

# Package ‘ggblanket’

June 14, 2024

**Title** Simplify 'ggplot2' Visualisation

**Version** 9.1.1

**Description** Simplify 'ggplot2' visualisation with 'ggblanket' wrapper functions.

**License** MIT + file LICENSE

**URL** <https://davidhodge931.github.io/ggblanket/>,  
<https://github.com/davidhodge931/ggblanket>

**BugReports** <https://github.com/davidhodge931/ggblanket/issues>

**Imports** colorspace, dplyr (>= 1.0.4), farver, forcats, ggplot2 (>= 3.5.0), grid, hms (>= 0.5.0), labelled, lubridate (>= 1.7.8), magrittr, purrr, rlang (>= 1.1.0), scales (>= 1.3.0), snakecase, stringr (>= 1.3.0), tidyr (>= 1.0.0), tidyselect (>= 1.2.0), viridisLite

**Suggests** spelling, hexbin, isoband, knitr, palmerpenguins, patchwork, quantreg, rmarkdown, sf, testthat (>= 3.0.0), tibble, vdiff, viridis

**VignetteBuilder** knitr

**Config/Needs/website** corrr, farver, ggbeeswarm, ggblend, ggdensity, ggdist, ggeasy, ggforce, ggh4x, gghighlight, ggnewscale, ggrepel, ggridges, ggpattern, glue, marquee, paletteer, showtext, sysfonts

**Config/testthat/edition** 3

**Encoding** UTF-8

**Language** en-GB

**RoxygenNote** 7.3.1

**NeedsCompilation** no

**Author** David Hodge [aut, cre, cph] (<<https://orcid.org/0000-0002-3868-7501>>)

**Maintainer** David Hodge <davidhodge931@gmail.com>

**Repository** CRAN

**Date/Publication** 2024-06-14 06:50:02 UTC

## Contents

aes_contrast . . . . .	3
blue . . . . .	4
dark_mode_r . . . . .	5
gg_area . . . . .	9
gg_bar . . . . .	13
gg_bin_2d . . . . .	17
gg_blanket . . . . .	21
gg_boxplot . . . . .	26
gg_col . . . . .	30
gg_contour . . . . .	34
gg_contour_filled . . . . .	38
gg_crossbar . . . . .	42
gg_density . . . . .	47
gg_density_2d . . . . .	51
gg_density_2d_filled . . . . .	55
gg_errorbar . . . . .	59
gg_freqpoly . . . . .	63
gg_function . . . . .	68
gg_hex . . . . .	72
gg_histogram . . . . .	76
gg_jitter . . . . .	80
gg_label . . . . .	84
gg_line . . . . .	88
gg_linerange . . . . .	93
gg_path . . . . .	97
gg_point . . . . .	101
gg_pointrange . . . . .	105
gg_polygon . . . . .	109
gg_qq . . . . .	114
gg_quantile . . . . .	118
gg_raster . . . . .	122
gg_rect . . . . .	126
gg_ribbon . . . . .	131
gg_rug . . . . .	135
gg_segment . . . . .	139
gg_sf . . . . .	143
gg_smooth . . . . .	147
gg_step . . . . .	152
gg_text . . . . .	156
gg_tile . . . . .	160
gg_violin . . . . .	164
grey . . . . .	169
grey_mode_r . . . . .	169
jumble . . . . .	173
lightness . . . . .	174
light_mode_r . . . . .	175

set\_blanket . . . . . 179

**Index** **181**

aes\_contrast *An auto-contrast colour aesthetic*

## Description

A colour aesthetic for annotation that automatically contrasts with fill. Can be spliced into `ggplot2::aes` with `rlang::!!!`.

## Usage

```
aes_contrast(..., dark = "#121B24FF", light = "#FFFFFFF")
```

## Arguments

<code>...</code>	Provided to force user argument naming etc.
<code>dark</code>	A dark colour.
<code>light</code>	A light colour.

## Value

An aesthetic

## Examples

```
library(ggplot2)
library(dplyr)
library(stringr)
library(palmerpenguins)

set_blanket()

penguins |>
  count(species, sex) |>
  gg_col(
    x = sex,
    y = n,
    col = species,
    position = position_dodge2(preserve = "single"),
    width = 0.75,
    x_labels = \(x) str_to_sentence(x),
  ) +
  geom_text(
    mapping = aes(label = n, !!!aes_contrast()),
    position = position_dodge2(width = 0.75, preserve = "single"),
    vjust = 1.33,
    show.legend = FALSE,
```

```
)  
penguins |>  
  count(species, sex) |>  
  gg_col(  
    x = sex,  
    y = n,  
    col = species,  
    position = position_dodge2(preserve = "single"),  
    width = 0.75,  
    x_labels = \(x) str_to_sentence(x),  
    mode = dark_mode_r(),  
  ) +  
  geom_text(  
    mapping = aes(label = n, !!!aes_contrast(dark = darkness[3], light = darkness[1])),  
    position = position_dodge2(width = 0.75, preserve = "single"),  
    vjust = 1.33,  
    show.legend = FALSE,  
  )
```

---

blue

*A blue colour*

---

### Description

A blue colour.

### Usage

```
blue
```

### Value

A character vector.

### Examples

```
scales::show_col(blue)
```

---

`dark_mode_r`*Dark mode theme family*

---

## Description

A dark mode family of functions:

- `dark_mode_r()` with legend on right
- `dark_mode_t()` with legend on top
- `dark_mode_b()` with legend on bottom
- `dark_mode_n()` with no legend

## Usage

```
dark_mode_r(  
  ...,  
  base_size = 11,  
  base_family = "",  
  base_colour = "#C8D7DFFF",  
  axis_line_colour = "#C8D7DFFF",  
  axis_line_linewidth = 0.33,  
  axis_ticks_colour = axis_line_colour,  
  axis_ticks_linewidth = axis_line_linewidth,  
  axis_ticks_length_x = grid::unit(base_size/3, "pt"),  
  axis_ticks_length_y = grid::unit(base_size/4, "pt"),  
  panel_grid_colour = "#00040AFF",  
  panel_grid_linewidth = 1.33,  
  panel_background_fill = "#050D1BFF",  
  plot_background_fill = "#00040AFF",  
  legend_axis_line_colour = plot_background_fill,  
  legend_axis_line_linewidth = 0.33,  
  legend_background_fill = plot_background_fill,  
  legend_key_fill = plot_background_fill,  
  legend_ticks_colour = legend_axis_line_colour,  
  legend_ticks_linewidth = legend_axis_line_linewidth,  
  legend_ticks_length = ggplot2::rel(c(0.175, 0)),  
  orientation = NULL  
)
```

```
dark_mode_t(  
  ...,  
  base_size = 11,  
  base_family = "",  
  base_colour = "#C8D7DFFF",  
  axis_line_colour = "#C8D7DFFF",  
  axis_line_linewidth = 0.33,
```

```

axis_ticks_colour = axis_line_colour,
axis_ticks_linewidth = axis_line_linewidth,
axis_ticks_length_x = grid::unit(base_size/3, "pt"),
axis_ticks_length_y = grid::unit(base_size/4, "pt"),
panel_grid_colour = "#00040AFF",
panel_grid_linewidth = 1.33,
panel_background_fill = "#050D1BFF",
plot_background_fill = "#00040AFF",
legend_axis_line_colour = plot_background_fill,
legend_axis_line_linewidth = 0.33,
legend_background_fill = plot_background_fill,
legend_key_fill = plot_background_fill,
legend_ticks_colour = legend_axis_line_colour,
legend_ticks_linewidth = legend_axis_line_linewidth,
legend_ticks_length = ggplot2::rel(c(0.175, 0)),
orientation = NULL
)

```

```

dark_mode_b(
  ...,
  base_size = 11,
  base_family = "",
  base_colour = "#C8D7DFFF",
  axis_line_colour = "#C8D7DFFF",
  axis_line_linewidth = 0.33,
  axis_ticks_colour = axis_line_colour,
  axis_ticks_linewidth = axis_line_linewidth,
  axis_ticks_length_x = grid::unit(base_size/3, "pt"),
  axis_ticks_length_y = grid::unit(base_size/4, "pt"),
  panel_grid_colour = "#00040AFF",
  panel_grid_linewidth = 1.33,
  panel_background_fill = "#050D1BFF",
  plot_background_fill = "#00040AFF",
  legend_axis_line_colour = plot_background_fill,
  legend_axis_line_linewidth = 0.33,
  legend_background_fill = plot_background_fill,
  legend_key_fill = plot_background_fill,
  legend_ticks_colour = legend_axis_line_colour,
  legend_ticks_linewidth = legend_axis_line_linewidth,
  legend_ticks_length = ggplot2::rel(c(0.175, 0)),
  orientation = NULL
)

```

```

dark_mode_n(
  ...,
  base_size = 11,
  base_family = "",
  base_colour = "#C8D7DFFF",

```

```

axis_line_colour = "#C8D7DFFF",
axis_line_linewidth = 0.33,
axis_ticks_colour = axis_line_colour,
axis_ticks_linewidth = axis_line_linewidth,
axis_ticks_length_x = grid::unit(base_size/3, "pt"),
axis_ticks_length_y = grid::unit(base_size/4, "pt"),
panel_grid_colour = "#00040AFF",
panel_grid_linewidth = 1.33,
panel_background_fill = "#050D1BFF",
plot_background_fill = "#00040AFF",
legend_axis_line_colour = plot_background_fill,
legend_axis_line_linewidth = 0.33,
legend_background_fill = plot_background_fill,
legend_key_fill = plot_background_fill,
legend_ticks_colour = legend_axis_line_colour,
legend_ticks_linewidth = legend_axis_line_linewidth,
legend_ticks_length = ggplot2::rel(c(0.175, 0)),
orientation = NULL
)

```

### Arguments

...	Provided to force user argument naming etc.
base_size	The base size of the text theme element. Defaults to 11.
base_family	The base family of the text theme element. Defaults to "".
base_colour	The base colour of the text theme element.
axis_line_colour	The colour of the axis.line theme element.
axis_line_linewidth	The linewidth of the axis.line theme element.
axis_ticks_colour	The colour of the axis.ticks theme element.
axis_ticks_linewidth	The linewidth of the axis.ticks theme element.
axis_ticks_length_x	The length of the axis.ticks.length.x theme element.
axis_ticks_length_y	The length of the axis.ticks.length.y theme element.
panel_grid_colour	The colour of the panel.grid theme element.
panel_grid_linewidth	The linewidth of the panel.grid theme element.
panel_background_fill	The fill (and colour) of the panel.background theme element.
plot_background_fill	The fill (and colour) of the plot.background theme element.

legend_axis_line_colour	The colour of the legend.axis.line theme element.
legend_axis_line_linewidth	The linewidth of the legend.axis.line theme element.
legend_background_fill	The fill (and colour) of the legend.background theme element.
legend_key_fill	The fill (and colour) of the legend.key theme element.
legend_ticks_colour	The colour of the legend.ticks theme element.
legend_ticks_linewidth	The linewidth of the legend.ticks theme element.
legend_ticks_length	The legend.ticks.length theme element.
orientation	The orientation of the plot. Either "x" or "y". Defaults to NULL. Not intended for use with the mode argument of gg_* functions.

### Value

A ggplot theme.

### Examples

```
library(palmerpenguins)
library(ggplot2)

set_blanket()

penguins |>
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = species,
    mode = dark_mode_r()
  )

penguins |>
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = species,
    mode = dark_mode_t()
  )

penguins |>
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = species,
    mode = dark_mode_b()
  )
```



```
)  
  
penguins |>  
gg_point(  
  x = flipper_length_mm,  
  y = body_mass_g,  
  col = species,  
  mode = dark_mode_n()  
)
```

---

gg\_area

*Area ggplot*

---

### Description

Create an area ggplot with a wrapper around `ggplot2::ggplot() + geom_area()`.

### Usage

```
gg_area(  
  data = NULL,  
  ...,  
  stat = "align",  
  position = "stack",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,
```

```
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,  
facet_ncol = NULL,  
facet_nrow = NULL,  
facet_scales = "fixed",  
facet_space = "fixed",  
title = NULL,  
subtitle = NULL,  
caption = NULL,  
label_to_case = snakecase::to_sentence_case  
)
```

**Arguments**

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.

<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> ( <code>)</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> ( <code>)</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)

set_blanket()

economics |>
  gg_area(
    x = date,
    y = unemploy,
    y_label = "Unemployment",
  )
```

---

gg\_bar

*Bar ggplot*

---

**Description**

Create a bar ggplot with a wrapper around `ggplot2::ggplot() + geom_bar()`.

**Usage**

```
gg_bar(
  data = NULL,
  ...,
  stat = "count",
  position = "stack",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
```

```
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,  
facet_ncol = NULL,  
facet_nrow = NULL,  
facet_scales = "fixed",  
facet_space = "fixed",
```

```

title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*</code> () function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*</code> () function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.
<code>mapping</code>	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
<code>x_breaks</code> , <code>y_breaks</code> , <code>col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
<code>x_expand</code> , <code>y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits</code> , <code>y_expand_limits</code> , <code>col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels</code> , <code>y_labels</code> , <code>col_labels</code> , <code>facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits</code> , <code>y_limits</code> , <code>col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob</code> , <code>y_oob</code> , <code>col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .

<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> ( <code>)</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> ( <code>)</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .



**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins |>
  tidyr::drop_na(sex) |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  gg_bar(
    y = species,
    col = sex,
    position = position_dodge(preserve = "single"),
    width = 0.75,
  )
```

---

*gg\_bin\_2d**Bin\_2d ggplot*

---

**Description**

Create a bin2d ggplot with a wrapper around `ggplot2::ggplot() + geom_bin_2d()`.

**Usage**

```
gg_bin_2d(
  data = NULL,
  ...,
  stat = "bin2d",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
```

```
facet = NULL,  
facet2 = NULL,  
group = NULL,  
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,
```

```

facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.
<code>mapping</code>	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
<code>x_breaks</code> , <code>y_breaks</code> , <code>col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
<code>x_expand</code> , <code>y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits</code> , <code>y_expand_limits</code> , <code>col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels</code> , <code>y_labels</code> , <code>col_labels</code> , <code>facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).

<code>x_limits, y_limits, col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_prefix</code> (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> ( <code>)</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>+*_mode_*</code> ( <code>)</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)

set_blanket()

diamonds |>
  gg_bin_2d(
    x = carat,
    y = price,
  )
```

---

gg\_blanket

*Blanket ggplot*


---

**Description**

Create a blanket ggplot with a wrapper around `ggplot2::ggplot() + layer()` with `geom_blank()` defaults. This function underlies all other `gg_*` functions. It contains a `geom` argument for maximum flexibility.

**Usage**

```
gg_blanket(
  data = NULL,
  ...,
  geom = "blank",
  stat = "identity",
  position = "identity",
  coord = NULL,
  mode = NULL,
```

```
x = NULL,  
xmin = NULL,  
xmax = NULL,  
xend = NULL,  
y = NULL,  
ymin = NULL,  
ymax = NULL,  
yend = NULL,  
z = NULL,  
col = NULL,  
facet = NULL,  
facet2 = NULL,  
group = NULL,  
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,
```

```

col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
geom	A geometric object to display the data. A snakecase character string of a ggproto Geom subclass object minus the Geom prefix (e.g. "point").
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.

<code>x_expand, y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits, y_expand_limits, col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels, y_labels, col_labels, facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits, y_limits, col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_prefix</code> (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>+ *_mode_*</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".



facet_layout	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
facet_ncol, facet_nrow	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins %>%
  tidyr::drop_na(sex) %>%
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) %>%
  gg_blanket(
    geom = "violin",
    stat = "ydensity",
    position = "dodge",
    x = sex,
    y = body_mass_g,
    col = sex,
    facet = species,
    mode = grey_mode_b(),
  )
```

---

`gg_boxplot`*Boxplot ggplot*

---

**Description**

Create a boxplot ggplot with a wrapper around `ggplot2::ggplot() + geom_boxplot()`.

**Usage**

```
gg_boxplot(  
  data = NULL,  
  ...,  
  stat = "boxplot",  
  position = "dodge2",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,  
  x_orientation = NULL,  
  x_position = "bottom",  
  x_label = NULL,  
  x_transform = NULL,  
  y_breaks = NULL,  
  y_expand = NULL,  
  y_expand_limits = NULL,  
)
```

```

y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).

mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use + <code>ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.

<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> () function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> () may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol</code> , <code>facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins |>
  tidyr::drop_na(sex) |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  gg_boxplot(
```

```
x = flipper_length_mm,  
y = sex,  
col = species,  
mode = light_mode_b(),  
)
```

---

gg\_col

*Col ggplot*

---

## Description

Create a col ggplot with a wrapper around `ggplot2::ggplot() + geom_col()`.

## Usage

```
gg_col(  
  data = NULL,  
  ...,  
  stat = "identity",  
  position = "stack",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,
```

```

x_orientation = NULL,
x_position = "bottom",
x_label = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a <code>params</code> list in <code>layer()</code> .

stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).
coord	A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).
mode	A ggplot2 theme (e.g. light_mode_t(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of gg_*.
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in ggplot2::aes(). Intended primarily for non-supported aesthetics (e.g. shape, linetype, linewidth, or size), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A scales::breaks_* function (e.g. scales::breaks_*()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. 0). For a discrete scale, manipulate the data instead with forcats::fct_expand.
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. \(\times\) stringr::str_to_sentence(x) or scales::label_*()), or a vector of labels. (Note this must be named for facet_labels).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with factor, forcats::fct_expand or forcats::fct_drop.
x_oob, y_oob, col_oob	For a continuous scale, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".
x_label, y_label, col_label	Label for the axis or legend title. Use + ggplot2::labs(... = NULL) for no title.



<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. <code>"log10"</code> ).
<code>col_drop, facet_drop</code>	For a discrete variable, <code>FALSE</code> or <code>TRUE</code> of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	<code>TRUE</code> or <code>FALSE</code> of whether to reverse the elements of a legend guide. Defaults to <code>FALSE</code> .
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, <code>TRUE</code> or <code>FALSE</code> of whether to colour in steps. Defaults to <code>FALSE</code> .
<code>facet_axes</code>	Whether to add interior axes and ticks with <code>"margins"</code> , <code>"all"</code> , <code>"all_x"</code> , or <code>"all_y"</code> . Sometimes + <code>*_mode_*</code> () may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with <code>"margins"</code> , <code>"all"</code> , <code>"all_x"</code> , or <code>"all_y"</code> .
<code>facet_layout</code>	Whether the layout is to be <code>"wrap"</code> or <code>"grid"</code> . If <code>NULL</code> and a single facet (or <code>facet2</code> ) argument is provided, then defaults to <code>"wrap"</code> . If <code>NULL</code> and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to <code>"grid"</code> .
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of <code>"wrap"</code> .
<code>facet_scales</code>	Whether facet scales should be <code>"fixed"</code> across facets, <code>"free"</code> in both directions, or free in just one direction (i.e. <code>"free_x"</code> or <code>"free_y"</code> ). Defaults to <code>"fixed"</code> .
<code>facet_space</code>	When the facet layout is <code>"grid"</code> and facet scales are not <code>"fixed"</code> , whether facet space should be <code>"fixed"</code> across facets, <code>"free"</code> to be proportional in both directions, or free to be proportional in just one direction (i.e. <code>"free_x"</code> or <code>"free_y"</code> ). Defaults to <code>"fixed"</code> .
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

## Examples

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins |>
  tidyr::drop_na(sex) |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  group_by(sex, species) |>
  summarise(across(flipper_length_mm, \(x) mean(x, na.rm = TRUE))) |>
  gg_col(
    x = flipper_length_mm,
    y = species,
    col = sex,
    position = position_dodge(preserve = "single"),
    width = 0.75,
  )
```

---

gg\_contour

*Contour ggplot*

---

## Description

Create a contour ggplot with a wrapper around `ggplot2::ggplot() + geom_contour()`.

## Usage

```
gg_contour(
  data = NULL,
  ...,
  stat = "contour",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  facet = NULL,
```

```
facet2 = NULL,  
group = NULL,  
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,  
facet_ncol = NULL,
```

```

facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.
<code>mapping</code>	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
<code>x_breaks</code> , <code>y_breaks</code> , <code>col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
<code>x_expand</code> , <code>y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits</code> , <code>y_expand_limits</code> , <code>col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels</code> , <code>y_labels</code> , <code>col_labels</code> , <code>facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits</code> , <code>y_limits</code> , <code>col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .

x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.
col_legend_rev	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
col_palette	A character vector of hex codes (or names) or a <code>scales::pal_*</code> ( <code>)</code> function.
col_palette_na	A hex code (or name) for the colour of NA values.
col_rescale	For a continuous variable, a <code>scales::rescale()</code> function.
col_steps	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
facet_axes	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> ( <code>)</code> may be needed.
facet_axis_labels	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_layout	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
facet_ncol, facet_nrow	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

### Value

A ggplot object.

### Examples

```
library(ggplot2)
library(dplyr)

set_blanket()

ggplot2::faithfuld |>
  gg_contour(
    x = waiting,
    y = eruptions,
    z = density,
  )
```

---

gg\_contour\_filled      *Contour\_filled ggplot*

---

### Description

Create a contour\_filled ggplot with a wrapper around `ggplot2::ggplot() + geom_contour_filled()`.

### Usage

```
gg_contour_filled(
  data = NULL,
  ...,
  stat = "contour_filled",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
```

```
ymax = NULL,  
yend = NULL,  
z = NULL,  
col = NULL,  
facet = NULL,  
facet2 = NULL,  
group = NULL,  
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,
```

```

facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*</code> () function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*</code> () function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .



x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.
col_legend_rev	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
col_palette	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
col_palette_na	A hex code (or name) for the colour of NA values.
col_rescale	For a continuous variable, a <code>scales::rescale()</code> function.
col_steps	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
facet_axes	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>+ *_mode_*</code> may be needed.
facet_axis_labels	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_layout	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
facet_ncol, facet_nrow	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".

facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)

set_blanket()

faithfuld |>
  gg_contour_filled(
    x = waiting,
    y = eruptions,
    z = density,
    bins = 8,
  )
```

---

gg\_crossbar

*Crossbar ggplot*


---

**Description**

Create a crossbar ggplot with a wrapper around `ggplot2::ggplot() + geom_crossbar()`.

**Usage**

```
gg_crossbar(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
```

```
coord = ggplot2::coord_cartesian(clip = "off"),
mode = NULL,
x = NULL,
xmin = NULL,
xmax = NULL,
xend = NULL,
y = NULL,
ymin = NULL,
ymax = NULL,
yend = NULL,
z = NULL,
col = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_orientation = NULL,
x_position = "bottom",
x_label = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
```

```

col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to <code>within</code> a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*</code> () function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*</code> () function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.
<code>mapping</code>	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
<code>x_breaks</code> , <code>y_breaks</code> , <code>col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.

<code>x_expand, y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits, y_expand_limits, col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels, y_labels, col_labels, facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits, y_limits, col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_prefix</code> (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>+ *_mode_*()</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_layout	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
facet_ncol, facet_nrow	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)

set_blanket()

data.frame(
  trt = factor(c(1, 1, 2, 2)),
  resp = c(1, 5, 3, 4),
  group = factor(c(1, 2, 1, 2)),
  upper = c(1.1, 5.3, 3.3, 4.2),
  lower = c(0.8, 4.6, 2.4, 3.6)
) |>
gg_crossbar(
  x = trt,
  y = resp,
  ymin = lower,
  ymax = upper,
  col = group,
  width = 0.5,
  x_label = "Treatment",
  y_label = "Response",
)
```

---

`gg_density`*Density ggplot*

---

**Description**

Create a density ggplot with a wrapper around `ggplot2::ggplot() + geom_density()`.

**Usage**

```
gg_density(  
  data = NULL,  
  ...,  
  stat = "density",  
  position = "identity",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,  
  x_orientation = NULL,  
  x_position = "bottom",  
  x_label = NULL,  
  x_transform = NULL,  
  y_breaks = NULL,  
  y_expand = NULL,  
  y_expand_limits = NULL,
```

```

y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).



mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use + <code>ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.

<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> () function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> () may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol</code> , <code>facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  tidyr::drop_na(sex) |>
  gg_density(
```

```
x = flipper_length_mm,  
col = species,  
mode = light_mode_t(),  
)
```

---

gg\_density\_2d

*Density\_2d ggplot*

---

## Description

Create a density\_2d ggplot with a wrapper around `ggplot2::ggplot()` + `geom_density_2d()`.

## Usage

```
gg_density_2d(  
  data = NULL,  
  ...,  
  stat = "density_2d",  
  position = "identity",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,  
  x_orientation = NULL,
```

```

x_position = "bottom",
x_label = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a

	ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).
coord	A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).
mode	A ggplot2 theme (e.g. light_mode_t(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of gg_*.
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in ggplot2::aes(). Intended primarily for non-supported aesthetics (e.g. shape, linetype, linewidth, or size), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A scales::breaks_* function (e.g. scales::breaks_*()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. 0). For a discrete scale, manipulate the data instead with forcats::fct_expand.
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. \(\times\) stringr::str_to_sentence(x) or scales::label_*()), or a vector of labels. (Note this must be named for facet_labels).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with factor, forcats::fct_expand or forcats::fct_drop.
x_oob, y_oob, col_oob	For a continuous scale, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".
x_label, y_label, col_label	Label for the axis or legend title. Use + ggplot2::labs(... = NULL) for no title.

<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_prefix</code> (e.g. <code>"log10"</code> ).
<code>col_drop, facet_drop</code>	For a discrete variable, <code>FALSE</code> or <code>TRUE</code> of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	<code>TRUE</code> or <code>FALSE</code> of whether to reverse the elements of a legend guide. Defaults to <code>FALSE</code> .
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, <code>TRUE</code> or <code>FALSE</code> of whether to colour in steps. Defaults to <code>FALSE</code> .
<code>facet_axes</code>	Whether to add interior axes and ticks with <code>"margins"</code> , <code>"all"</code> , <code>"all_x"</code> , or <code>"all_y"</code> . Sometimes + <code>*_mode_*</code> () may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with <code>"margins"</code> , <code>"all"</code> , <code>"all_x"</code> , or <code>"all_y"</code> .
<code>facet_layout</code>	Whether the layout is to be <code>"wrap"</code> or <code>"grid"</code> . If <code>NULL</code> and a single facet (or <code>facet2</code> ) argument is provided, then defaults to <code>"wrap"</code> . If <code>NULL</code> and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to <code>"grid"</code> .
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of <code>"wrap"</code> .
<code>facet_scales</code>	Whether facet scales should be <code>"fixed"</code> across facets, <code>"free"</code> in both directions, or free in just one direction (i.e. <code>"free_x"</code> or <code>"free_y"</code> ). Defaults to <code>"fixed"</code> .
<code>facet_space</code>	When the facet layout is <code>"grid"</code> and facet scales are not <code>"fixed"</code> , whether facet space should be <code>"fixed"</code> across facets, <code>"free"</code> to be proportional in both directions, or free to be proportional in just one direction (i.e. <code>"free_x"</code> or <code>"free_y"</code> ). Defaults to <code>"fixed"</code> .
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A `ggplot` object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

faithful |>
  gg_density_2d(
    x = waiting,
    y = eruptions,
    bins = 8,
    contour = TRUE,
  )
```

---

`gg_density_2d_filled` *Density\_2d\_filled* ggplot

---

**Description**

Create a `density_2d_filled` ggplot with a wrapper around `ggplot2::ggplot() + geom_density_2d_filled()`.

**Usage**

```
gg_density_2d_filled(
  data = NULL,
  ...,
  stat = "density_2d_filled",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
```

```
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,  
facet_ncol = NULL,  
facet_nrow = NULL,  
facet_scales = "fixed",  
facet_space = "fixed",  
title = NULL,  
subtitle = NULL,
```



```

  caption = NULL,
  label_to_case = snakecase::to_sentence_case
)

```

## Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. shape, linetype, linewidth, or size), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. 0). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .

<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> ( <code>)</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> ( <code>)</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

faithful |>
  gg_density_2d_filled(
    x = waiting,
    y = eruptions,
    bins = 8,
    contour = TRUE,
  )
```

---

gg\_errorbar

*Errorbar ggplot*

---

**Description**

Create a errorbar ggplot with a wrapper around `ggplot2::ggplot() + geom_errorbar()`.

**Usage**

```
gg_errorbar(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  facet = NULL,
  facet2 = NULL,
```

```
group = NULL,  
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,  
facet_ncol = NULL,  
facet_nrow = NULL,
```

```

facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a <code>params</code> list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.
<code>mapping</code>	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
<code>x_breaks</code> , <code>y_breaks</code> , <code>col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
<code>x_expand</code> , <code>y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits</code> , <code>y_expand_limits</code> , <code>col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels</code> , <code>y_labels</code> , <code>col_labels</code> , <code>facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits</code> , <code>y_limits</code> , <code>col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .

<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> ( <code>)</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>+ *_mode_*</code> ( <code>)</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

### Value

A ggplot object.

### Examples

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

data.frame(
  trt = factor(c(1, 1, 2, 2)),
  resp = c(1, 5, 3, 4),
  group = factor(c(1, 2, 1, 2)),
  upper = c(1.1, 5.3, 3.3, 4.2),
  lower = c(0.8, 4.6, 2.4, 3.6)
) |>
gg_errorbar(
  x = trt,
  ymin = lower,
  ymax = upper,
  col = group,
  width = 0.1,
  x_label = "Treatment",
  y_label = "Response",
)
```

---

gg\_freqpoly

*Freqpoly ggplot*

---

### Description

Create a freqpoly ggplot with a wrapper around `ggplot2::ggplot() + geom_freqpoly()`.

### Usage

```
gg_freqpoly(
  data = NULL,
  ...,
```

```
stat = "bin",
position = "identity",
coord = ggplot2::coord_cartesian(clip = "off"),
mode = NULL,
x = NULL,
xmin = NULL,
xmax = NULL,
xend = NULL,
y = NULL,
ymin = NULL,
ymax = NULL,
yend = NULL,
z = NULL,
col = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_orientation = NULL,
x_position = "bottom",
x_label = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
```



```

col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*</code> () function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*</code> () function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.

<code>x_breaks, y_breaks, col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
<code>x_expand, y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits, y_expand_limits, col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels, y_labels, col_labels, facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits, y_limits, col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use + <code>ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_prefix</code> (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> () function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> () may be needed.

facet_axis_labels	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_layout	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
facet_ncol, facet_nrow	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

## Value

A ggplot object.

## Examples

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  gg_freqpoly(
    x = flipper_length_mm,
    col = sex,
    col_label = "",
    mode = light_mode_t(),
  )
```

---

`gg_function`*Function ggplot*

---

**Description**

Create a function `ggplot` with a wrapper around `ggplot2::ggplot()` + `geom_function()`.

**Usage**

```
gg_function(  
  data = NULL,  
  ...,  
  stat = "function",  
  position = "identity",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,  
  x_orientation = NULL,  
  x_position = "bottom",  
  x_label = NULL,  
  x_transform = NULL,  
  y_breaks = NULL,  
  y_expand = NULL,  
  y_expand_limits = NULL,  
)
```

```

y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).

mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use + <code>ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.

<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> () function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> () may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol</code> , <code>facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

## Value

A ggplot object.

## Examples

```
library(ggplot2)
library(dplyr)

set_blanket()

gg_function(
  fun = \(x) dnorm(x, mean = 0, sd = 5),
  x_limits = qnorm(p = c(0.005, 0.995), mean = 0, sd = 5),
  y_expand_limits = 0,
)
```

---

`gg_hex`*Hex ggplot*

---

**Description**

Create a hex ggplot with a wrapper around `ggplot2::ggplot() + geom_hex()`.

**Usage**

```
gg_hex(  
  data = NULL,  
  ...,  
  stat = "binhex",  
  position = "identity",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,  
  x_orientation = NULL,  
  x_position = "bottom",  
  x_label = NULL,  
  x_transform = NULL,  
  y_breaks = NULL,  
  y_expand = NULL,  
  y_expand_limits = NULL,  
)
```



```

y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*</code> () function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*</code> () function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).

mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use + <code>ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.

<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> () function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> () may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol</code> , <code>facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

diamonds |>
  gg_hex(
    x = carat,
    y = price,
```

```
coord = coord_cartesian(clip = "on"),
y_limits = c(0, 20000),
)
```

---

gg\_histogram

*Histogram ggplot*

---

### Description

Create a histogram ggplot with a wrapper around `ggplot2::ggplot() + geom_histogram()`.

### Usage

```
gg_histogram(  
  data = NULL,  
  ...,  
  stat = "bin",  
  position = "stack",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,  
  x_orientation = NULL,  
  x_position = "bottom",
```

```

x_label = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").

position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).
coord	A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).
mode	A ggplot2 theme (e.g. light_mode_t(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of gg_*.
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in ggplot2::aes(). Intended primarily for non-supported aesthetics (e.g. shape, linetype, linewidth, or size), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A scales::breaks_* function (e.g. scales::breaks_*()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the ggplot2::expansion() function, or a vector of length 2 (e.g. c(0, 0)).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. 0). For a discrete scale, manipulate the data instead with forcats::fct_expand.
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. \(\x\) stringr::str_to_sentence(x) or scales::label_*()), or a vector of labels. (Note this must be named for facet_labels).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with factor, forcats::fct_expand or forcats::fct_drop.
x_oob, y_oob, col_oob	For a continuous scale, a scales::oob_* function of how to handle values outside of limits. Defaults to scales::oob_keep.
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using y_position = "top" with a *_mode_* theme, add caption = "" or caption = "\n".
x_label, y_label, col_label	Label for the axis or legend title. Use + ggplot2::labs(... = NULL) for no title.

<code>x_transform</code> , <code>y_transform</code> , <code>col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. <code>"log10"</code> ).
<code>col_drop</code> , <code>facet_drop</code>	For a discrete variable, <code>FALSE</code> or <code>TRUE</code> of whether to drop unused levels.
<code>col_legend_ncol</code> , <code>col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	<code>TRUE</code> or <code>FALSE</code> of whether to reverse the elements of a legend guide. Defaults to <code>FALSE</code> .
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, <code>TRUE</code> or <code>FALSE</code> of whether to colour in steps. Defaults to <code>FALSE</code> .
<code>facet_axes</code>	Whether to add interior axes and ticks with <code>"margins"</code> , <code>"all"</code> , <code>"all_x"</code> , or <code>"all_y"</code> . Sometimes + <code>*_mode_*</code> () may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with <code>"margins"</code> , <code>"all"</code> , <code>"all_x"</code> , or <code>"all_y"</code> .
<code>facet_layout</code>	Whether the layout is to be <code>"wrap"</code> or <code>"grid"</code> . If <code>NULL</code> and a single facet (or <code>facet2</code> ) argument is provided, then defaults to <code>"wrap"</code> . If <code>NULL</code> and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to <code>"grid"</code> .
<code>facet_ncol</code> , <code>facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of <code>"wrap"</code> .
<code>facet_scales</code>	Whether facet scales should be <code>"fixed"</code> across facets, <code>"free"</code> in both directions, or free in just one direction (i.e. <code>"free_x"</code> or <code>"free_y"</code> ). Defaults to <code>"fixed"</code> .
<code>facet_space</code>	When the facet layout is <code>"grid"</code> and facet scales are not <code>"fixed"</code> , whether facet space should be <code>"fixed"</code> across facets, <code>"free"</code> to be proportional in both directions, or free to be proportional in just one direction (i.e. <code>"free_x"</code> or <code>"free_y"</code> ). Defaults to <code>"fixed"</code> .
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  gg_histogram(
    x = flipper_length_mm,
    col = sex,
    facet = species,
    bins = 50,
    mode = light_mode_b(),
  )
```

---

gg\_jitter

*Jitter ggplot*

---

**Description**

Create a jitter ggplot with a wrapper around `ggplot2::ggplot() + geom_jitter()`.

**Usage**

```
gg_jitter(
  data = NULL,
  ...,
  stat = "identity",
  position = "jitter",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
```



```
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,  
facet_ncol = NULL,  
facet_nrow = NULL,  
facet_scales = "fixed",  
facet_space = "fixed",
```

```

  title = NULL,
  subtitle = NULL,
  caption = NULL,
  label_to_case = snakecase::to_sentence_case
)

```

## Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*</code> () function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*</code> () function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.
<code>mapping</code>	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
<code>x_breaks</code> , <code>y_breaks</code> , <code>col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
<code>x_expand</code> , <code>y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits</code> , <code>y_expand_limits</code> , <code>col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. 0). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels</code> , <code>y_labels</code> , <code>col_labels</code> , <code>facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits</code> , <code>y_limits</code> , <code>col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob</code> , <code>y_oob</code> , <code>col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .

<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> ( <code>)</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> ( <code>)</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

set.seed(123)

penguins |>
  gg_jitter(
    x = species,
    y = body_mass_g,
    col = flipper_length_mm,
    position = position_jitter(height = 0),
    y_expand_limits = 0,
    col_steps = TRUE,
  )
```

---

gg\_label

*Label ggplot*

---

**Description**

Create a label ggplot with a wrapper around `ggplot2::ggplot()` + `geom_label()`.

**Usage**

```
gg_label(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
```

```
z = NULL,  
col = NULL,  
facet = NULL,  
facet2 = NULL,  
group = NULL,  
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,
```

```

facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a <code>params</code> list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.
<code>mapping</code>	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
<code>x_breaks</code> , <code>y_breaks</code> , <code>col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
<code>x_expand</code> , <code>y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits</code> , <code>y_expand_limits</code> , <code>col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. 0). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels</code> , <code>y_labels</code> , <code>col_labels</code> , <code>facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).

x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use <code>+ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_prefix</code> (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.
col_legend_rev	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
col_palette	A character vector of hex codes (or names) or a <code>scales::pal_*</code> function.
col_palette_na	A hex code (or name) for the colour of NA values.
col_rescale	For a continuous variable, a <code>scales::rescale()</code> function.
col_steps	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
facet_axes	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>*_mode_*</code> may be needed.
facet_axis_labels	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_layout	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
facet_ncol, facet_nrow	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

### Value

A ggplot object.

### Examples

```
library(ggplot2)
library(dplyr)

set_blanket()

bind_rows(
  mtcars |> slice_min(order_by = mpg),
  mtcars |> slice_max(order_by = mpg)
) |>
tibble::rownames_to_column("model") |>
gg_label(
  x = model,
  y = mpg,
  col = mpg,
  label = model,
  y_expand_limits = 0,
  y_label = "Miles per gallon",
  col_palette = c(orange, "white", teal),
)
```

---

gg\_line

*Line ggplot*

---

### Description

Create a line ggplot with a wrapper around `ggplot2::ggplot() + geom_line()`.



**Usage**

```
gg_line(  
  data = NULL,  
  ...,  
  stat = "identity",  
  position = "identity",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,  
  x_orientation = NULL,  
  x_position = "bottom",  
  x_label = NULL,  
  x_transform = NULL,  
  y_breaks = NULL,  
  y_expand = NULL,  
  y_expand_limits = NULL,  
  y_labels = NULL,  
  y_limits = NULL,  
  y_oob = scales::oob_keep,  
  y_orientation = NULL,  
  y_position = "left",  
  y_label = NULL,  
  y_transform = NULL,  
  col_breaks = NULL,  
  col_drop = FALSE,  
)
```

```

col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.

mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.
col_legend_rev	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
col_palette	A character vector of hex codes (or names) or a <code>scales::pal_*</code> () function.
col_palette_na	A hex code (or name) for the colour of NA values.
col_rescale	For a continuous variable, a <code>scales::rescale()</code> function.

<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + *_mode_*() may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
<code>facet_ncol</code> , <code>facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)

set_blanket()

economics |>
  gg_line(
    x = date,
    y = unemploy,
    y_expand_limits = 0,
    y_label = "Unemployment",
  )
```

---

gg_linerange	<i>Linerange ggplot</i>
--------------	-------------------------

---

### Description

Create a linerange ggplot with a wrapper around `ggplot2::ggplot() + geom_linerange()`.

### Usage

```
gg_linerange(  
  data = NULL,  
  ...,  
  stat = "identity",  
  position = "identity",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,  
  x_orientation = NULL,  
  x_position = "bottom",  
  x_label = NULL,  
  x_transform = NULL,  
  y_breaks = NULL,  
  y_expand = NULL,  
  y_expand_limits = NULL,  
)
```

```

y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).

mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use + <code>ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.

<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> () may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol</code> , <code>facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

### Value

A ggplot object.

### Examples

```
library(ggplot2)
library(dplyr)

set_blanket()

data.frame(
  trt = factor(c(1, 1, 2, 2)),
  resp = c(1, 5, 3, 4),
  group = factor(c(1, 2, 1, 2)),
  upper = c(1.1, 5.3, 3.3, 4.2),
```



```
  lower = c(0.8, 4.6, 2.4, 3.6)
) |>
gg_linerange(
  x = trt,
  ymin = lower,
  ymax = upper,
  col = group,
  position = position_dodge(width = 0.2),
  x_label = "Treatment",
  y_label = "Response",
)
```

---

gg\_path

*Path ggplot*

---

### Description

Create a path ggplot with a wrapper around `ggplot2::ggplot() + geom_path()`.

### Usage

```
gg_path(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
```

```
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,  
facet_ncol = NULL,  
facet_nrow = NULL,  
facet_scales = "fixed",  
facet_space = "fixed",  
title = NULL,  
subtitle = NULL,  
caption = NULL,  
label_to_case = snakecase::to_sentence_case  
)
```

**Arguments**

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. 0). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.

<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> ( <code>)</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> ( <code>)</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)

set_blanket()

economics |>
  mutate(unemploy_rate = unemploy / pop) |>
  gg_path(
    x = unemploy_rate,
    y = psavert,
    x_label = "Unemployment rate",
    y_expand_limits = 0,
    y_label = "Personal savings rate",
  )
```

---

gg\_point

*Point ggplot*

---

**Description**

Create a point ggplot with a wrapper around `ggplot2::ggplot() + geom_point()`.

**Usage**

```
gg_point(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  facet = NULL,
```

```
facet2 = NULL,  
group = NULL,  
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,  
facet_ncol = NULL,
```

```

facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .

<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> ( <code>)</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> ( <code>)</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".



title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = sex,
  )
```

---

gg_pointrange	<i>Pointrange ggplot</i>
---------------	--------------------------

---

**Description**

Create a pointrange ggplot with a wrapper around `ggplot2::ggplot() + geom_pointrange()`.

**Usage**

```
gg_pointrange(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
```

```
ymin = NULL,  
ymax = NULL,  
yend = NULL,  
z = NULL,  
col = NULL,  
facet = NULL,  
facet2 = NULL,  
group = NULL,  
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,
```

```

facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .

<code>x_labels, y_labels, col_labels, facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits, y_limits, col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>+ *_mode_*</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".

facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

### Value

A ggplot object.

### Examples

```
library(ggplot2)
library(dplyr)

set_blanket()

data.frame(
  trt = factor(c(1, 1, 2, 2)),
  resp = c(1, 5, 3, 4),
  group = factor(c(1, 2, 1, 2)),
  upper = c(1.1, 5.3, 3.3, 4.2),
  lower = c(0.8, 4.6, 2.4, 3.6)
) |>
gg_pointrange(
  x = trt,
  y = resp,
  col = group,
  ymin = lower,
  ymax = upper,
  position = position_dodge(width = 0.2),
  x_label = "Treatment",
  y_label = "Response",
)
```

**Description**

Create a polygon ggplot with a wrapper around `ggplot2::ggplot() + geom_polygon()`.

**Usage**

```
gg_polygon(  
  data = NULL,  
  ...,  
  stat = "identity",  
  position = "identity",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,  
  x_orientation = NULL,  
  x_position = "bottom",  
  x_label = NULL,  
  x_transform = NULL,  
  y_breaks = NULL,  
  y_expand = NULL,  
  y_expand_limits = NULL,  
  y_labels = NULL,  
  y_limits = NULL,  
  y_oob = scales::oob_keep,  
  y_orientation = NULL,  
  y_position = "left",
```

```

y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*</code> () function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*</code> () function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .

x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use <code>+ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.
col_legend_rev	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
col_palette	A character vector of hex codes (or names) or a <code>scales::pal_*</code> () function.



<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> () may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol</code> , <code>facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)

set_blanket()

ids <- factor(c("1.1", "2.1", "1.2", "2.2", "1.3", "2.3"))

values <- data.frame(
  id = ids,
  value = c(3, 3.1, 3.1, 3.2, 3.15, 3.5)
)

positions <- data.frame(
```

```

id = rep(ids, each = 4),
x = c(
  2, 1, 1.1, 2.2, 1, 0, 0.3, 1.1, 2.2, 1.1, 1.2, 2.5, 1.1, 0.3,
  0.5, 1.2, 2.5, 1.2, 1.3, 2.7, 1.2, 0.5, 0.6, 1.3
),
y = c(
  -0.5, 0, 1, 0.5, 0, 0.5, 1.5, 1, 0.5, 1, 2.1, 1.7, 1, 1.5,
  2.2, 2.1, 1.7, 2.1, 3.2, 2.8, 2.1, 2.2, 3.3, 3.2
)
)
)

datapoly <- merge(values, positions, by = c("id"))

datapoly |>
  gg_polygon(
    x = x,
    y = y,
    col = value,
    group = id,
  )

```

---

gg\_qq

*Qq ggplot*


---

## Description

Create a qq ggplot with a wrapper around `ggplot2::ggplot() + geom_qq()`.

## Usage

```

gg_qq(
  data = NULL,
  ...,
  stat = "qq",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  facet = NULL,

```

```
facet2 = NULL,  
group = NULL,  
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,  
facet_ncol = NULL,
```

```

facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.
<code>mapping</code>	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
<code>x_breaks</code> , <code>y_breaks</code> , <code>col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
<code>x_expand</code> , <code>y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits</code> , <code>y_expand_limits</code> , <code>col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels</code> , <code>y_labels</code> , <code>col_labels</code> , <code>facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits</code> , <code>y_limits</code> , <code>col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .

<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> ( <code>)</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> ( <code>)</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase: : to_sentence_case.

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins |>
  gg_qq(
    sample = body_mass_g,
    facet = species,
    coord = coord_cartesian(clip = "on"),
  ) +
  geom_qq_line(
    colour = blue,
  )
```

---

 gg\_quantile

*Quantile ggplot*


---

**Description**

Create an quantile ggplot with a wrapper around `ggplot2::ggplot() + geom_quantile()`.

**Usage**

```
gg_quantile(
  data = NULL,
  ...,
  stat = "quantile",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
```

```
xend = NULL,  
y = NULL,  
ymin = NULL,  
ymax = NULL,  
yend = NULL,  
z = NULL,  
col = NULL,  
facet = NULL,  
facet2 = NULL,  
group = NULL,  
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,
```

```

col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*</code> () function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*</code> () function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.
<code>mapping</code>	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
<code>x_breaks</code> , <code>y_breaks</code> , <code>col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
<code>x_expand</code> , <code>y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits</code> , <code>y_expand_limits</code> , <code>col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .



<code>x_labels, y_labels, col_labels, facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits, y_limits, col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>+ *_mode_*</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".

facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

**Value**

A ggplot object.

**Examples**

```
if (requireNamespace("quantreg", quietly = TRUE)) {
  library(ggplot2)
  library(palmerpenguins)

  set_blanket()

  penguins |>
    gg_quantile(
      x = flipper_length_mm,
      y = body_mass_g,
    )
}
```

---

 gg\_raster

*Raster ggplot*


---

**Description**

Create a raster ggplot with a wrapper around `ggplot2::ggplot() + geom_raster()`.

**Usage**

```
gg_raster(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
```

```
coord = ggplot2::coord_cartesian(clip = "off"),
mode = NULL,
x = NULL,
xmin = NULL,
xmax = NULL,
xend = NULL,
y = NULL,
ymin = NULL,
ymax = NULL,
yend = NULL,
z = NULL,
col = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_orientation = NULL,
x_position = "bottom",
x_label = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
```

```

col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to <code>within</code> a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*</code> () function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*</code> () function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.
<code>mapping</code>	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
<code>x_breaks</code> , <code>y_breaks</code> , <code>col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.

<code>x_expand, y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits, y_expand_limits, col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels, y_labels, col_labels, facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits, y_limits, col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_prefix</code> (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>+ *_mode_*</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_layout	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
facet_ncol, facet_nrow	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)

set_blanket()

faithfuld |>
  gg_raster(
    x = waiting,
    y = eruptions,
    col = density,
  )
```

---

gg\_rect

*Rect ggplot*


---

**Description**

Create a rect ggplot with a wrapper around `ggplot2::ggplot() + geom_rect()`.

**Usage**

```
gg_rect(  
  data = NULL,  
  ...,  
  stat = "identity",  
  position = "identity",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,  
  x_orientation = NULL,  
  x_position = "bottom",  
  x_label = NULL,  
  x_transform = NULL,  
  y_breaks = NULL,  
  y_expand = NULL,  
  y_expand_limits = NULL,  
  y_labels = NULL,  
  y_limits = NULL,  
  y_oob = scales::oob_keep,  
  y_orientation = NULL,  
  y_position = "left",  
  y_label = NULL,  
  y_transform = NULL,  
  col_breaks = NULL,  
  col_drop = FALSE,
```

```

col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.



mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use <code>+ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.
col_legend_rev	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
col_palette	A character vector of hex codes (or names) or a <code>scales::pal_*</code> () function.
col_palette_na	A hex code (or name) for the colour of NA values.
col_rescale	For a continuous variable, a <code>scales::rescale()</code> function.

<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + *_mode_*() may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
<code>facet_ncol</code> , <code>facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

## Value

A ggplot object.

## Examples

```
library(ggplot2)
library(dplyr)

set_blanket()

data.frame(
  x = rep(c(2, 5, 7, 9, 12), 2),
  y = rep(c(1, 2), each = 5),
  z = factor(c(rep(1:4, each = 2), 5, NA)),
  w = rep(diff(c(0, 4, 6, 8, 10, 14)), 2)
) |>
mutate(
  xmin = x - w / 2,
  xmax = x + w / 2,
  ymin = y,
  ymax = y + 1
```

```
) |>
gg_rect(
  xmin = xmin,
  xmax = xmax,
  ymin = ymin,
  ymax = ymax,
  col = z,
)
```

---

gg\_ribbon

*Ribbon ggplot*

---

### Description

Create a ribbon ggplot with a wrapper around `ggplot2::ggplot() + geom_ribbon()`

### Usage

```
gg_ribbon(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
```

```
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,  
facet_ncol = NULL,  
facet_nrow = NULL,  
facet_scales = "fixed",  
facet_space = "fixed",  
title = NULL,  
subtitle = NULL,  
caption = NULL,  
label_to_case = snakecase::to_sentence_case  
)
```

### Arguments

data            A data frame or tibble.

...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .

<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. <code>"log10"</code> ).
<code>col_drop, facet_drop</code>	For a discrete variable, <code>FALSE</code> or <code>TRUE</code> of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	<code>TRUE</code> or <code>FALSE</code> of whether to reverse the elements of a legend guide. Defaults to <code>FALSE</code> .
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, <code>TRUE</code> or <code>FALSE</code> of whether to colour in steps. Defaults to <code>FALSE</code> .
<code>facet_axes</code>	Whether to add interior axes and ticks with <code>"margins"</code> , <code>"all"</code> , <code>"all_x"</code> , or <code>"all_y"</code> . Sometimes <code>+ *_mode_*</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with <code>"margins"</code> , <code>"all"</code> , <code>"all_x"</code> , or <code>"all_y"</code> .
<code>facet_layout</code>	Whether the layout is to be <code>"wrap"</code> or <code>"grid"</code> . If <code>NULL</code> and a single facet (or <code>facet2</code> ) argument is provided, then defaults to <code>"wrap"</code> . If <code>NULL</code> and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to <code>"grid"</code> .
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of <code>"wrap"</code> .
<code>facet_scales</code>	Whether facet scales should be <code>"fixed"</code> across facets, <code>"free"</code> in both directions, or free in just one direction (i.e. <code>"free_x"</code> or <code>"free_y"</code> ). Defaults to <code>"fixed"</code> .
<code>facet_space</code>	When the facet layout is <code>"grid"</code> and facet scales are not <code>"fixed"</code> , whether facet space should be <code>"fixed"</code> across facets, <code>"free"</code> to be proportional in both directions, or free to be proportional in just one direction (i.e. <code>"free_x"</code> or <code>"free_y"</code> ). Defaults to <code>"fixed"</code> .
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A `ggplot` object.

**Examples**

```
library(ggplot2)
library(dplyr)

set_blanket()

data.frame(year = 1875:1972, level = as.vector(LakeHuron)) |>
  mutate(level_min = level - 1, level_max = level + 1) |>
  gg_ribbon(
    x = year,
    ymin = level_min,
    ymax = level_max,
    colour = NA,
    x_labels = \"(x) x\",
    y_label = \"Level\",
  ) +
  geom_line(mapping = aes(y = level))
```

---

gg\_rug

*Rug ggplot*

---

**Description**

Create a rug ggplot with a wrapper around `ggplot2::ggplot() + geom_rug()`.

**Usage**

```
gg_rug(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
```

```
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,  
facet_ncol = NULL,  
facet_nrow = NULL,  
facet_scales = "fixed",
```



```

facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .

<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> ( <code>)</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + <code>*_mode_*</code> ( <code>)</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
<code>facet_space</code>	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  gg_rug(
    x = flipper_length_mm,
    y = body_mass_g,
    col = sex,
  )
```

---

 gg\_segment

*Segment ggplot*


---

**Description**

Create a segment ggplot with a wrapper around `ggplot2::ggplot() + geom_segment()`.

**Usage**

```
gg_segment(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
```

```
ymin = NULL,  
ymax = NULL,  
yend = NULL,  
z = NULL,  
col = NULL,  
facet = NULL,  
facet2 = NULL,  
group = NULL,  
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,
```

```

facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .

<code>x_labels, y_labels, col_labels, facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits, y_limits, col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>+ *_mode_*</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".

facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

### Value

A ggplot object.

### Examples

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

data.frame(x1 = 2.62, x2 = 3.57, y1 = 21.0, y2 = 15.0) |>
  gg_segment(
    x = x1,
    xend = x2,
    y = y1,
    yend = y2,
  )
```

---

gg\_sf

*Sf ggplot*

---

### Description

Create a blank ggplot with a wrapper around `ggplot2::ggplot() + geom_sf()`.

### Usage

```
gg_sf(
  data = NULL,
  ...,
  stat = "sf",
```

```
position = "identity",
coord = ggplot2::coord_sf(clip = "off"),
mode = NULL,
x = NULL,
xmin = NULL,
xmax = NULL,
xend = NULL,
y = NULL,
ymin = NULL,
ymax = NULL,
yend = NULL,
z = NULL,
col = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_orientation = NULL,
x_position = "bottom",
x_label = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
```



```

col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*</code> () function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*</code> () function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.

<code>x_expand, y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits, y_expand_limits, col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels, y_labels, col_labels, facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits, y_limits, col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_prefix</code> (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>+ *_mode_*()</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_layout	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
facet_ncol, facet_nrow	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)

set_blanket()

if (requireNamespace("sf", quietly = TRUE)) {
  sf::st_read(system.file("shape/nc.shp", package = "sf")) |>
    gg_sf(
      col = AREA,
    )
}
```

---

gg\_smooth

*Smooth ggplot*


---

**Description**

Create a smooth ggplot with a wrapper around `ggplot2::ggplot() + geom_smooth()`.

**Usage**

```
gg_smooth(  
  data = NULL,  
  ...,  
  stat = "smooth",  
  position = "identity",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,  
  x_orientation = NULL,  
  x_position = "bottom",  
  x_label = NULL,  
  x_transform = NULL,  
  y_breaks = NULL,  
  y_expand = NULL,  
  y_expand_limits = NULL,  
  y_labels = NULL,  
  y_limits = NULL,  
  y_oob = scales::oob_keep,  
  y_orientation = NULL,  
  y_position = "left",  
  y_label = NULL,  
  y_transform = NULL,  
  col_breaks = NULL,  
  col_drop = FALSE,
```

```

col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in layer().
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*() function that outputs a ggproto Position subclass object (e.g. ggplot2::position_identity()).
coord	A coordinate system. A coord_*() function that outputs a constructed ggproto Coord subclass object (e.g. ggplot2::coord_cartesian()).
mode	A ggplot2 theme (e.g. light_mode_t(), grey_mode_r(), or dark_mode_r()). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of gg_*.
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.

mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use <code>+ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_*</code> prefix (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.
col_legend_rev	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
col_palette	A character vector of hex codes (or names) or a <code>scales::pal_*</code> () function.
col_palette_na	A hex code (or name) for the colour of NA values.
col_rescale	For a continuous variable, a <code>scales::rescale()</code> function.

col_steps	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
facet_axes	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + *_mode_*() may be needed.
facet_axis_labels	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_layout	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
facet_ncol, facet_nrow	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  tidyr::drop_na(sex) |>
  gg_smooth(
    x = flipper_length_mm,
    y = body_mass_g,
    col = sex,
    se = TRUE,
  )
```

---

`gg_step`*Step ggplot*

---

**Description**

Create a step plot with a wrapper around `ggplot2::ggplot() + geom_step()`.

**Usage**

```
gg_step(  
  data = NULL,  
  ...,  
  stat = "identity",  
  position = "identity",  
  coord = ggplot2::coord_cartesian(clip = "off"),  
  mode = NULL,  
  x = NULL,  
  xmin = NULL,  
  xmax = NULL,  
  xend = NULL,  
  y = NULL,  
  ymin = NULL,  
  ymax = NULL,  
  yend = NULL,  
  z = NULL,  
  col = NULL,  
  facet = NULL,  
  facet2 = NULL,  
  group = NULL,  
  subgroup = NULL,  
  label = NULL,  
  text = NULL,  
  sample = NULL,  
  mapping = NULL,  
  x_breaks = NULL,  
  x_expand = NULL,  
  x_expand_limits = NULL,  
  x_labels = NULL,  
  x_limits = NULL,  
  x_oob = scales::oob_keep,  
  x_orientation = NULL,  
  x_position = "bottom",  
  x_label = NULL,  
  x_transform = NULL,  
  y_breaks = NULL,  
  y_expand = NULL,  
  y_expand_limits = NULL,  
)
```



```

y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

data	A data frame or tibble.
...	Other arguments passed to within a params list in <code>layer()</code> .
stat	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*</code> () function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A <code>coord_*</code> () function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).

mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use + <code>ggplot2::labs(... = NULL)</code> for no title.
x_transform, y_transform, col_transform	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_</code> prefix (e.g. "log10").
col_drop, facet_drop	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
col_legend_ncol, col_legend_nrow	The number of columns and rows in a legend guide.

col_legend_rev	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
col_palette	A character vector of hex codes (or names) or a scales::pal_*() function.
col_palette_na	A hex code (or name) for the colour of NA values.
col_rescale	For a continuous variable, a scales::rescale() function.
col_steps	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
facet_axes	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes + *_mode_*() may be needed.
facet_axis_labels	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
facet_layout	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
facet_ncol, facet_nrow	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

### Value

A ggplot object.

### Examples

```
library(ggplot2)
library(dplyr)

set_blanket()

economics |>
  gg_step(
    x = date,
    y = unemploy,
    coord = ggplot2::coord_cartesian(clip = "on"),
```

```
x_limits = c(lubridate::ymd("2010-01-01"), lubridate::NA_Date_),
y_expand_limits = 0,
y_label = "Unemployment",
)
```

---

gg\_text

*Text ggplot*

---

## Description

Create a text plot with a wrapper around `ggplot2::ggplot() + geom_text()`.

## Usage

```
gg_text(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
  facet = NULL,
  facet2 = NULL,
  group = NULL,
  subgroup = NULL,
  label = NULL,
  text = NULL,
  sample = NULL,
  mapping = NULL,
  x_breaks = NULL,
  x_expand = NULL,
  x_expand_limits = NULL,
  x_labels = NULL,
  x_limits = NULL,
  x_oob = scales::oob_keep,
  x_orientation = NULL,
```

```

x_position = "bottom",
x_label = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

### Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a

	ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
position	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a position_*( <code>)</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
coord	A coordinate system. A coord_*( <code>)</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
mode	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of gg_*
x, xmin, xmax, xend, y, ymin, ymax, yend, z, col, facet, facet2, group, subgroup, label, text, sample	An unquoted aesthetic variable.
mapping	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. shape, linetype, linewidth, or size), but can also be used for delayed evaluation etc.
x_breaks, y_breaks, col_breaks	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.
x_expand, y_expand	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
x_expand_limits, y_expand_limits, col_expand_limits	For a continuous variable, any values that the limits should encompass (e.g. 0). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
x_labels, y_labels, col_labels, facet_labels	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*</code> ()), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
x_limits, y_limits, col_limits	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
x_oob, y_oob, col_oob	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
x_orientation, y_orientation	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
x_position, y_position	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
x_label, y_label, col_label	Label for the axis or legend title. Use + <code>ggplot2::labs(... = NULL)</code> for no title.

<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_prefix</code> (e.g. <code>"log10"</code> ).
<code>col_drop, facet_drop</code>	For a discrete variable, <code>FALSE</code> or <code>TRUE</code> of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	<code>TRUE</code> or <code>FALSE</code> of whether to reverse the elements of a legend guide. Defaults to <code>FALSE</code> .
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, <code>TRUE</code> or <code>FALSE</code> of whether to colour in steps. Defaults to <code>FALSE</code> .
<code>facet_axes</code>	Whether to add interior axes and ticks with <code>"margins"</code> , <code>"all"</code> , <code>"all_x"</code> , or <code>"all_y"</code> . Sometimes + <code>*_mode_*</code> () may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with <code>"margins"</code> , <code>"all"</code> , <code>"all_x"</code> , or <code>"all_y"</code> .
<code>facet_layout</code>	Whether the layout is to be <code>"wrap"</code> or <code>"grid"</code> . If <code>NULL</code> and a single facet (or <code>facet2</code> ) argument is provided, then defaults to <code>"wrap"</code> . If <code>NULL</code> and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to <code>"grid"</code> .
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of <code>"wrap"</code> .
<code>facet_scales</code>	Whether facet scales should be <code>"fixed"</code> across facets, <code>"free"</code> in both directions, or free in just one direction (i.e. <code>"free_x"</code> or <code>"free_y"</code> ). Defaults to <code>"fixed"</code> .
<code>facet_space</code>	When the facet layout is <code>"grid"</code> and facet scales are not <code>"fixed"</code> , whether facet space should be <code>"fixed"</code> across facets, <code>"free"</code> to be proportional in both directions, or free to be proportional in just one direction (i.e. <code>"free_x"</code> or <code>"free_y"</code> ). Defaults to <code>"fixed"</code> .
<code>title</code>	Title string.
<code>subtitle</code>	Subtitle string.
<code>caption</code>	Caption title string.
<code>label_to_case</code>	A function to format the default <code>x_label</code> , <code>y_label</code> and <code>col_label</code> of unlabelled variables. Defaults to <code>snakecase::to_sentence_case</code> .

**Value**

A ggplot object.

### Examples

```
library(ggplot2)
library(dplyr)

set_blanket()

bind_rows(
  mtcars |> slice_min(order_by = mpg),
  mtcars |> slice_max(order_by = mpg)
) |>
  tibble::rownames_to_column("model") |>
  gg_text(
    x = model,
    y = mpg,
    col = mpg,
    label = model,
    y_expand_limits = 0,
    y_label = "Miles per gallon",
    col_palette = c(orange, "white", teal),
  )
```

---

gg\_tile

*Tile ggplot*

---

### Description

Create a tile plot with a wrapper around `ggplot2::ggplot() + geom_tile()`.

### Usage

```
gg_tile(
  data = NULL,
  ...,
  stat = "identity",
  position = "identity",
  coord = ggplot2::coord_cartesian(clip = "off"),
  mode = NULL,
  x = NULL,
  xmin = NULL,
  xmax = NULL,
  xend = NULL,
  y = NULL,
  ymin = NULL,
  ymax = NULL,
  yend = NULL,
  z = NULL,
  col = NULL,
```



```
facet = NULL,  
facet2 = NULL,  
group = NULL,  
subgroup = NULL,  
label = NULL,  
text = NULL,  
sample = NULL,  
mapping = NULL,  
x_breaks = NULL,  
x_expand = NULL,  
x_expand_limits = NULL,  
x_labels = NULL,  
x_limits = NULL,  
x_oob = scales::oob_keep,  
x_orientation = NULL,  
x_position = "bottom",  
x_label = NULL,  
x_transform = NULL,  
y_breaks = NULL,  
y_expand = NULL,  
y_expand_limits = NULL,  
y_labels = NULL,  
y_limits = NULL,  
y_oob = scales::oob_keep,  
y_orientation = NULL,  
y_position = "left",  
y_label = NULL,  
y_transform = NULL,  
col_breaks = NULL,  
col_drop = FALSE,  
col_expand_limits = NULL,  
col_labels = NULL,  
col_legend_ncol = NULL,  
col_legend_nrow = NULL,  
col_legend_rev = FALSE,  
col_limits = NULL,  
col_oob = scales::oob_keep,  
col_palette = NULL,  
col_palette_na = NULL,  
col_rescale = scales::rescale(),  
col_steps = FALSE,  
col_label = NULL,  
col_transform = NULL,  
facet_axes = NULL,  
facet_axis_labels = "margins",  
facet_drop = FALSE,  
facet_labels = NULL,  
facet_layout = NULL,
```

```

facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to within a params list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*()</code> function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*()</code> function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.
<code>mapping</code>	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
<code>x_breaks</code> , <code>y_breaks</code> , <code>col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*()</code> ), or a vector of breaks.
<code>x_expand</code> , <code>y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits</code> , <code>y_expand_limits</code> , <code>col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels</code> , <code>y_labels</code> , <code>col_labels</code> , <code>facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).

<code>x_limits, y_limits, col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_prefix</code> (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*</code> () function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>+*_mode_*</code> () may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".
<code>facet_layout</code>	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or <code>facet2</code> ) argument is provided, then defaults to "wrap". If NULL and both <code>facet</code> and <code>facet2</code> arguments are provided, defaults to "grid".
<code>facet_ncol, facet_nrow</code>	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
<code>facet_scales</code>	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".

facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

**Value**

A ggplot object.

**Examples**

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  group_by(species, sex) |>
  summarise(across(flipper_length_mm, \(x) mean(x, na.rm = TRUE))) |>
  gg_tile(
    x = sex,
    y = species,
    col = flipper_length_mm,
  )
```

---

gg\_violin

*Violin ggplot*


---

**Description**

Create a violin plot with a wrapper around `ggplot2::ggplot() + geom_violin()`.

**Usage**

```
gg_violin(
  data = NULL,
  ...,
  stat = "ydensity",
  position = "dodge",
```

```
coord = ggplot2::coord_cartesian(clip = "off"),
mode = NULL,
x = NULL,
xmin = NULL,
xmax = NULL,
xend = NULL,
y = NULL,
ymin = NULL,
ymax = NULL,
yend = NULL,
z = NULL,
col = NULL,
facet = NULL,
facet2 = NULL,
group = NULL,
subgroup = NULL,
label = NULL,
text = NULL,
sample = NULL,
mapping = NULL,
x_breaks = NULL,
x_expand = NULL,
x_expand_limits = NULL,
x_labels = NULL,
x_limits = NULL,
x_oob = scales::oob_keep,
x_orientation = NULL,
x_position = "bottom",
x_label = NULL,
x_transform = NULL,
y_breaks = NULL,
y_expand = NULL,
y_expand_limits = NULL,
y_labels = NULL,
y_limits = NULL,
y_oob = scales::oob_keep,
y_orientation = NULL,
y_position = "left",
y_label = NULL,
y_transform = NULL,
col_breaks = NULL,
col_drop = FALSE,
col_expand_limits = NULL,
col_labels = NULL,
col_legend_ncol = NULL,
col_legend_nrow = NULL,
col_legend_rev = FALSE,
col_limits = NULL,
```

```

col_oob = scales::oob_keep,
col_palette = NULL,
col_palette_na = NULL,
col_rescale = scales::rescale(),
col_steps = FALSE,
col_label = NULL,
col_transform = NULL,
facet_axes = NULL,
facet_axis_labels = "margins",
facet_drop = FALSE,
facet_labels = NULL,
facet_layout = NULL,
facet_ncol = NULL,
facet_nrow = NULL,
facet_scales = "fixed",
facet_space = "fixed",
title = NULL,
subtitle = NULL,
caption = NULL,
label_to_case = snakecase::to_sentence_case
)

```

## Arguments

<code>data</code>	A data frame or tibble.
<code>...</code>	Other arguments passed to <code>within</code> a <code>params</code> list in <code>layer()</code> .
<code>stat</code>	A statistical transformation to use on the data. A snakecase character string of a ggproto Stat subclass object minus the Stat prefix (e.g. "identity").
<code>position</code>	A position adjustment. A snakecase character string of a ggproto Position subclass object minus the Position prefix (e.g. "identity"), or a <code>position_*</code> () function that outputs a ggproto Position subclass object (e.g. <code>ggplot2::position_identity()</code> ).
<code>coord</code>	A coordinate system. A <code>coord_*</code> () function that outputs a constructed ggproto Coord subclass object (e.g. <code>ggplot2::coord_cartesian()</code> ).
<code>mode</code>	A ggplot2 theme (e.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> ). This argument adds the theme with side-effects of removing relevant axis line/ticks and gridlines per the orientation. To avoid these side-effects, + the theme on to the output of <code>gg_*</code> .
<code>x</code> , <code>xmin</code> , <code>xmax</code> , <code>xend</code> , <code>y</code> , <code>ymin</code> , <code>ymax</code> , <code>yend</code> , <code>z</code> , <code>col</code> , <code>facet</code> , <code>facet2</code> , <code>group</code> , <code>subgroup</code> , <code>label</code> , <code>text</code> , <code>sample</code>	An unquoted aesthetic variable.
<code>mapping</code>	A set of additional aesthetic mappings in <code>ggplot2::aes()</code> . Intended primarily for non-supported aesthetics (e.g. <code>shape</code> , <code>linetype</code> , <code>linewidth</code> , or <code>size</code> ), but can also be used for delayed evaluation etc.
<code>x_breaks</code> , <code>y_breaks</code> , <code>col_breaks</code>	A <code>scales::breaks_*</code> function (e.g. <code>scales::breaks_*</code> ()), or a vector of breaks.

<code>x_expand, y_expand</code>	Padding to the limits with the <code>ggplot2::expansion()</code> function, or a vector of length 2 (e.g. <code>c(0, 0)</code> ).
<code>x_expand_limits, y_expand_limits, col_expand_limits</code>	For a continuous variable, any values that the limits should encompass (e.g. <code>0</code> ). For a discrete scale, manipulate the data instead with <code>forcats::fct_expand</code> .
<code>x_labels, y_labels, col_labels, facet_labels</code>	A function that takes the breaks as inputs (e.g. <code>\(x) stringr::str_to_sentence(x)</code> or <code>scales::label_*()</code> ), or a vector of labels. (Note this must be named for <code>facet_labels</code> ).
<code>x_limits, y_limits, col_limits</code>	For a continuous scale, a vector of length 2 to determine the limits of the scale. For a discrete scale, manipulate the data instead with <code>factor</code> , <code>forcats::fct_expand</code> or <code>forcats::fct_drop</code> .
<code>x_oob, y_oob, col_oob</code>	For a continuous scale, a <code>scales::oob_*</code> function of how to handle values outside of limits. Defaults to <code>scales::oob_keep</code> .
<code>x_orientation, y_orientation</code>	TRUE or FALSE of whether the mode and scales orientation should be to x or y. Note, these arguments do not affect the orientation of the layer itself.
<code>x_position, y_position</code>	The position of the axis (i.e. "left", "right", "bottom" or "top"). If using <code>y_position = "top"</code> with a <code>*_mode_*</code> theme, add <code>caption = ""</code> or <code>caption = "\n"</code> .
<code>x_label, y_label, col_label</code>	Label for the axis or legend title. Use <code>+ ggplot2::labs(... = NULL)</code> for no title.
<code>x_transform, y_transform, col_transform</code>	For a continuous scale, a transformation object (e.g. <code>scales::transform_log10()</code> ) or character string of this minus the <code>transform_prefix</code> (e.g. "log10").
<code>col_drop, facet_drop</code>	For a discrete variable, FALSE or TRUE of whether to drop unused levels.
<code>col_legend_ncol, col_legend_nrow</code>	The number of columns and rows in a legend guide.
<code>col_legend_rev</code>	TRUE or FALSE of whether to reverse the elements of a legend guide. Defaults to FALSE.
<code>col_palette</code>	A character vector of hex codes (or names) or a <code>scales::pal_*()</code> function.
<code>col_palette_na</code>	A hex code (or name) for the colour of NA values.
<code>col_rescale</code>	For a continuous variable, a <code>scales::rescale()</code> function.
<code>col_steps</code>	For a continuous variable, TRUE or FALSE of whether to colour in steps. Defaults to FALSE.
<code>facet_axes</code>	Whether to add interior axes and ticks with "margins", "all", "all_x", or "all_y". Sometimes <code>+ *_mode_*()</code> may be needed.
<code>facet_axis_labels</code>	Whether to add interior axis labels with "margins", "all", "all_x", or "all_y".

facet_layout	Whether the layout is to be "wrap" or "grid". If NULL and a single facet (or facet2) argument is provided, then defaults to "wrap". If NULL and both facet and facet2 arguments are provided, defaults to "grid".
facet_ncol, facet_nrow	The number of columns and rows of facet panels. Only applies to a facet layout of "wrap".
facet_scales	Whether facet scales should be "fixed" across facets, "free" in both directions, or free in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
facet_space	When the facet layout is "grid" and facet scales are not "fixed", whether facet space should be "fixed" across facets, "free" to be proportional in both directions, or free to be proportional in just one direction (i.e. "free_x" or "free_y"). Defaults to "fixed".
title	Title string.
subtitle	Subtitle string.
caption	Caption title string.
label_to_case	A function to format the default x_label, y_label and col_label of unlabelled variables. Defaults to snakecase::to_sentence_case.

## Value

A ggplot object.

## Examples

```
library(ggplot2)
library(dplyr)
library(palmerpenguins)

set_blanket()

penguins |>
  tidyr::drop_na(sex) |>
  mutate(across(sex, \(x) stringr::str_to_sentence(x))) |>
  gg_violin(
    x = sex,
    y = body_mass_g,
    col = sex,
    facet = species,
    mode = light_mode_b(),
  )
```



---

grey	<i>A grey colour</i>
------	----------------------

---

**Description**

A grey colour.

**Usage**

```
grey
```

**Value**

A character vector.

**Examples**

```
scales::show_col(grey)
```

---

grey_mode_r	<i>Grey mode theme family</i>
-------------	-------------------------------

---

**Description**

A grey mode family of functions:

- `grey_mode_r()` with legend on right
- `grey_mode_t()` with legend on top
- `grey_mode_b()` with legend on bottom
- `grey_mode_n()` with no legend

**Usage**

```
grey_mode_r(
  ...,
  base_size = 11,
  base_family = "",
  base_colour = "#121B24FF",
  axis_line_colour = "#121B24FF",
  axis_line_linewidth = 0.33,
  axis_ticks_colour = axis_line_colour,
  axis_ticks_linewidth = axis_line_linewidth,
  axis_ticks_length_x = grid::unit(base_size/3, "pt"),
  axis_ticks_length_y = grid::unit(base_size/4, "pt"),
  panel_grid_colour = "#F6F8FAFF",
```

```
panel_grid_linewidth = 1.33,  
panel_background_fill = "#FCDFE9",  
plot_background_fill = "#F6F8FA",  
legend_axis_line_colour = plot_background_fill,  
legend_axis_line_linewidth = 0.33,  
legend_background_fill = plot_background_fill,  
legend_key_fill = plot_background_fill,  
legend_ticks_colour = legend_axis_line_colour,  
legend_ticks_linewidth = legend_axis_line_linewidth,  
legend_ticks_length = ggplot2::rel(c(0.175, 0)),  
orientation = NULL  
)  
  
grey_mode_t(  
  ...,  
  base_size = 11,  
  base_family = "",  
  base_colour = "#121B24",  
  axis_line_colour = "#121B24",  
  axis_line_linewidth = 0.33,  
  axis_ticks_colour = axis_line_colour,  
  axis_ticks_linewidth = axis_line_linewidth,  
  axis_ticks_length_x = grid::unit(base_size/3, "pt"),  
  axis_ticks_length_y = grid::unit(base_size/4, "pt"),  
  panel_grid_colour = "#F6F8FA",  
  panel_grid_linewidth = 1.33,  
  panel_background_fill = "#FCDFE9",  
  plot_background_fill = "#F6F8FA",  
  legend_axis_line_colour = plot_background_fill,  
  legend_axis_line_linewidth = 0.33,  
  legend_background_fill = plot_background_fill,  
  legend_key_fill = plot_background_fill,  
  legend_ticks_colour = legend_axis_line_colour,  
  legend_ticks_linewidth = legend_axis_line_linewidth,  
  legend_ticks_length = ggplot2::rel(c(0.175, 0)),  
  orientation = NULL  
)  
  
grey_mode_b(  
  ...,  
  base_size = 11,  
  base_family = "",  
  base_colour = "#121B24",  
  axis_line_colour = "#121B24",  
  axis_line_linewidth = 0.33,  
  axis_ticks_colour = axis_line_colour,  
  axis_ticks_linewidth = axis_line_linewidth,  
  axis_ticks_length_x = grid::unit(base_size/3, "pt"),
```

```

axis_ticks_length_y = grid::unit(base_size/4, "pt"),
panel_grid_colour = "#F6F8FAFF",
panel_grid_linewidth = 1.33,
panel_background_fill = "#FCDFEFF",
plot_background_fill = "#F6F8FAFF",
legend_axis_line_colour = plot_background_fill,
legend_axis_line_linewidth = 0.33,
legend_background_fill = plot_background_fill,
legend_key_fill = plot_background_fill,
legend_ticks_colour = legend_axis_line_colour,
legend_ticks_linewidth = legend_axis_line_linewidth,
legend_ticks_length = ggplot2::rel(c(0.175, 0)),
orientation = NULL
)

grey_mode_n(
  ...,
  base_size = 11,
  base_family = "",
  base_colour = "#121B24FF",
  axis_line_colour = "#121B24FF",
  axis_line_linewidth = 0.33,
  axis_ticks_colour = axis_line_colour,
  axis_ticks_linewidth = axis_line_linewidth,
  axis_ticks_length_x = grid::unit(base_size/3, "pt"),
  axis_ticks_length_y = grid::unit(base_size/4, "pt"),
  panel_grid_colour = "#F6F8FAFF",
  panel_grid_linewidth = 1.33,
  panel_background_fill = "#FCDFEFF",
  plot_background_fill = "#F6F8FAFF",
  legend_axis_line_colour = plot_background_fill,
  legend_axis_line_linewidth = 0.33,
  legend_background_fill = plot_background_fill,
  legend_key_fill = plot_background_fill,
  legend_ticks_colour = legend_axis_line_colour,
  legend_ticks_linewidth = legend_axis_line_linewidth,
  legend_ticks_length = ggplot2::rel(c(0.175, 0)),
  orientation = NULL
)

```

### Arguments

...	Provided to force user argument naming etc.
base_size	The base size of the text theme element. Defaults to 11.
base_family	The base family of the text theme element. Defaults to "".
base_colour	The base colour of the text theme element.
axis_line_colour	The colour of the axis.line theme element.

<code>axis_line_linewidth</code>	The linewidth of the <code>axis.line</code> theme element.
<code>axis_ticks_colour</code>	The colour of the <code>axis.ticks</code> theme element.
<code>axis_ticks_linewidth</code>	The linewidth of the <code>axis.ticks</code> theme element.
<code>axis_ticks_length_x</code>	The length of the <code>axis.ticks.length.x</code> theme element.
<code>axis_ticks_length_y</code>	The length of the <code>axis.ticks.length.y</code> theme element.
<code>panel_grid_colour</code>	The colour of the <code>panel.grid</code> theme element.
<code>panel_grid_linewidth</code>	The linewidth of the <code>panel.grid</code> theme element.
<code>panel_background_fill</code>	The fill (and colour) of the <code>panel.background</code> theme element.
<code>plot_background_fill</code>	The fill (and colour) of the <code>plot.background</code> theme element.
<code>legend_axis_line_colour</code>	The colour of the <code>legend.axis.line</code> theme element.
<code>legend_axis_line_linewidth</code>	The linewidth of the <code>legend.axis.line</code> theme element.
<code>legend_background_fill</code>	The fill (and colour) of the <code>legend.background</code> theme element.
<code>legend_key_fill</code>	The fill (and colour) of the <code>legend.key</code> theme element.
<code>legend_ticks_colour</code>	The colour of the <code>legend.ticks</code> theme element.
<code>legend_ticks_linewidth</code>	The linewidth of the <code>legend.ticks</code> theme element.
<code>legend_ticks_length</code>	The <code>legend.ticks.length</code> theme element.
<code>orientation</code>	The orientation of the plot. Either "x" or "y". Defaults to NULL. Not intended for use with the <code>mode</code> argument of <code>gg_*</code> functions.

**Value**

A ggplot theme.

**Examples**

```
library(palmerpenguins)
library(ggplot2)

set_blanket()

penguins |>
```

```
gg_point(  
  x = flipper_length_mm,  
  y = body_mass_g,  
  col = species,  
  mode = grey_mode_r()  
)  
  
penguins |>  
gg_point(  
  x = flipper_length_mm,  
  y = body_mass_g,  
  col = species,  
  mode = grey_mode_t()  
)  
  
penguins |>  
gg_point(  
  x = flipper_length_mm,  
  y = body_mass_g,  
  col = species,  
  mode = grey_mode_b()  
)  
  
penguins |>  
gg_point(  
  x = flipper_length_mm,  
  y = body_mass_g,  
  col = species,  
  mode = grey_mode_n()  
)
```

---

jumble

*The jumble palette*

---

### **Description**

A discrete palette that is relatively colour-blind safe.

### **Usage**

jumble

teal

orange

navy

red  
pink  
purple

**Value**

A character vector.

**Examples**

```
colorspace::swatchplot(c(jumble, grey), cvd = TRUE)
```

---

lightness

*Mode colour and linewidth defaults*

---

**Description**

lightness, greyness and darkness are vectors of 3 colours used in the \*\_mode\_\* themes for the for the text, axis.line (and axis.ticks), panel.grid, panel.background and plot.background etc.

linewidthness is a vector of 2 integers used in the \*\_mode\_\* themes for the linewidth of the axis.line (axis.ticks and legend.ticks) and panel.grid theme elements.

**Usage**

lightness  
greyness  
darkness  
linewidthness

**Value**

A character vector.

**Examples**

```
scales::show_col(c(lightness, greyness, darkness), ncol = 3)
```

---

light_mode_r	<i>Light mode theme family</i>
--------------	--------------------------------

---

## Description

A dark mode family of functions:

- `light_mode_r()` with legend on right
- `light_mode_t()` with legend on top
- `light_mode_b()` with legend on bottom
- `light_mode_n()` with no legend

## Usage

```
light_mode_r(  
  ...,  
  base_size = 11,  
  base_family = "",  
  base_colour = "#121B24FF",  
  axis_line_colour = "#121B24FF",  
  axis_line_linewidth = 0.33,  
  axis_ticks_colour = axis_line_colour,  
  axis_ticks_linewidth = axis_line_linewidth,  
  axis_ticks_length_x = grid::unit(base_size/3, "pt"),  
  axis_ticks_length_y = grid::unit(base_size/4, "pt"),  
  panel_grid_colour = "#F6F8FAFF",  
  panel_grid_linewidth = 1.33,  
  panel_background_fill = "#FFFFFF",  
  plot_background_fill = "#FFFFFF",  
  legend_axis_line_colour = plot_background_fill,  
  legend_axis_line_linewidth = 0.33,  
  legend_background_fill = plot_background_fill,  
  legend_key_fill = plot_background_fill,  
  legend_ticks_colour = legend_axis_line_colour,  
  legend_ticks_linewidth = legend_axis_line_linewidth,  
  legend_ticks_length = ggplot2::rel(c(0.175, 0)),  
  orientation = NULL  
)
```

```
light_mode_t(  
  ...,  
  base_size = 11,  
  base_family = "",  
  base_colour = "#121B24FF",  
  axis_line_colour = "#121B24FF",  
  axis_line_linewidth = 0.33,
```

```

axis_ticks_colour = axis_line_colour,
axis_ticks_linewidth = axis_line_linewidth,
axis_ticks_length_x = grid::unit(base_size/3, "pt"),
axis_ticks_length_y = grid::unit(base_size/4, "pt"),
panel_grid_colour = "#F6F8FAFF",
panel_grid_linewidth = 1.33,
panel_background_fill = "#FFFFFF",
plot_background_fill = "#FFFFFF",
legend_axis_line_colour = plot_background_fill,
legend_axis_line_linewidth = 0.33,
legend_background_fill = plot_background_fill,
legend_key_fill = plot_background_fill,
legend_ticks_colour = legend_axis_line_colour,
legend_ticks_linewidth = legend_axis_line_linewidth,
legend_ticks_length = ggplot2::rel(c(0.175, 0)),
orientation = NULL
)

```

```

light_mode_b(
  ...,
  base_size = 11,
  base_family = "",
  base_colour = "#121B24FF",
  axis_line_colour = "#121B24FF",
  axis_line_linewidth = 0.33,
  axis_ticks_colour = axis_line_colour,
  axis_ticks_linewidth = axis_line_linewidth,
  axis_ticks_length_x = grid::unit(base_size/3, "pt"),
  axis_ticks_length_y = grid::unit(base_size/4, "pt"),
  panel_grid_colour = "#F6F8FAFF",
  panel_grid_linewidth = 1.33,
  panel_background_fill = "#FFFFFF",
  plot_background_fill = "#FFFFFF",
  legend_axis_line_colour = plot_background_fill,
  legend_axis_line_linewidth = 0.33,
  legend_background_fill = plot_background_fill,
  legend_key_fill = plot_background_fill,
  legend_ticks_colour = legend_axis_line_colour,
  legend_ticks_linewidth = legend_axis_line_linewidth,
  legend_ticks_length = ggplot2::rel(c(0.175, 0)),
  orientation = NULL
)

```

```

light_mode_n(
  ...,
  base_size = 11,
  base_family = "",
  base_colour = "#121B24FF",

```



```

axis_line_colour = "#121B24FF",
axis_line_linewidth = 0.33,
axis_ticks_colour = axis_line_colour,
axis_ticks_linewidth = axis_line_linewidth,
axis_ticks_length_x = grid::unit(base_size/3, "pt"),
axis_ticks_length_y = grid::unit(base_size/4, "pt"),
panel_grid_colour = "#F6F8FAFF",
panel_grid_linewidth = 1.33,
panel_background_fill = "#FFFFFF",
plot_background_fill = "#FFFFFF",
legend_axis_line_colour = plot_background_fill,
legend_axis_line_linewidth = 0.33,
legend_background_fill = plot_background_fill,
legend_key_fill = plot_background_fill,
legend_ticks_colour = legend_axis_line_colour,
legend_ticks_linewidth = legend_axis_line_linewidth,
legend_ticks_length = ggplot2::rel(c(0.175, 0)),
orientation = NULL
)

```

### Arguments

...	Provided to force user argument naming etc.
base_size	The base size of the text theme element. Defaults to 11.
base_family	The base family of the text theme element. Defaults to "".
base_colour	The base colour of the text theme element.
axis_line_colour	The colour of the axis.line theme element.
axis_line_linewidth	The linewidth of the axis.line theme element.
axis_ticks_colour	The colour of the axis.ticks theme element.
axis_ticks_linewidth	The linewidth of the axis.ticks theme element.
axis_ticks_length_x	The length of the axis.ticks.length.x theme element.
axis_ticks_length_y	The length of the axis.ticks.length.y theme element.
panel_grid_colour	The colour of the panel.grid theme element.
panel_grid_linewidth	The linewidth of the panel.grid theme element.
panel_background_fill	The fill (and colour) of the panel.background theme element.
plot_background_fill	The fill (and colour) of the plot.background theme element.

<code>legend_axis_line_colour</code>	The colour of the <code>legend.axis.line</code> theme element.
<code>legend_axis_line_linewidth</code>	The linewidth of the <code>legend.axis.line</code> theme element.
<code>legend_background_fill</code>	The fill (and colour) of the <code>legend.background</code> theme element.
<code>legend_key_fill</code>	The fill (and colour) of the <code>legend.key</code> theme element.
<code>legend_ticks_colour</code>	The colour of the <code>legend.ticks</code> theme element.
<code>legend_ticks_linewidth</code>	The linewidth of the <code>legend.ticks</code> theme element.
<code>legend_ticks_length</code>	The <code>legend.ticks.length</code> theme element.
<code>orientation</code>	The orientation of the plot. Either "x" or "y". Defaults to NULL. Not intended for use with the <code>mode</code> argument of <code>gg_*</code> functions.

**Value**

A ggplot theme.

**Examples**

```
library(palmerpenguins)
library(ggplot2)

set_blanket()

penguins |>
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = species,
    mode = light_mode_r()
  )

penguins |>
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = species,
    mode = light_mode_t()
  )

penguins |>
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = species,
    mode = light_mode_b()
  )
```

```

)

penguins |>
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    col = species,
    mode = light_mode_n()
  )

```

---

set\_blanket

*Set a style*


---

## Description

Set a style by setting a mode, a series of geom and annotate aesthetic defaults, and a col\_palette for discrete and continuous scales.

## Usage

```

set_blanket(
  ...,
  mode = light_mode_r(),
  geom_colour = "#357BA2FF",
  annotate_colour = "#121B24FF",
  annotate_linewidth = 0.33,
  annotate_family = "",
  annotate_size = 3.88,
  col_palette_d = jumble,
  col_palette_na_d = "#CDC5BFFF",
  col_palette_c = viridisLite::mako(n = 9, direction = -1),
  col_palette_na_c = "#988F88FF",
  col_palette_o = scales::pal_viridis(option = "G", direction = -1),
  col_palette_na_o = "#988F88FF",
  theme = light_mode_r(orientation = "x")
)

```

## Arguments

...	Provided to force user argument naming etc.
mode	A *_mode_*. E.g. <code>light_mode_t()</code> , <code>grey_mode_r()</code> , or <code>dark_mode_r()</code> .
geom_colour	A hex colour (and fill) for most geoms. Fill inherits from this colour. Defaults to blue (i.e. #357BA2FF).
annotate_colour	A hex colour (and fill) for other geoms commonly used for annotation (i.e. *_hline/*_vline/*_abline and *_curve). Defaults to "#121b24" (i.e. <code>lightness[1]</code> ).

`annotate_linewidth` A linewidth for `*_hline/*_vline/*_abline` and `*_curve`. Defaults to 0.33.  
`annotate_family` A family for `*_text` and `*_label`. Defaults to `""`.  
`annotate_size` A size for `*_text` and `*_label`. Defaults to 3.88.  
`col_palette_d` For a discrete scale, a character vector of hex codes (or names) for the `col_palette`.  
`col_palette_na_d` For a discrete scale, a hex code or name for the `col_palette_na`.  
`col_palette_c` For a continuous scale, a character vector of hex codes (or names) for the `col_palette`.  
`col_palette_na_c` For a continuous scale, a hex code or name for the `col_palette_na`.  
`col_palette_o` For an ordinal scale, a `scales::pal_*()` function for the `col_palette`.  
`col_palette_na_o` For an ordinal scale, a hex code or name for the `col_palette_na`.  
`theme` A `ggplot2` theme to be +ed on unmodified to `gg_*` functions. Note, `mode` takes precedence, unless `mode = NULL`.

### Value

A globally set style.

### Examples

```

library(ggplot2)
library(ggblanket)
library(palmerpenguins)

set_blanket(
  mode = dark_mode_r(),
  geom_colour = orange,
  annotate_colour = darkness[1],
)

penguins |>
  gg_point(
    x = flipper_length_mm,
    y = body_mass_g,
    x_breaks = scales::breaks_pretty(3),
  ) +
  geom_vline(xintercept = 200) +
  annotate("text", x = I(0.25), y = I(0.75), label = "Here")

penguins |>
  gg_histogram(
    x = flipper_length_mm,
    x_breaks = scales::breaks_pretty(3),
  ) +
  geom_vline(xintercept = 200) +
  annotate("text", x = I(0.75), y = I(0.75), label = "Here")

```

# Index

- \* **datasets**
  - blue, 4
  - grey, 169
  - jumble, 173
  - lightness, 174
- aes\_contrast, 3
- blue, 4
- dark\_mode\_b (dark\_mode\_r), 5
- dark\_mode\_n (dark\_mode\_r), 5
- dark\_mode\_r, 5
- dark\_mode\_r(), 11, 15, 19, 23, 28, 32, 36, 40, 44, 49, 53, 57, 61, 65, 70, 74, 78, 82, 86, 90, 95, 99, 103, 107, 111, 116, 120, 124, 128, 133, 137, 141, 145, 149, 154, 158, 162, 166, 179
- dark\_mode\_t (dark\_mode\_r), 5
- darkness (lightness), 174
- geom\_area(), 9
- geom\_bar(), 13
- geom\_bin\_2d(), 17
- geom\_blank(), 21
- geom\_boxplot(), 26
- geom\_col(), 30
- geom\_contour(), 34
- geom\_contour\_filled(), 38
- geom\_crossbar(), 42
- geom\_density(), 47
- geom\_density\_2d(), 51
- geom\_density\_2d\_filled(), 55
- geom\_errorbar(), 59
- geom\_freqpoly(), 63
- geom\_function(), 68
- geom\_hex(), 72
- geom\_histogram(), 76
- geom\_jitter(), 80
- geom\_label(), 84
- geom\_line(), 88
- geom\_linerange(), 93
- geom\_path(), 97
- geom\_point(), 101
- geom\_pointrange(), 105
- geom\_polygon(), 110
- geom\_qq(), 114
- geom\_quantile(), 118
- geom\_raster(), 122
- geom\_rect(), 126
- geom\_ribbon(), 131
- geom\_rug(), 135
- geom\_segment(), 139
- geom\_sf(), 143
- geom\_smooth(), 147
- geom\_step(), 152
- geom\_text(), 156
- geom\_tile(), 160
- geom\_violin(), 164
- gg\_area, 9
- gg\_bar, 13
- gg\_bin\_2d, 17
- gg\_blanket, 21
- gg\_boxplot, 26
- gg\_col, 30
- gg\_contour, 34
- gg\_contour\_filled, 38
- gg\_crossbar, 42
- gg\_density, 47
- gg\_density\_2d, 51
- gg\_density\_2d\_filled, 55
- gg\_errorbar, 59
- gg\_freqpoly, 63
- gg\_function, 68
- gg\_hex, 72
- gg\_histogram, 76
- gg\_jitter, 80
- gg\_label, 84
- gg\_line, 88

gg\_linerange, 93  
 gg\_path, 97  
 gg\_point, 101  
 gg\_pointrange, 105  
 gg\_polygon, 109  
 gg\_qq, 114  
 gg\_quantile, 118  
 gg\_raster, 122  
 gg\_rect, 126  
 gg\_ribbon, 131  
 gg\_rug, 135  
 gg\_segment, 139  
 gg\_sf, 143  
 gg\_smooth, 147  
 gg\_step, 152  
 gg\_text, 156  
 gg\_tile, 160  
 gg\_violin, 164  
 ggplot2::aes, 3  
 ggplot2::aes(), 11, 15, 19, 23, 28, 32, 36, 40, 44, 49, 53, 57, 61, 65, 70, 74, 78, 82, 86, 91, 95, 99, 103, 107, 112, 116, 120, 124, 129, 133, 137, 141, 145, 150, 154, 158, 162, 166  
 ggplot2::coord\_cartesian(), 11, 15, 19, 23, 27, 32, 36, 40, 44, 48, 53, 57, 61, 65, 69, 73, 78, 82, 86, 90, 94, 99, 103, 107, 111, 116, 120, 124, 128, 133, 137, 141, 145, 149, 153, 158, 162, 166  
 ggplot2::expansion(), 11, 15, 19, 24, 28, 32, 36, 40, 45, 49, 53, 57, 61, 66, 70, 74, 78, 82, 86, 91, 95, 99, 103, 107, 112, 116, 120, 125, 129, 133, 137, 141, 146, 150, 154, 158, 162, 167  
 ggplot2::ggplot(), 9, 13, 17, 21, 26, 30, 34, 38, 42, 47, 51, 55, 59, 63, 68, 72, 76, 80, 84, 88, 93, 97, 101, 105, 110, 114, 118, 122, 126, 131, 135, 139, 143, 147, 152, 156, 160, 164  
 grey, 169  
 grey\_mode\_b(grey\_mode\_r), 169  
 grey\_mode\_n(grey\_mode\_r), 169  
 grey\_mode\_r, 169  
 grey\_mode\_r(), 11, 15, 19, 23, 28, 32, 36, 40, 44, 49, 53, 57, 61, 65, 70, 74, 78, 82, 86, 90, 95, 99, 103, 107, 111, 116, 120, 124, 128, 133, 137, 141, 145, 149, 154, 158, 162, 166, 179  
 grey\_mode\_t(grey\_mode\_r), 169  
 greyness(lightness), 174  
 jumble, 173  
 light\_mode\_b(light\_mode\_r), 175  
 light\_mode\_n(light\_mode\_r), 175  
 light\_mode\_r, 175  
 light\_mode\_t(light\_mode\_r), 175  
 light\_mode\_t(), 11, 15, 19, 23, 28, 32, 36, 40, 44, 49, 53, 57, 61, 65, 70, 74, 78, 82, 86, 90, 95, 99, 103, 107, 111, 116, 120, 124, 128, 133, 137, 141, 145, 149, 154, 158, 162, 166, 179  
 lightness, 174  
 linewidthness(lightness), 174  
 navy(jumble), 173  
 orange(jumble), 173  
 pink(jumble), 173  
 purple(jumble), 173  
 red(jumble), 173  
 scales::transform\_log10(), 12, 16, 20, 24, 28, 33, 37, 41, 45, 49, 54, 58, 62, 66, 70, 74, 79, 83, 87, 91, 95, 100, 104, 108, 112, 117, 121, 125, 129, 134, 138, 142, 146, 150, 154, 159, 163, 167  
 set\_blanket, 179  
 teal(jumble), 173