.title.color = "#fee8c8",
  axis.text.color = "#fee8c8", legend.title.color = "#ffffff",
  legend.text.color = "#ffffff", legend.position = "bottom",
  ticks = FALSE)
```

**Arguments**

text.font	text font, Default: NULL
title.font	title font, Default: NULL
legend.font	legend font, Default: NULL
title.size	title font size, Default: 18
text.size	text font size, Default: 14
subtitle.size	subtitle font size, Default: 12
axis.title.size	axis title font size, Default: 14
axis.text.size	axis text font size, Default: 10
legend.title.size	legend title font size, Default: 10
legend.text.size	legend text font size, Default: 9
title.color	title color, Default: "#FFD235"
subtitle.color	subtitle color, Default: "#fee8c8"
text.color	text color, Default: "#fee8c8"
axis.title.color	axis title color, Default: "#fee8c8"
axis.text.color	axis text color, Default: "#fee8c8"
legend.title.color	legend title color, Default: "#ffffff"
legend.text.color	legend text color, Default: "#ffffff"
legend.position	legend position, Default: "bottom"
ticks	add axis ticks, Default: FALSE

**Details**

In part inspired by '@nathancunn's blog posts on The Simpsons!

**See Also**

[ggplot2::theme]

**Examples**

```
library(ggplot2)

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_simpsons() +
  theme_simpsons()
```

---

theme_spongeBob	<i>theme_spongeBob</i>
-----------------	------------------------

---

## Description

Spongebob Squarepants theme, Recommended font: "Some Time Later"

## Usage

```
theme_spongeBob(text.font = NULL, title.font = NULL,
  legend.font = NULL, title.size = 18, text.size = 12,
  subtitle.size = 12, axis.title.size = 14, axis.text.size = 12,
  legend.title.size = 10, legend.text.size = 9,
  title.color = "#F9FEFF", subtitle.color = "#F9FEFF",
  text.color = "#F9FEFF", axis.title.color = "#F9FEFF",
  axis.text.color = "#F9FEFF", legend.title.color = "#F9FEFF",
  legend.text.color = "#F9FEFF", legend.position = "bottom",
  ticks = FALSE)
```

## Arguments

text.font	text font, Default: NULL
title.font	title font, Default: NULL
legend.font	legend font, Default: NULL
title.size	size of title, Default: 18
text.size	text font size, Default: 12
subtitle.size	subtitle font size, Default: 12
axis.title.size	axis title font size, Default: 14
axis.text.size	axis text font size, Default: 12
legend.title.size	legend title font size, Default: 10
legend.text.size	legend text font size, Default: 9
title.color	title color, Default: "F9FEFF"
subtitle.color	subtitle color, Default: "F9FEFF"
text.color	text color, Default: "F9FEFF"
axis.title.color	axis title color, Default: "F9FEFF"
axis.text.color	axis text color, Default: "F9FEFF"
legend.title.color	legend title color, Default: "F9FEFF"

legend.text.color	legend text color, Default: "F9FEFF"
legend.position	legend position, Default: "bottom"
ticks	add axis ticks, Default: FALSE

### Details

Spongebobify your plots even more by combining with `'paintBikiniBottom()'`!

### See Also

[[tvthemes::paintBikiniBottom](#)]

### Examples

```
library(ggplot2)

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_spongeBob() +
  theme_spongeBob()
```

---

theme\_theLastAirbender

*theme\_theLastAirbender*

---

### Description

Avatar: The Last Airbender theme, Recommended font: "Slayer"

### Usage

```
theme_theLastAirbender(text.font = NULL, title.font = NULL,
  legend.font = NULL, title.size = 14, text.size = 10,
  subtitle.size = 12, axis.title.size = 10, axis.text.size = 8,
  legend.title.size = 10, legend.text.size = 8, title.color = NULL,
  subtitle.color = "grey20", text.color = NULL,
  axis.title.color = "grey20", axis.text.color = "grey20",
  legend.title.color = "grey20", legend.text.color = "grey20",
  legend.position = "bottom", ticks = FALSE)
```



**Arguments**

text.font	text font, Default: NULL
title.font	title font, Default: NULL
legend.font	legend font, Default: NULL
title.size	title font size, Default: 14
text.size	text font size, Default: 10
subtitle.size	subtitle font size, Default: 12
axis.title.size	axis title font size, Default: 10
axis.text.size	axis text font size, Default: 8
legend.title.size	legend title font size, Default: 10
legend.text.size	legend text font size, Default: 8
title.color	title color, Default: NULL
subtitle.color	subtitle color, Default: "grey20"
text.color	text color, Default: NULL
axis.title.color	axis title color, Default: "grey20"
axis.text.color	axis text color, Default: "grey20"
legend.title.color	legend title color, Default: "grey20"
legend.text.color	legend text color, Default: "grey20"
legend.position	legend position, Default: "bottom"
ticks	add axis ticks, Default: FALSE

**See Also**

[ggplot2::theme]

**Examples**

```
library(ggplot2)

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_avatarTLA() +
  theme_theLastAirbender()
```

westeros\_pal

*Great Houses of Westeros palette***Description**

Houses Stark, Lannister, Tyrell, Targaryen, Tully, Greyjoy, Manderly, Martell, Stannis Baratheon, & Arryn

**Usage**

```
westeros_pal(palette = "Stark", n, type = c("discrete", "continuous"),
  reverse = FALSE)
```

```
scale_color_westeros(palette = "Stark", n, type = "discrete",
  reverse = FALSE, ...)
```

```
scale_colour_westeros(palette = "Stark", n, type = "discrete",
  reverse = FALSE, ...)
```

```
scale_fill_westeros(palette = "Stark", n, type = "discrete",
  reverse = FALSE, ...)
```

**Arguments**

**palette** name of palette, Default: "Stark"

**n** number of colors

**type** discrete or continuous

**reverse** reverse order, Default: FALSE

**...** Arguments passed on to `ggplot2::discrete_scale`

**aesthetics** The names of the aesthetics that this scale works with

**scale\_name** The name of the scale

**palette** A palette function that when called with a single integer argument (the number of levels in the scale) returns the values that they should take.

**name** The name of the scale. Used as the axis or legend title. If `waiver()`, the default, the name of the scale is taken from the first mapping used for that aesthetic. If `NULL`, the legend title will be omitted.

**breaks** One of:

- `NULL` for no breaks
- `waiver()` for the default breaks computed by the transformation object
- A character vector of breaks
- A function that takes the limits as input and returns breaks as output

**labels** One of:

- `NULL` for no labels
- `waiver()` for the default labels computed by the transformation object

- A character vector giving labels (must be same length as breaks)
- A function that takes the breaks as input and returns labels as output

**limits** A character vector that defines possible values of the scale and their order.

**expand** Vector of range expansion constants used to add some padding around the data, to ensure that they are placed some distance away from the axes. Use the convenience function `expand_scale()` to generate the values for the `expand` argument. The defaults are to expand the scale by 5% on each side for continuous variables, and by 0.6 units on each side for discrete variables.

**na.translate** Unlike continuous scales, discrete scales can easily show missing values, and do so by default. If you want to remove missing values from a discrete scale, specify `na.translate = FALSE`.

**na.value** If `na.translate = TRUE`, what value aesthetic value should missing be displayed as? Does not apply to position scales where NA is always placed at the far right.

**drop** Should unused factor levels be omitted from the scale? The default, TRUE, uses the levels that appear in the data; FALSE uses all the levels in the factor.

**guide** A function used to create a guide or its name. See `guides()` for more info.

**position** The position of the axis. "left" or "right" for vertical scales, "top" or "bottom" for horizontal scales

**super** The super class to use for the constructed scale

## Examples

```
library(scales)
show_col(westeros_pal(palette = "Stark")(5))
show_col(westeros_pal(palette = "Stannis")(5))

library(ggplot2)
ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_westeros(palette = "Stark")

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_color_westeros(palette = "Stannis")

ggplot(airquality, aes(x = Day, y = Temp,
  group = as.factor(Month), color = as.factor(Month))) +
  geom_point(size = 2.5) +
  scale_colour_westeros(palette = "Stannis")

ggplot(mpg, aes(displ)) +
  geom_histogram(aes(fill = class), col = "black", size = 0.1) +
  scale_fill_westeros(palette = "Stannis")
```

# Index

attackOnTitan\_pal, 2  
avatarTLA\_pal, 4

bigHero6\_pal, 6  
brooklyn99\_pal, 8

expand\_scale(), 3, 5, 7, 9, 11, 17, 19, 21, 23, 24, 43

font\_import, 12–15

guides(), 3, 5, 7, 9, 11, 17, 19, 21, 23, 25, 43

hilda\_pal, 10

import\_ChelseaMarket, 12  
import\_cinzel, 12  
import\_rickAndMorty, 13  
import\_roboto\_condensed, 13  
import\_simpsons, 14  
import\_spongeBob, 14  
import\_theLastAirbender, 15  
import\_titillium\_web, 15

kimPossible\_pal, 16

paintBikiniBottom, 17  
parksAndRec\_pal, 18

rickAndMorty\_pal, 20

scale\_color\_attackOnTitan  
(attackOnTitan\_pal), 2  
scale\_color\_avatarTLA (avatarTLA\_pal), 4  
scale\_color\_bigHero6 (bigHero6\_pal), 6  
scale\_color\_brooklyn99  
(brooklyn99\_pal), 8  
scale\_color\_hilda (hilda\_pal), 10  
scale\_color\_kimPossible  
(kimPossible\_pal), 16  
scale\_color\_parksAndRec  
(parksAndRec\_pal), 18

scale\_color\_rickAndMorty  
(rickAndMorty\_pal), 20  
scale\_color\_simpsons (simpsons\_pal), 22  
scale\_color\_spongeBob (spongeBob\_pal), 23

scale\_color\_westeros (westeros\_pal), 42  
scale\_colour\_attackOnTitan  
(attackOnTitan\_pal), 2  
scale\_colour\_avatarTLA (avatarTLA\_pal), 4  
scale\_colour\_bigHero6 (bigHero6\_pal), 6  
scale\_colour\_brooklyn99  
(brooklyn99\_pal), 8  
scale\_colour\_hilda (hilda\_pal), 10  
scale\_colour\_kimPossible  
(kimPossible\_pal), 16  
scale\_colour\_parksAndRec  
(parksAndRec\_pal), 18  
scale\_colour\_rickAndMorty  
(rickAndMorty\_pal), 20  
scale\_colour\_simpsons (simpsons\_pal), 22  
scale\_colour\_spongeBob (spongeBob\_pal), 23

scale\_colour\_westeros (westeros\_pal), 42  
scale\_fill\_attackOnTitan  
(attackOnTitan\_pal), 2  
scale\_fill\_avatarTLA (avatarTLA\_pal), 4  
scale\_fill\_bigHero6 (bigHero6\_pal), 6  
scale\_fill\_brooklyn99 (brooklyn99\_pal), 8  
scale\_fill\_hilda (hilda\_pal), 10  
scale\_fill\_kimPossible  
(kimPossible\_pal), 16  
scale\_fill\_parksAndRec  
(parksAndRec\_pal), 18  
scale\_fill\_rickAndMorty  
(rickAndMorty\_pal), 20  
scale\_fill\_simpsons (simpsons\_pal), 22  
scale\_fill\_spongeBob (spongeBob\_pal), 23

scale\_fill\_westeros (westeros\_pal), [42](#)  
simpsons\_pal, [22](#)  
sponge\_images, [25](#)  
spongeBob\_pal, [23](#)

theme\_brooklyn99, [26](#)  
theme\_hildaDay, [27](#)  
theme\_hildaDusk, [29](#)  
theme\_hildaNight, [30](#)  
theme\_parksAndRec, [31](#)  
theme\_parksAndRec\_light, [34](#)  
theme\_parksAndRecLight, [33](#)  
theme\_rickAndMorty, [36](#)  
theme\_simpsons, [37](#)  
theme\_spongeBob, [39](#)  
theme\_theLastAirbender, [40](#)

westeros\_pal, [42](#)