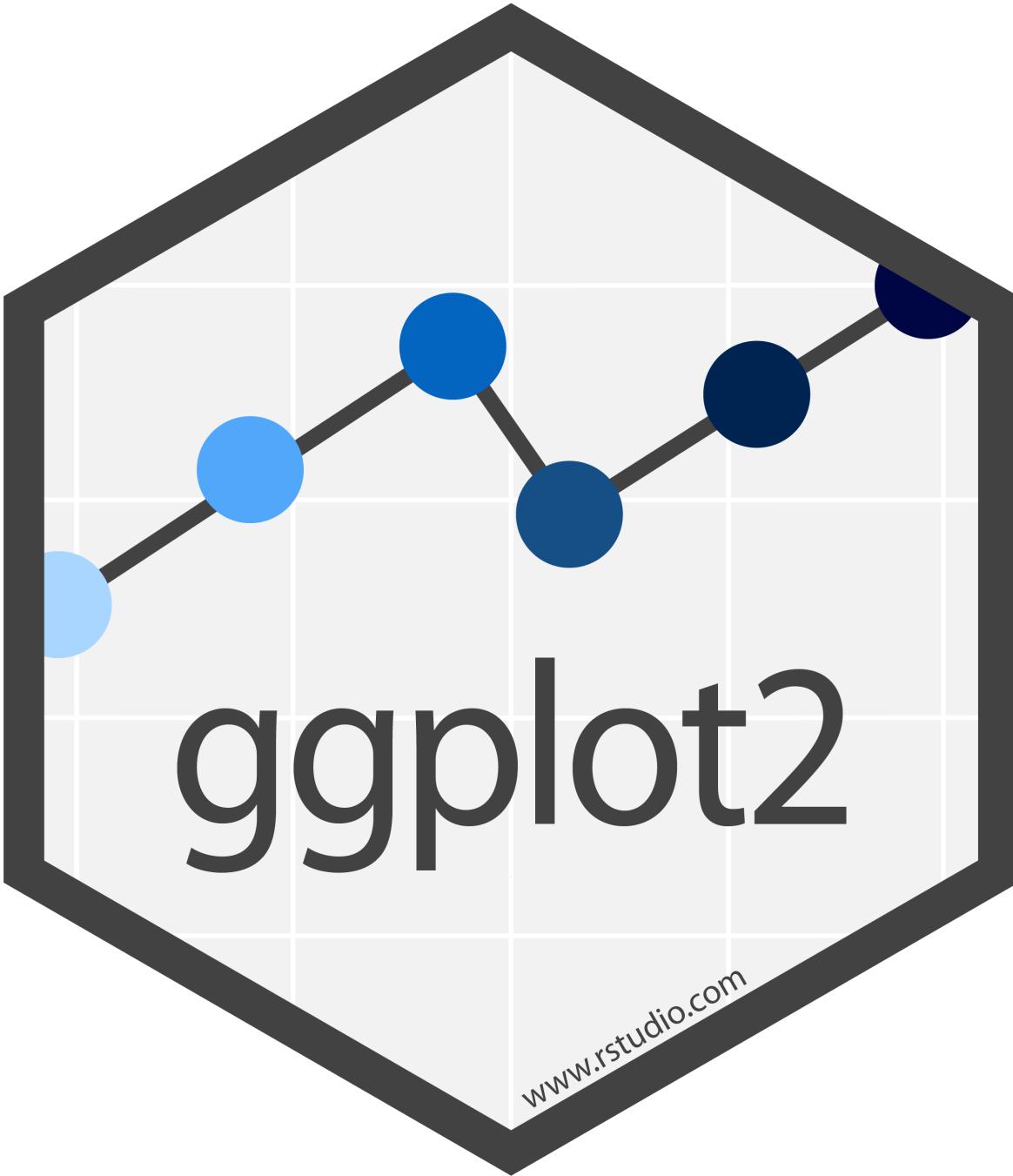


# Lec 09 - Visualization with ggplot2

Lecture 09

Dr. Colin Rundel



# The Grammar of Graphics

- Visualization concept created by Leland Wilkinson (*The Grammar of Graphics*, 1999)
- attempt to taxonimize the basic elements of statistical graphics
- Adapted for R by Hadley Wickham (2009)
  - consistent and compact syntax to describe statistical graphics
  - highly modular as it breaks up graphs into semantic components
  - ggplot2 is not meant as a guide to which graph to use and how to best convey your data (more on that later), but it does have some strong opinions.

# Terminology

A statistical graphic is a...

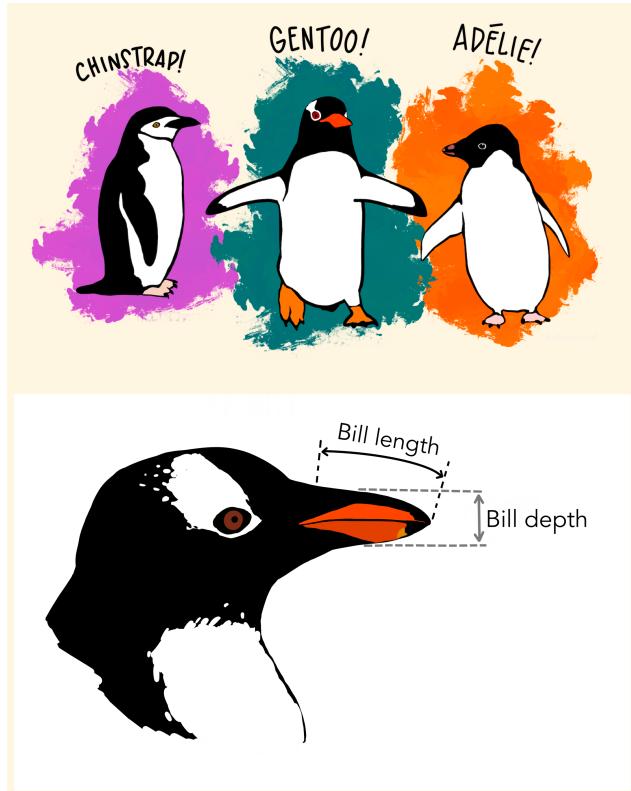
- mapping of **data**
- which may be **statistically transformed** (summarized, log-transformed, etc.)
- to **aesthetic attributes** (color, size, xy-position, etc.)
- using **geometric objects** (points, lines, bars, etc.)
- and mapped onto a specific **facet and coordinate system**

# Anatomy of a ggplot call

```
1 ggplot(  
2   data = [dataframe],  
3   mapping = aes(  
4     x = [var x], y = [var y],  
5     color = [var color],  
6     shape = [var shape],  
7     ...  
8   )  
9 ) +  
10 geom_[some geom] (  
11   mapping = aes(  
12     color = [var geom color],  
13     ...  
14   ))  
15 ) +  
16 ... # other geometries  
17 scale_[some axis]_[some scale]() +
```

# Data - Palmer Penguins

Measurements for penguin species, island in Palmer Archipelago, size (flipper length, body mass, bill dimensions), and sex.



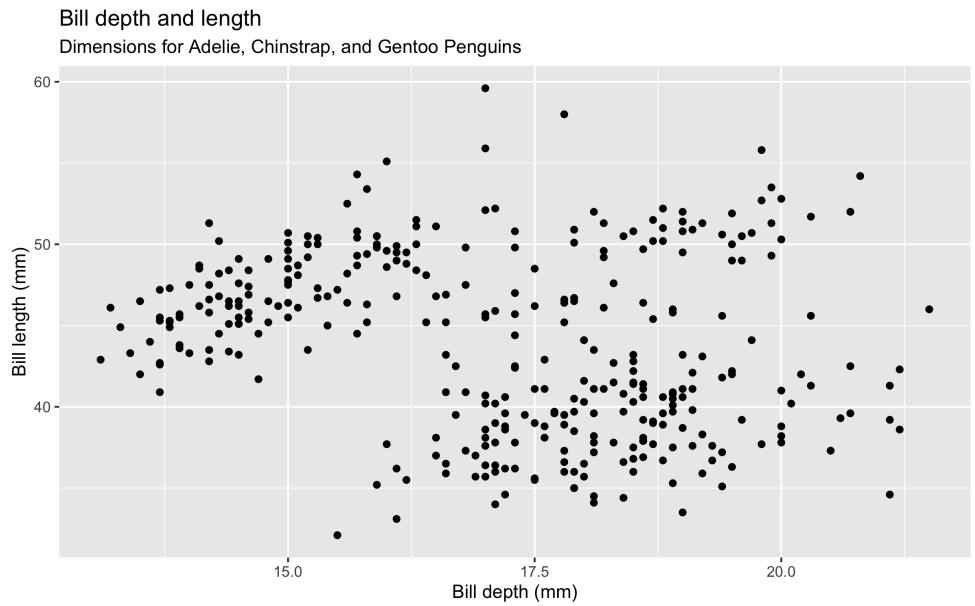
```
1 library(palmerpenguins)
2 penguins

# A tibble: 344 × 8
  species   island   bill_le...¹ bill_...² flipp...³ body_...⁴ sex
  <fct>     <fct>     <dbl>     <dbl>     <int>     <int> <fct>
  1 Adelie   Torgersen    39.1      18.7      181     3750 male 
  2 Adelie   Torgersen    39.5      17.4      186     3800 fema...
  3 Adelie   Torgersen    40.3       18        195     3250 fema...
  4 Adelie   Torgersen     NA        NA        NA      NA <NA>
  5 Adelie   Torgersen    36.7      19.3      193     3450 fema...
  6 Adelie   Torgersen    39.3      20.6      190     3650 male 
  7 Adelie   Torgersen    38.9      17.8      181     3625 fema...
  8 Adelie   Torgersen    39.2      19.6      195     4675 male 
  9 Adelie   Torgersen    34.1      18.1      193     3475 <NA>
 10 Adelie  Torgersen     42        20.2      190     4250 <NA>

# ... with 334 more rows, 1 more variable: year <int>, and
#   abbreviated variable names ¹bill_length_mm,
#   ²bill_depth_mm, ³flipper_length_mm, ⁴body_mass_g
```

# A basic ggplot

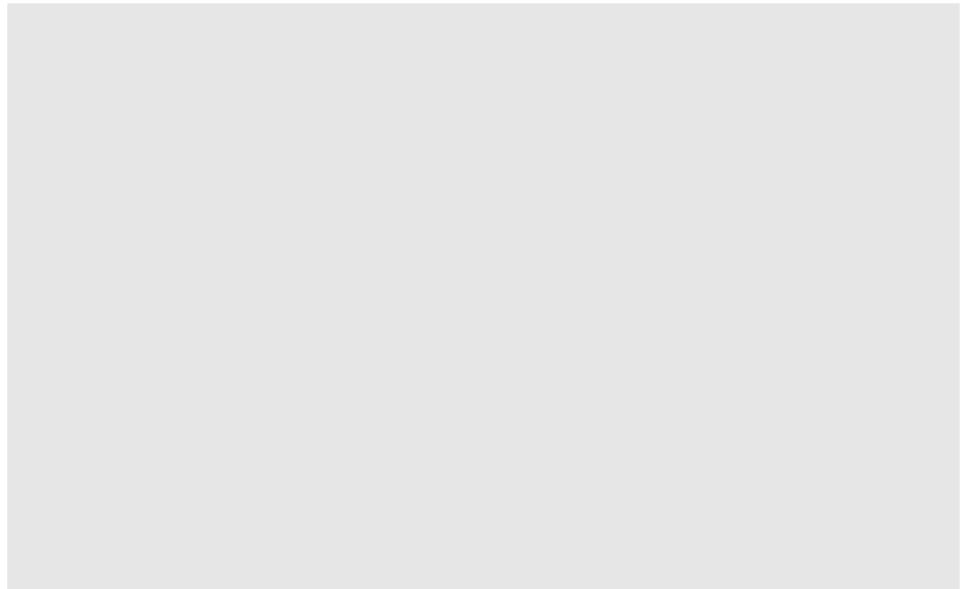
```
1 ggplot(  
2   data = penguins,  
3   mapping = aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )  
7 ) +  
8   geom_point() +  
9   labs(  
10    title = "Bill depth and length",  
11    subtitle = paste(  
12      "Dimensions for Adelie, Chinstrap,",  
13      "and Gentoo Penguins"  
14    ),  
15    x = "Bill depth (mm)",  
16    y = "Bill length (mm)",  
17    color = "Species"  
18  )
```



# Text <-> Plot

*Start with the `penguins` data frame*

```
1 ggplot(data = penguins)
```

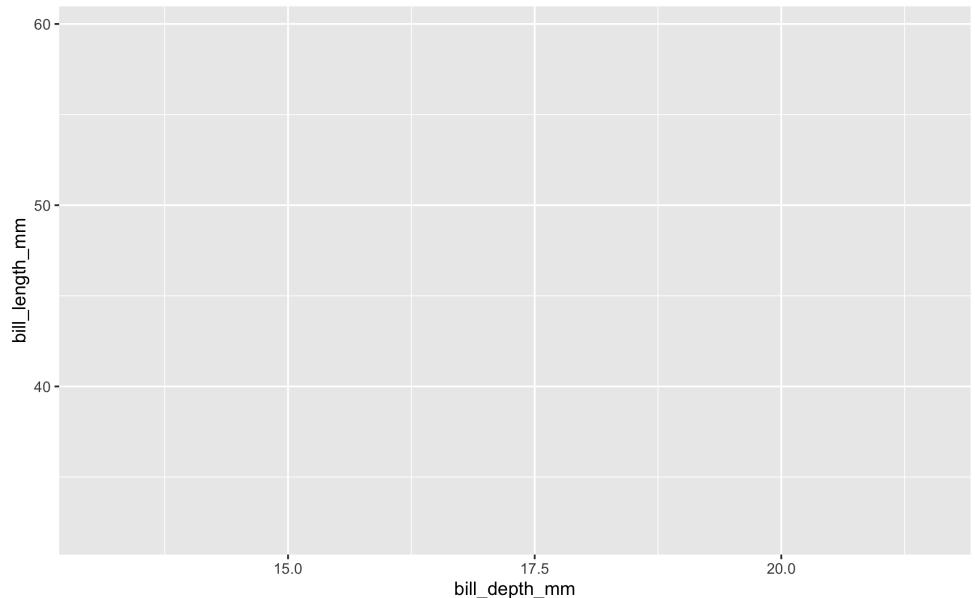


| Start with the `penguins` data frame, *map bill depth to the x-axis*

```
{r penguins-1, fig.show = "hide", warning =  
FALSE}\#| code-line-numbers: "2" ggplot(  
data = penguins,   mapping = aes(x =  
bill_depth_mm) )
```

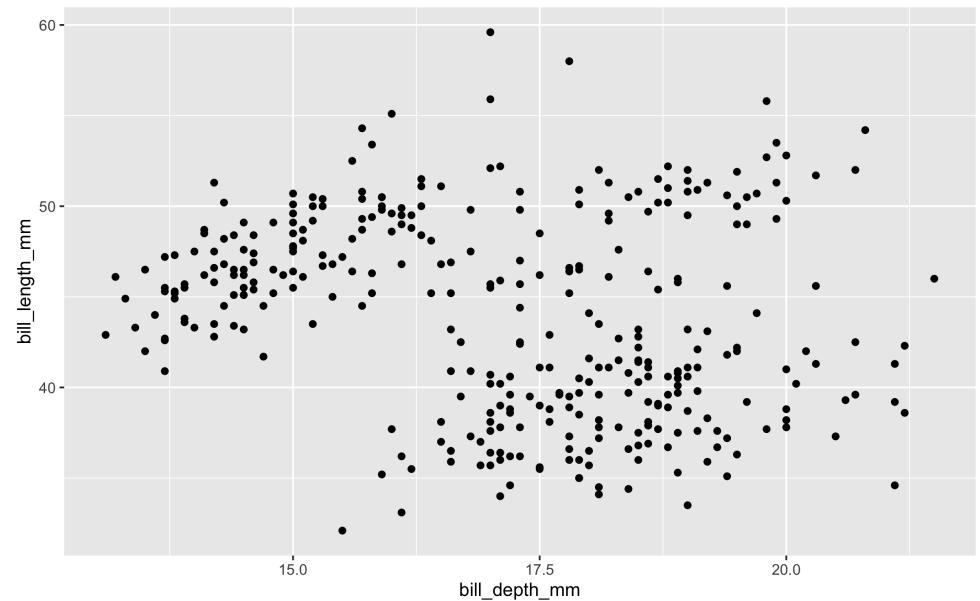
Start with the `penguins` data frame, map bill depth to the x-axis *and* map bill length to the y-axis.

```
1 ggplot(  
2   data = penguins,  
3   mapping = aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )  
7 )
```



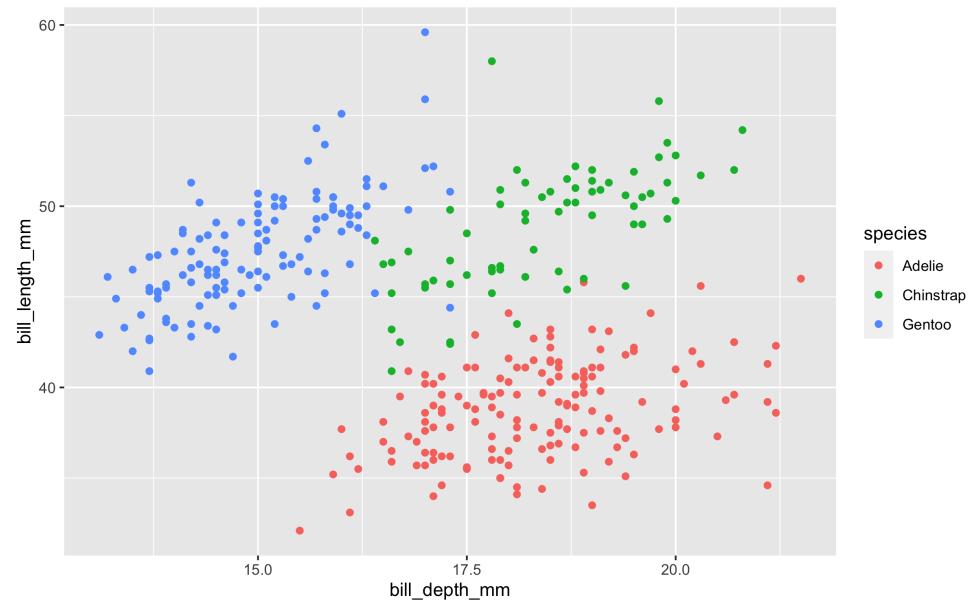
Start with the `penguins` data frame, map bill depth to the x-axis and map bill length to the y-axis.  
*Represent each observation with a point*

```
1 ggplot(  
2   data = penguins,  
3   mapping = aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )  
7 ) +  
8 geom_point()
```



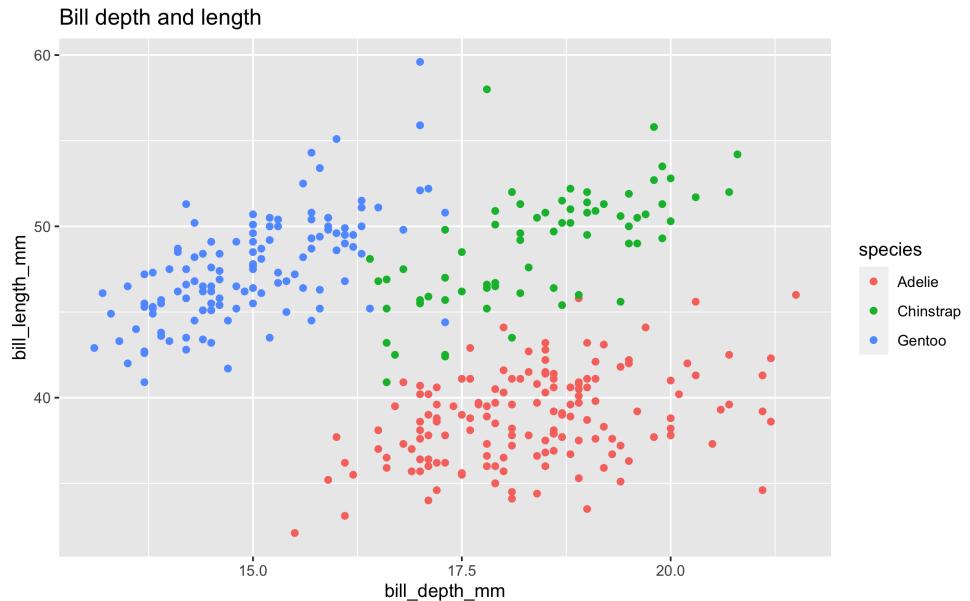
Start with the `penguins` data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point *and map species to the color of each point.*

```
1 ggplot(  
2   data = penguins,  
3   mapping = aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )  
7 ) +  
8 geom_point(  
9   mapping = aes(color = species)  
10 )
```



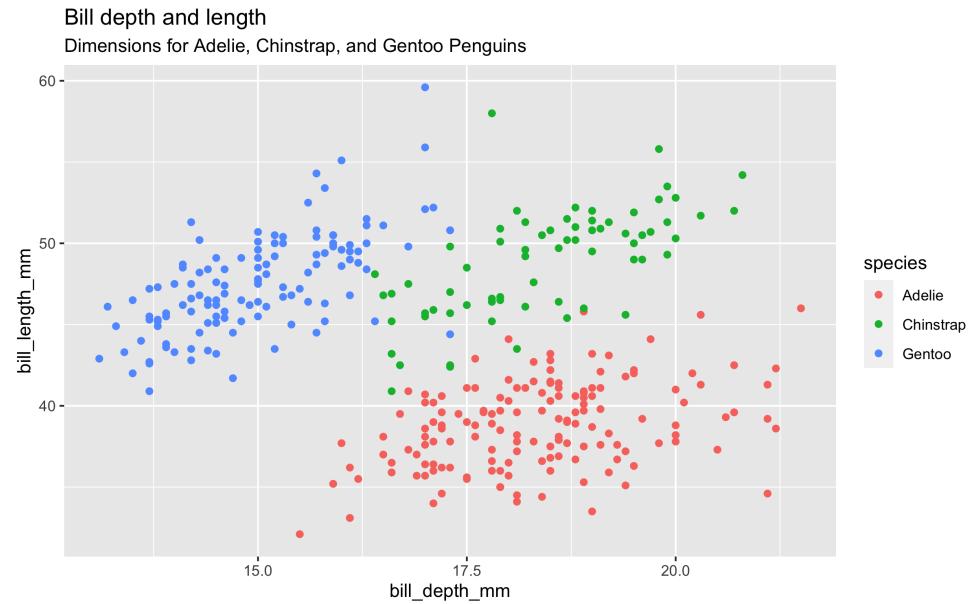
Start with the `penguins` data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the color of each point. *Title the plot “Bill depth and length”*

```
1 ggplot(  
2   data = penguins,  
3   mapping = aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )  
7 ) +  
8 geom_point(  
9   mapping = aes(color = species)  
10 ) +  
11 labs(title = "Bill depth and length")
```



Start with the `penguins` data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the color of each point. Title the plot “Bill depth and length”, *add the subtitle “Dimensions for Adelie, Chinstrap, and Gentoo Penguins”*

```
1 ggplot(  
2   data = penguins,  
3   mapping = aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )  
7 ) +  
8   geom_point(  
9     mapping = aes(color = species)  
10  ) +  
11  labs(  
12    title = "Bill depth and length",  
13    subtitle = paste("Dimensions for Adelie,",  
14          "Chinstrap, and Gentoo",  
15          "Penguins")  
16  )
```

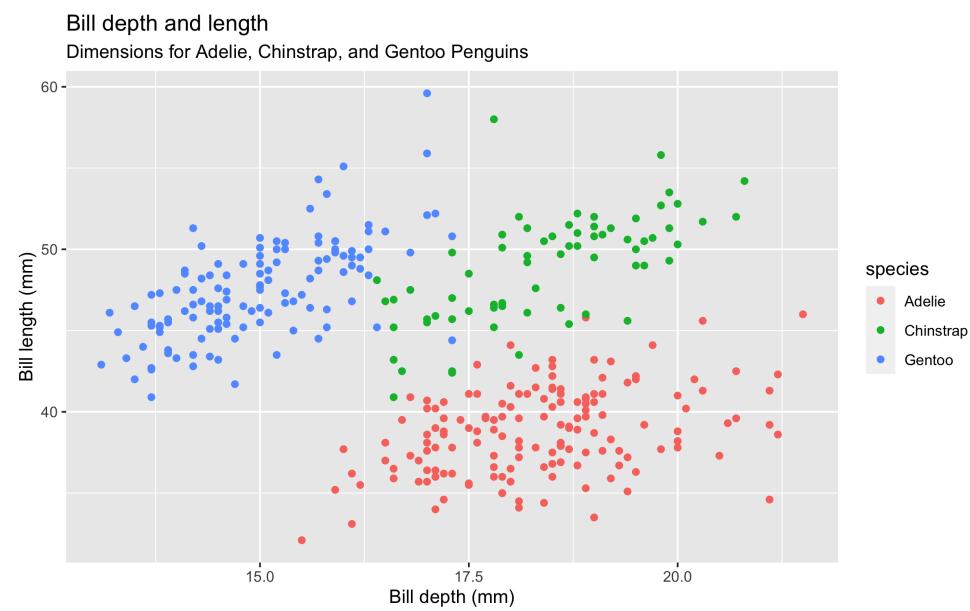


Start with the `penguins` data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the color of each point. Title the plot “Bill depth and length”, add the subtitle “Dimensions for Adelie, Chinstrap, and Gentoo Penguins”, *label the x and y axes as “Bill depth (mm)” and “Bill length (mm)”, respectively*

```

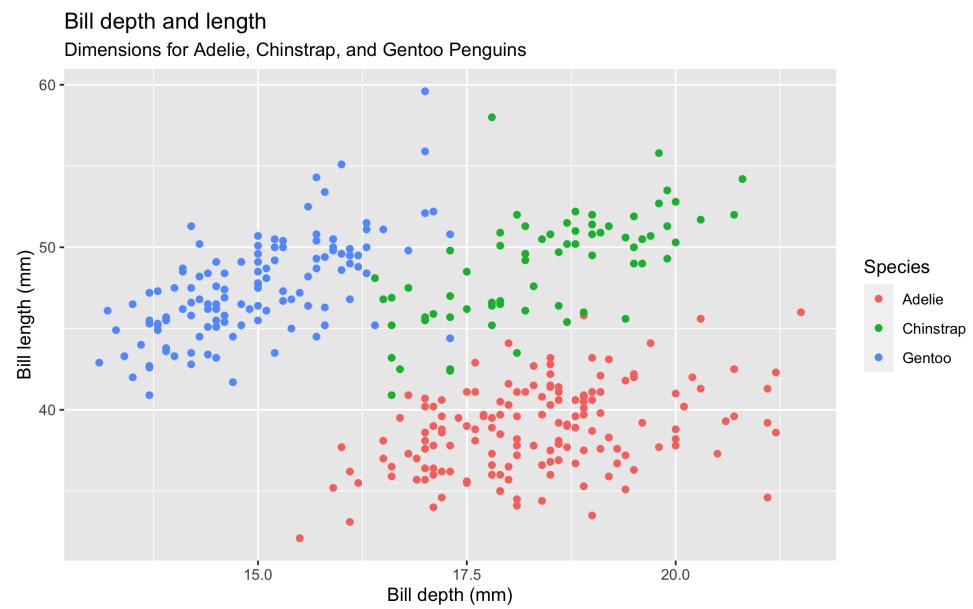
1 ggplot(
2   data = penguins,
3   mapping = aes(
4     x = bill_depth_mm,
5     y = bill_length_mm
6   )
7 ) +
8   geom_point(
9     mapping = aes(color = species)
10 ) +
11   labs(
12     title = "Bill depth and length",
13     subtitle = paste("Dimensions for Adelie,",
14                      "Chinstrap, and Gentoo",
15                      "Penguins"),
16     x = "Bill depth (mm)",
17     y = "Bill length (mm)"
18 )

```



Start with the `penguins` data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the color of each point. Title the plot “Bill depth and length”, add the subtitle “Dimensions for Adelie, Chinstrap, and Gentoo Penguins”, label the x and y axes as “Bill depth (mm)” and “Bill length (mm)”, respectively, *label the legend “Species”*

```
1 ggplot(  
2   data = penguins,  
3   mapping = aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )  
7 ) +  
8   geom_point(  
9     mapping = aes(color = species)  
10 ) +  
11   labs(  
12     title = "Bill depth and length",  
13     subtitle = paste("Dimensions for Adelie,",  
14                     "Chinstrap, and Gentoo",  
15                     "Penguins"),  
16     x = "Bill depth (mm)",  
17     y = "Bill length (mm)",  
18     color = "Species"  
19 )
```

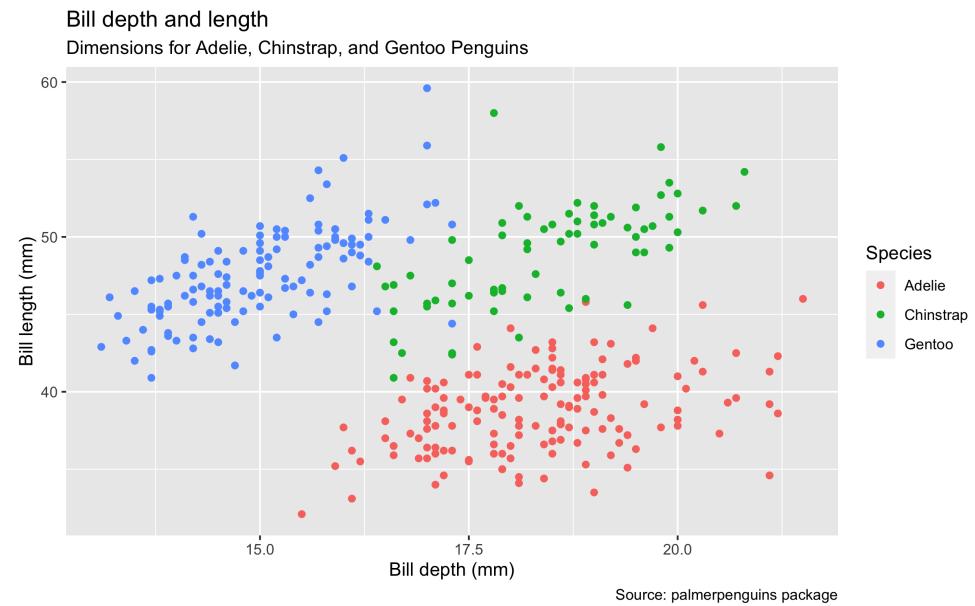


Start with the `penguins` data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the color of each point. Title the plot “Bill depth and length”, add the subtitle “Dimensions for Adelie, Chinstrap, and Gentoo Penguins”, label the x and y axes as “Bill depth (mm)” and “Bill length (mm)”, respectively, label the legend “Species”, and add a caption for the data source.

```

1 ggplot(
2   data = penguins,
3   mapping = aes(
4     x = bill_depth_mm,
5     y = bill_length_mm
6   )
7 ) +
8   geom_point(
9     mapping = aes(color = species)
10 ) +
11   labs(
12     title = "Bill depth and length",
13     subtitle = paste("Dimensions for Adelie,",
14                      "Chinstrap, and Gentoo",
15                      "Penguins"),
16     x = "Bill depth (mm)",
17     y = "Bill length (mm)",
18     color = "Species",
19     caption = "Source: palmerpenguins package"
20 )

```

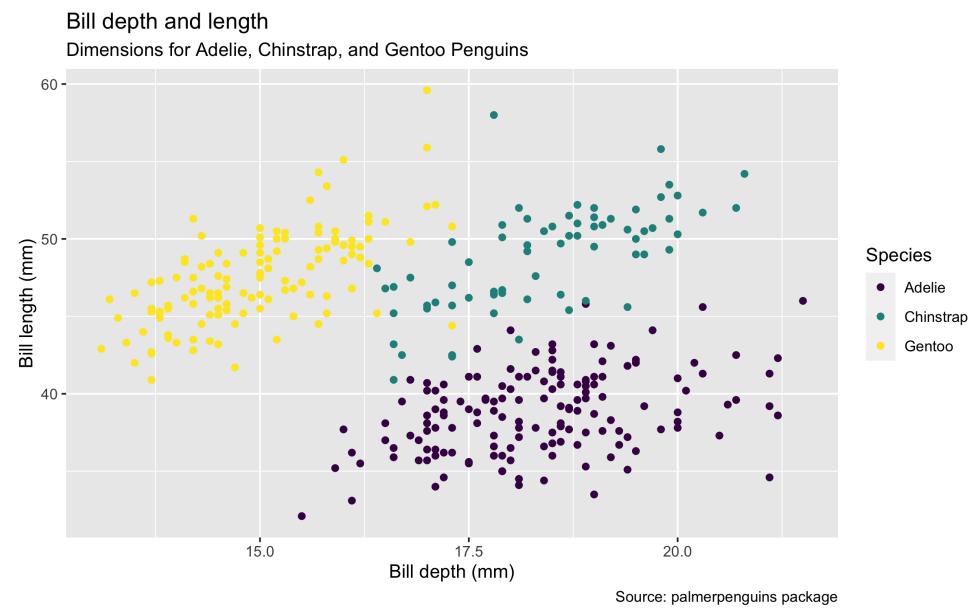


Start with the `penguins` data frame, map bill depth to the x-axis and map bill length to the y-axis. Represent each observation with a point and map species to the color of each point. Title the plot “Bill depth and length”, add the subtitle “Dimensions for Adelie, Chinstrap, and Gentoo Penguins”, label the x and y axes as “Bill depth (mm)” and “Bill length (mm)”, respectively, label the legend “Species”, and add a caption for the data source. *Finally, use the viridis color palette for all points.*

```

1 ggplot(
2   data = penguins,
3   mapping = aes(
4     x = bill_depth_mm,
5     y = bill_length_mm
6   )
7 ) +
8   geom_point(
9     mapping = aes(color = species)
10 ) +
11   labs(
12     title = "Bill depth and length",
13     subtitle = paste("Dimensions for Adelie,",
14                      "Chinstrap, and Gentoo",
15                      "Penguins"),
16     x = "Bill depth (mm)",
17     y = "Bill length (mm)",
18     color = "Species",
19     caption = "Source: palmerpenguins package"
20   ) +
21   scale_color_viridis_d()

```



# Argument names

Often we omit the names of first two arguments when building plots with `ggplot()`.

```
1 ggplot(  
2   data = penguins,  
3   mapping = aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )  
7 ) +  
8 geom_point(  
9   mapping = aes(color = species)  
10 ) +  
11 scale_color_viridis_d()
```

```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )  
7 ) +  
8 geom_point(  
9   aes(color = species)  
10 ) +  
11 scale_color_viridis_d()
```

# Aesthetics

# Aesthetics options

Commonly used characteristics of plotting geometries that can be **mapped to a specific variable** in the data, examples include:

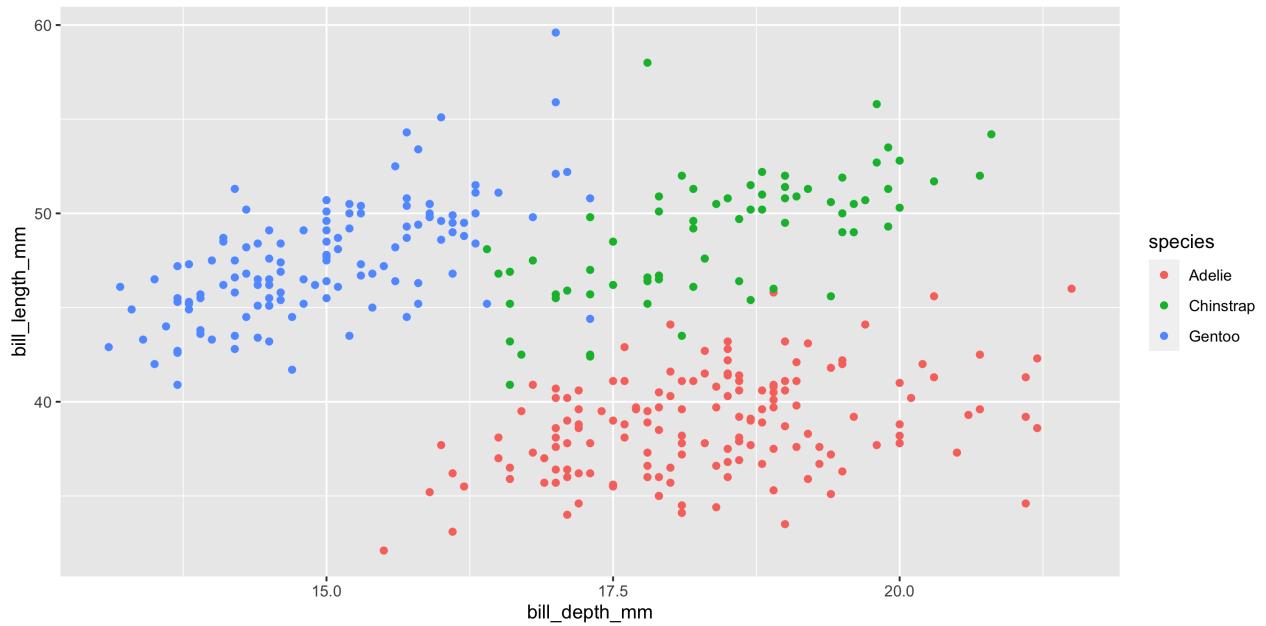
- position (`x`, `y`)
- `color`
- `shape`
- `size`
- `alpha` (transparency)

Different geometries have different aesthetics that can be used - see the `ggplot2 geoms` help files for listings.

- Aesthetics given in `ggplot()` apply to all `geoms`.
- Aesthetics for a specific `geom_*`() can be overridden via `mapping` or as an argument.

# color

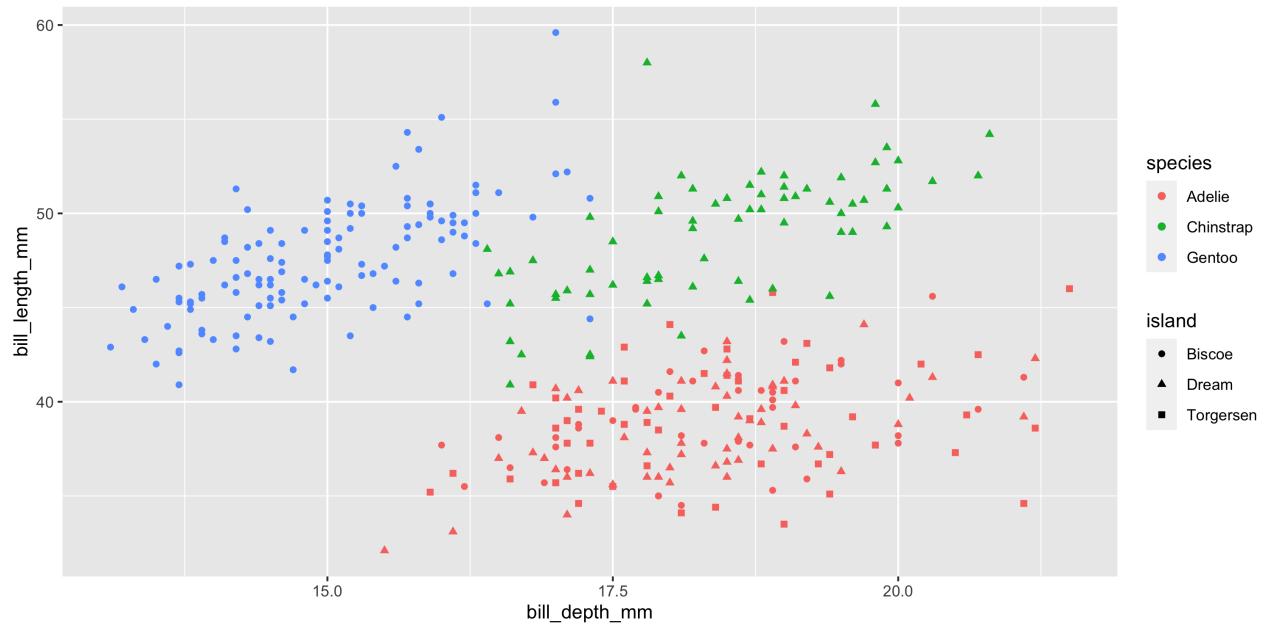
```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   ))  
7 ) +  
8 geom_point(  
9   aes(color = species))  
10 )
```



# Shape

Mapped to a different variable than color

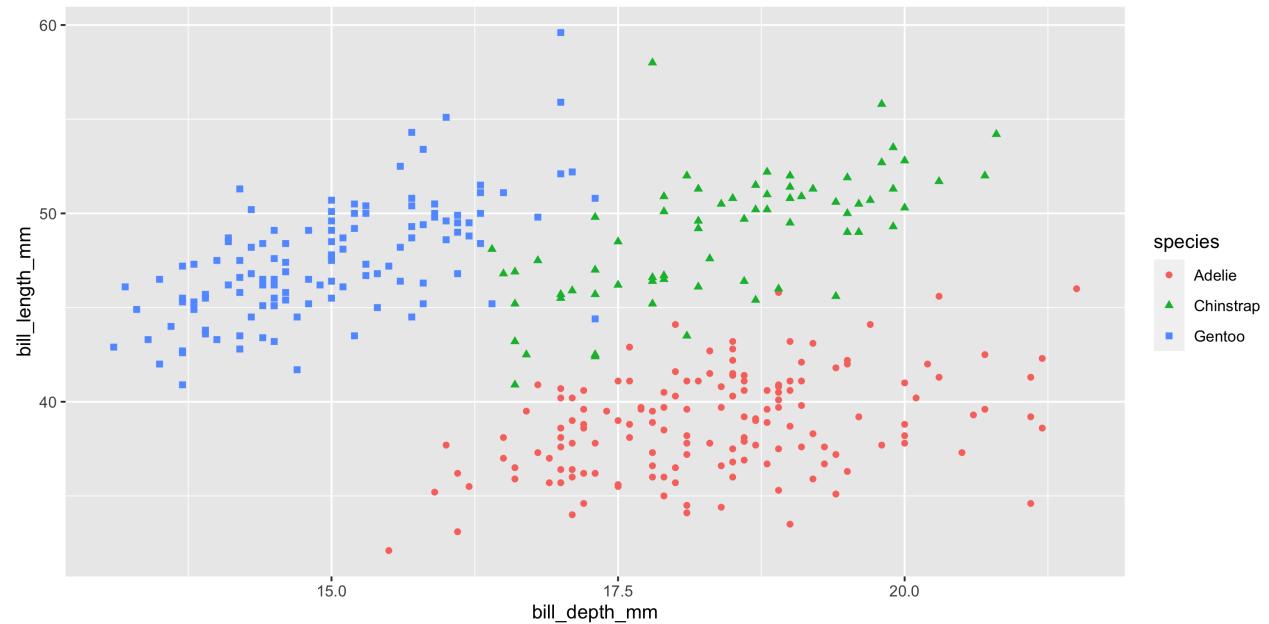
```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )  
7 ) +  
8   geom_point(  
9     aes(  
10    color = species,  
11    shape = island  
12  )  
13 )
```



# Shape

Mapped to same variable as color

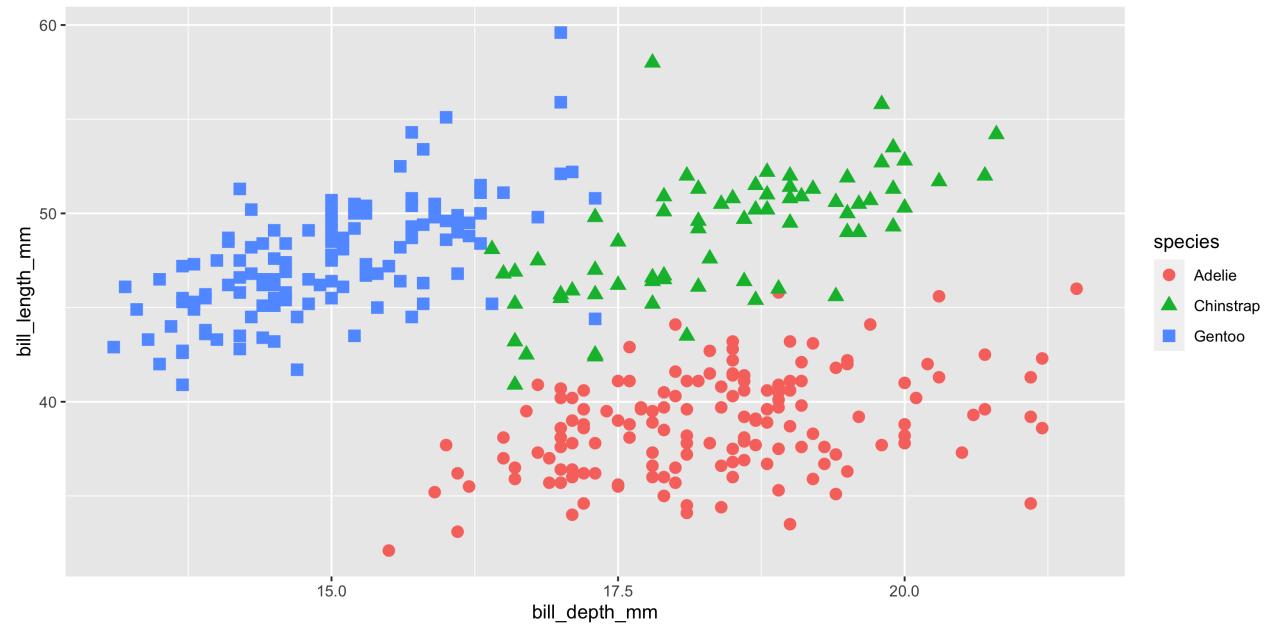
```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )  
7 ) +  
8   geom_point(  
9     aes(  
10    color = species,  
11    shape = species  
12  )  
13 )
```



# Size

Using a fixed value (note this value is outside of the aes call)

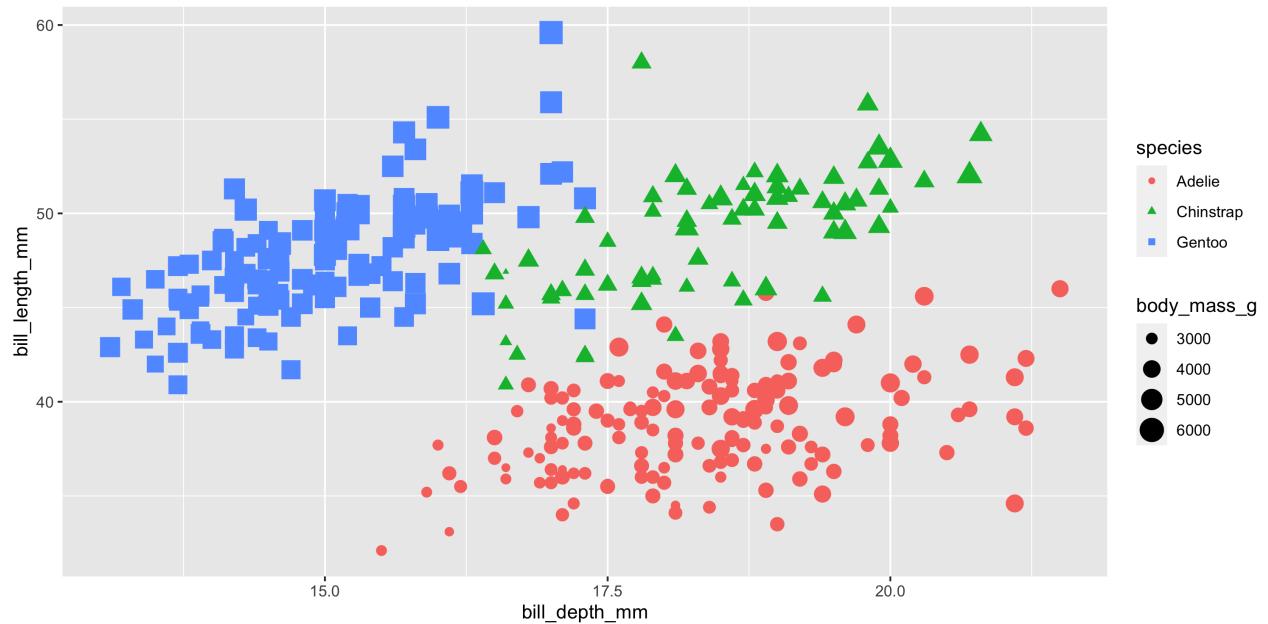
```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )) +  
7   geom_point(  
8     aes(  
9       color = species,  
10      shape = species  
11    ),  
12    size = 3  
13  )
```



# Size

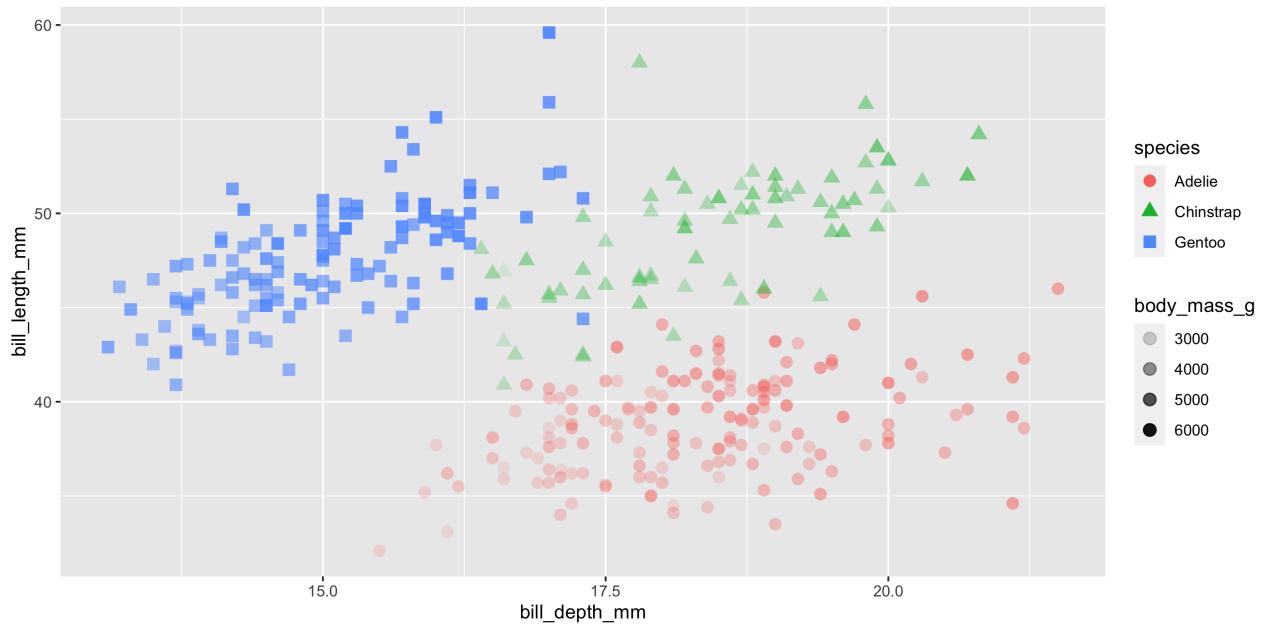
Mapped to a variable

```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   )  
7 ) +  
8   geom_point(  
9     aes(  
10    color = species,  
11    shape = species,  
12    size = body_mass_g  
13  ),  
14 )
```



# Alpha

```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   ))  
7 ) +  
8 geom_point(  
9   aes(  
10    color = species,  
11    shape = species,  
12    alpha = body_mass_g  
13  ),  
14  size = 3  
15 )
```



# Mapping vs settings

- **Mapping** - Determine an aesthetic (the size, alpha, etc.) of a geom based on the values of a variable in the data
  - wrapped by `aes()` and pass as `mapping` argument to `ggplot()` or `geom_*`().
- **Setting** - Determine an aesthetic (the size, alpha, etc.) of a geom **not** based on the values of a variable in the data, usually a constant value.
  - passed directly into `geom_*`() as an argument.

From the previous slide `color`, `shape`, and `alpha` are all **aesthetics** while `size` was a **setting**.

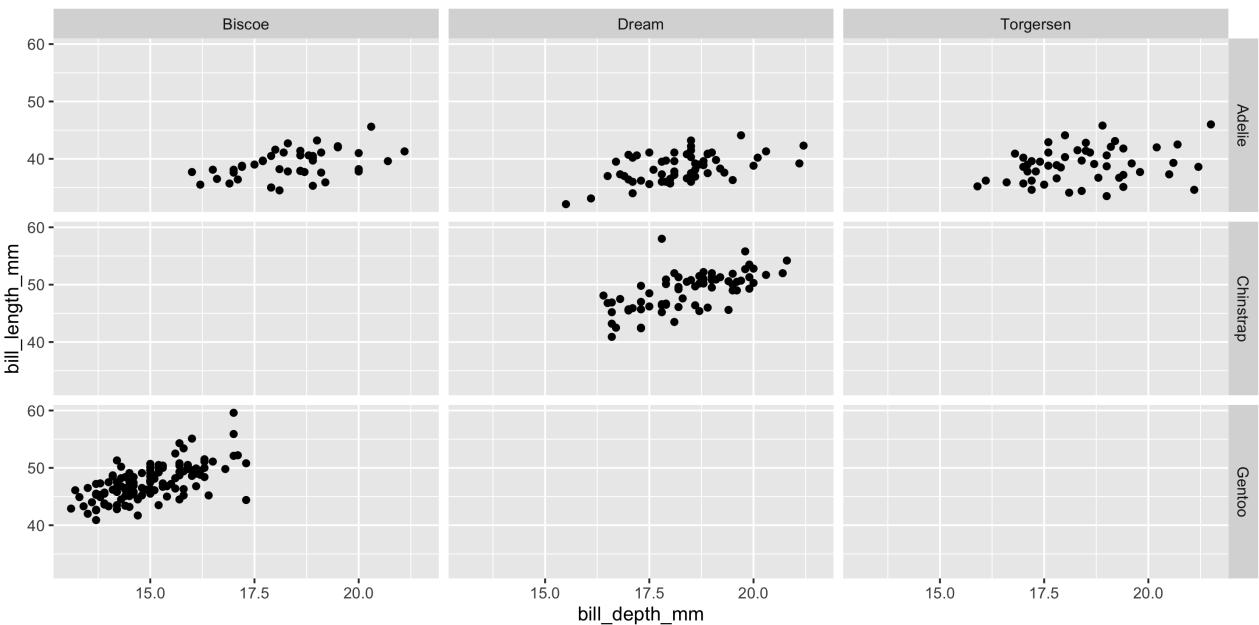
# Faceting

# Faceting

- Smaller plots that display different subsets of the data
- Useful for exploring conditional relationships and large data
- Sometimes referred to as “small multiples”

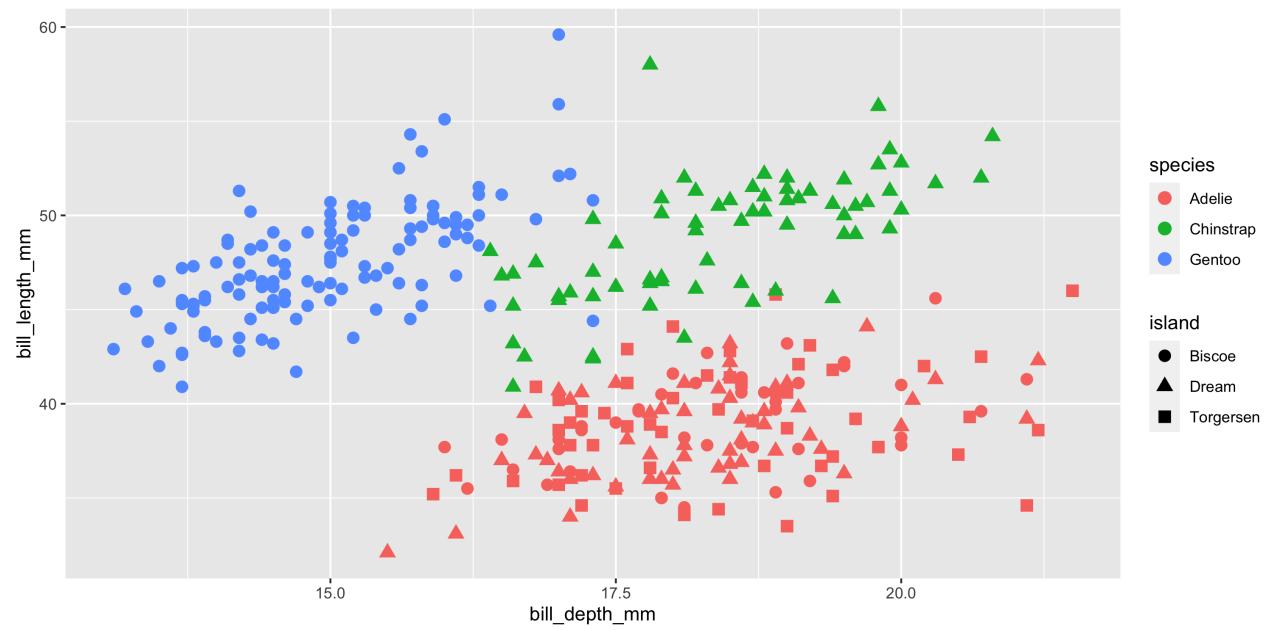
# facet\_grid

```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   ))  
7 ) +  
8 geom_point() +  
9 facet_grid(  
  species ~ island  
)
```



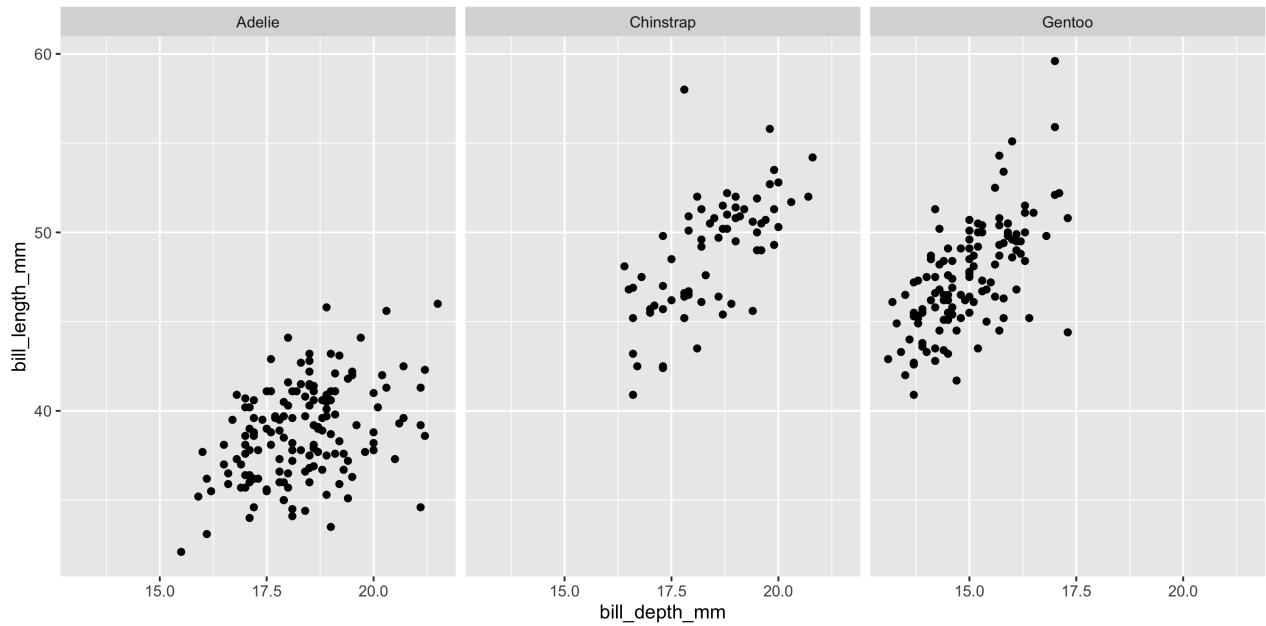
# Compare with ...

```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   ))  
7 ) +  
8 geom_point(  
9   aes(  
10    color = species,  
11    shape = island  
12  ),  
13  size = 3  
14 )
```



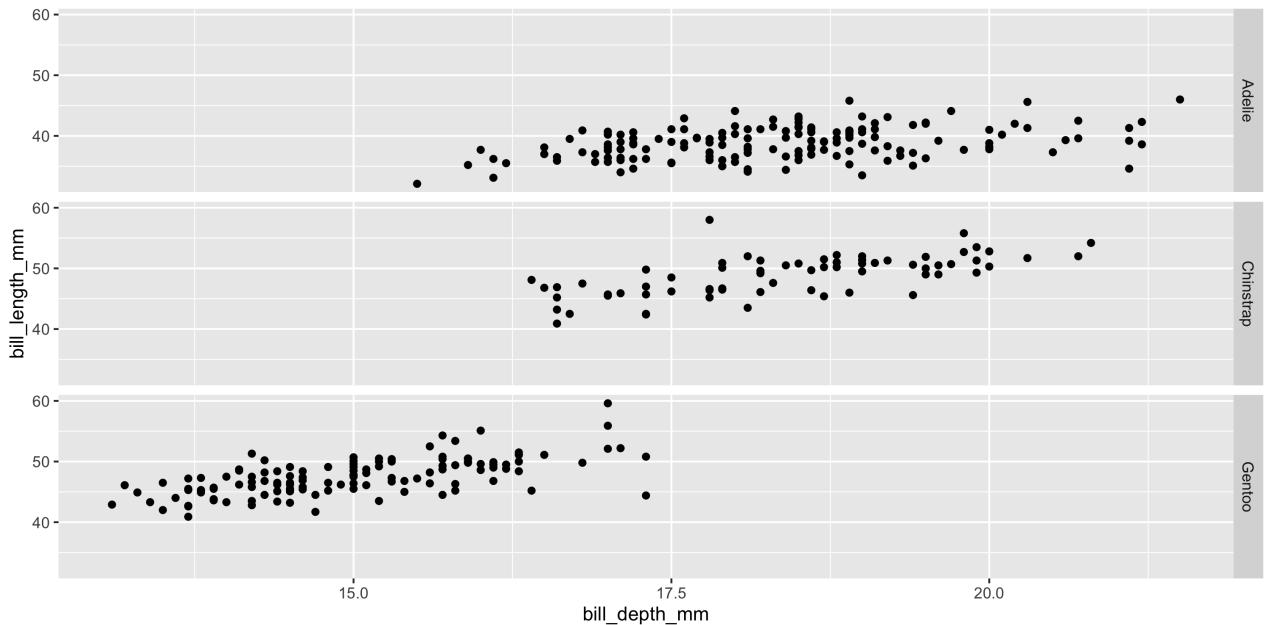
# facet\_grid (cols)

```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   ))  
7 ) +  
8 geom_point() +  
9 facet_grid(  
~ species  
)
```



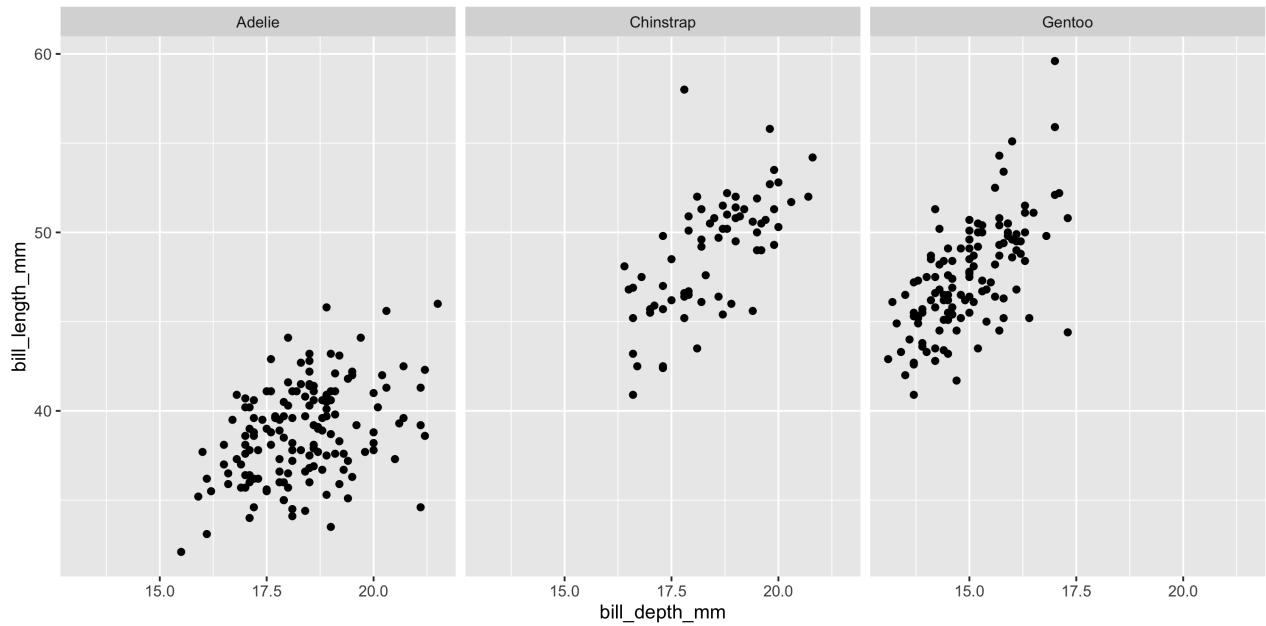
# facet\_grid (rows)

```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   ))  
7 ) +  
8 geom_point() +  
9 facet_grid(  
  species ~ .  
)
```



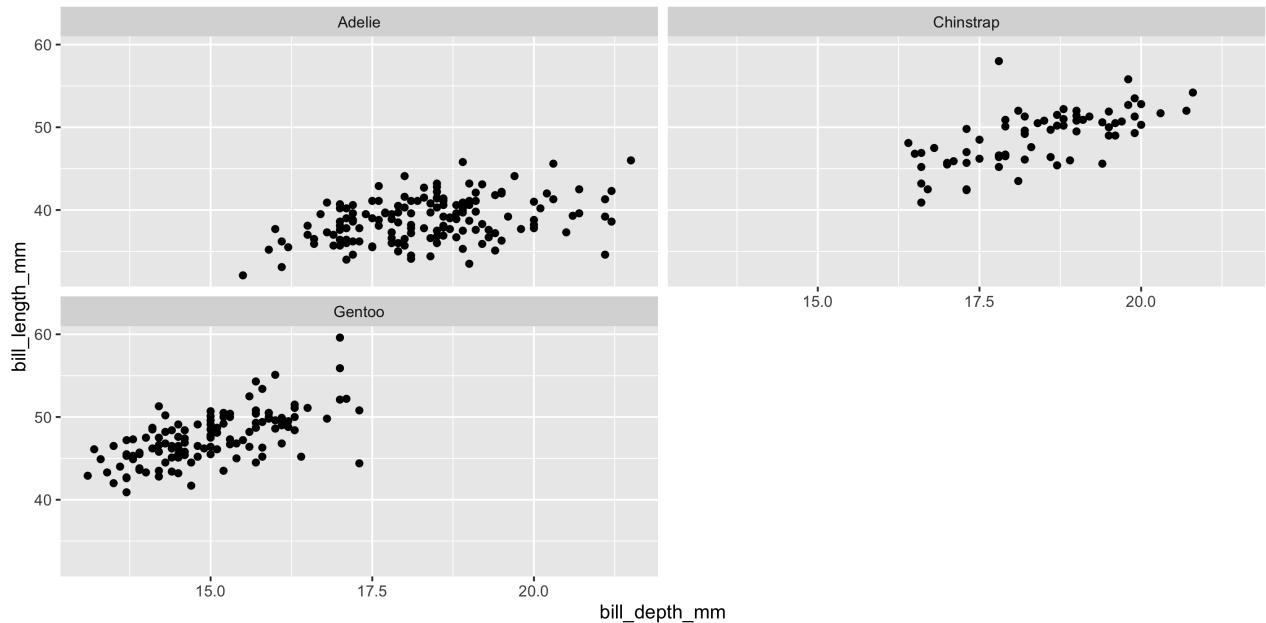
# facet\_wrap

```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   ))  
7 ) +  
8 geom_point() +  
9 facet_wrap(  
~ species  
)
```



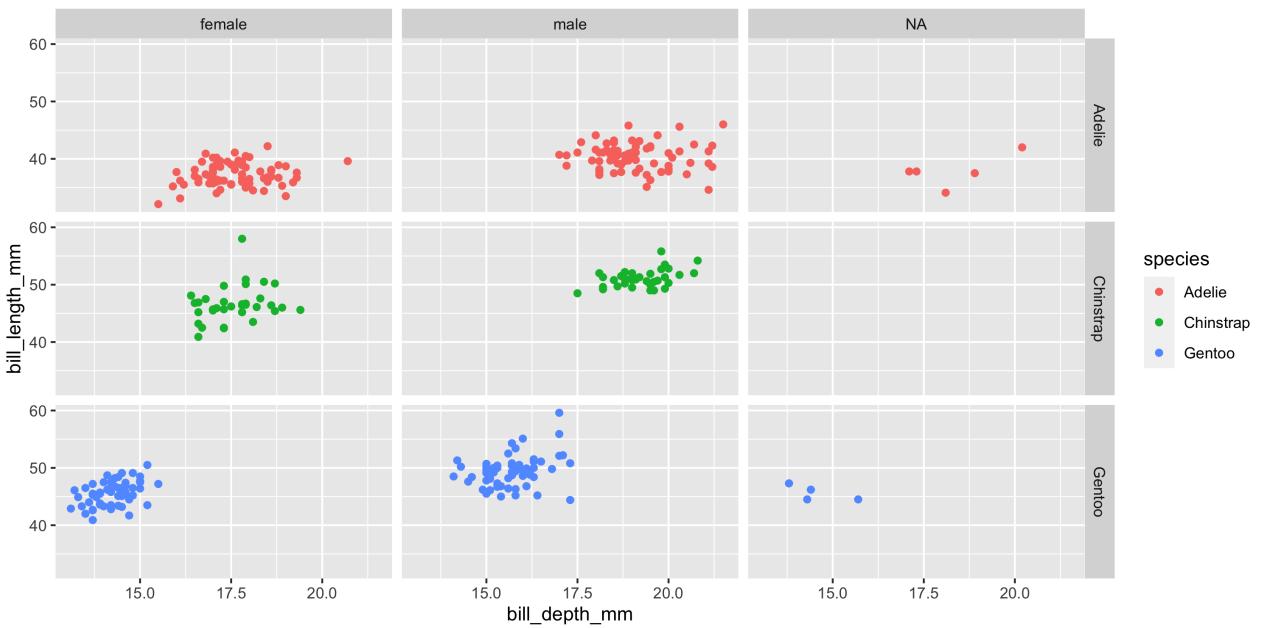
# facet\_wrap

```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm  
6   ))  
7 ) +  
8 geom_point() +  
9 facet_wrap(  
10   ~ species, ncol = 2  
11 )
```



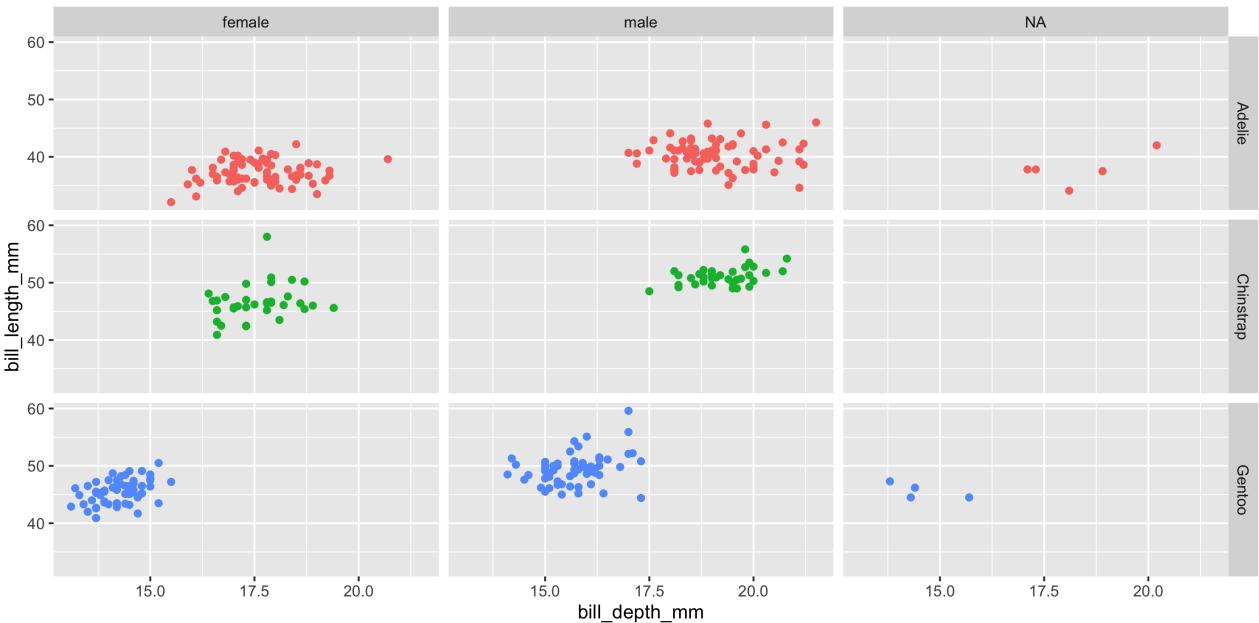
# Faceting and color

```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm,  
6     color = species  
7   )  
8 ) +  
9   geom_point() +  
10  facet_grid(  
11    species ~ sex  
12 )
```



# Hiding legend elements

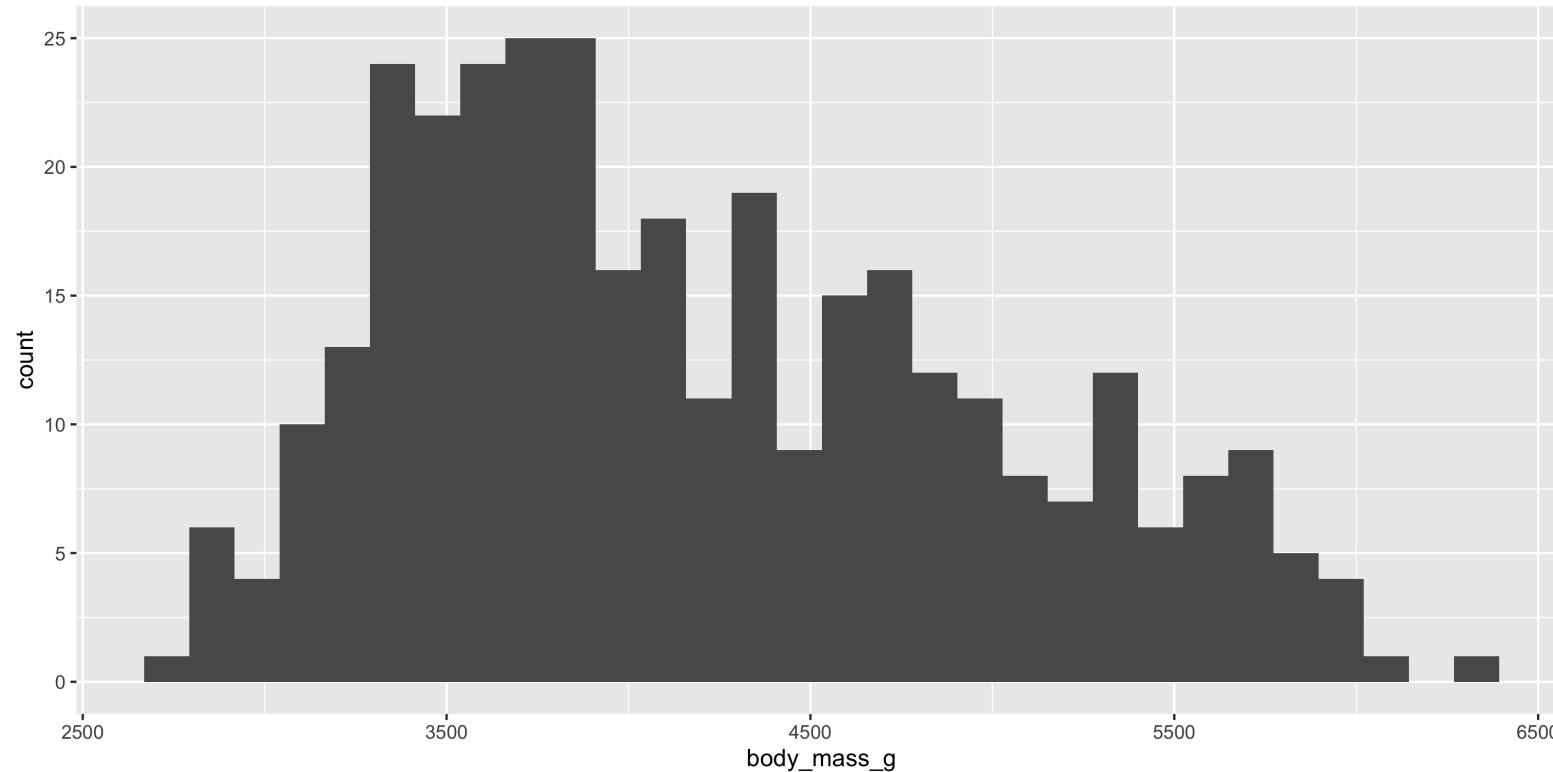
```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm,  
6     color = species  
7   )  
8 ) +  
9 geom_point() +  
10 facet_grid(  
11   species ~ sex  
12 ) +  
13 guides(color = "none")
```



# A brief plot tour of ggplot2 plots

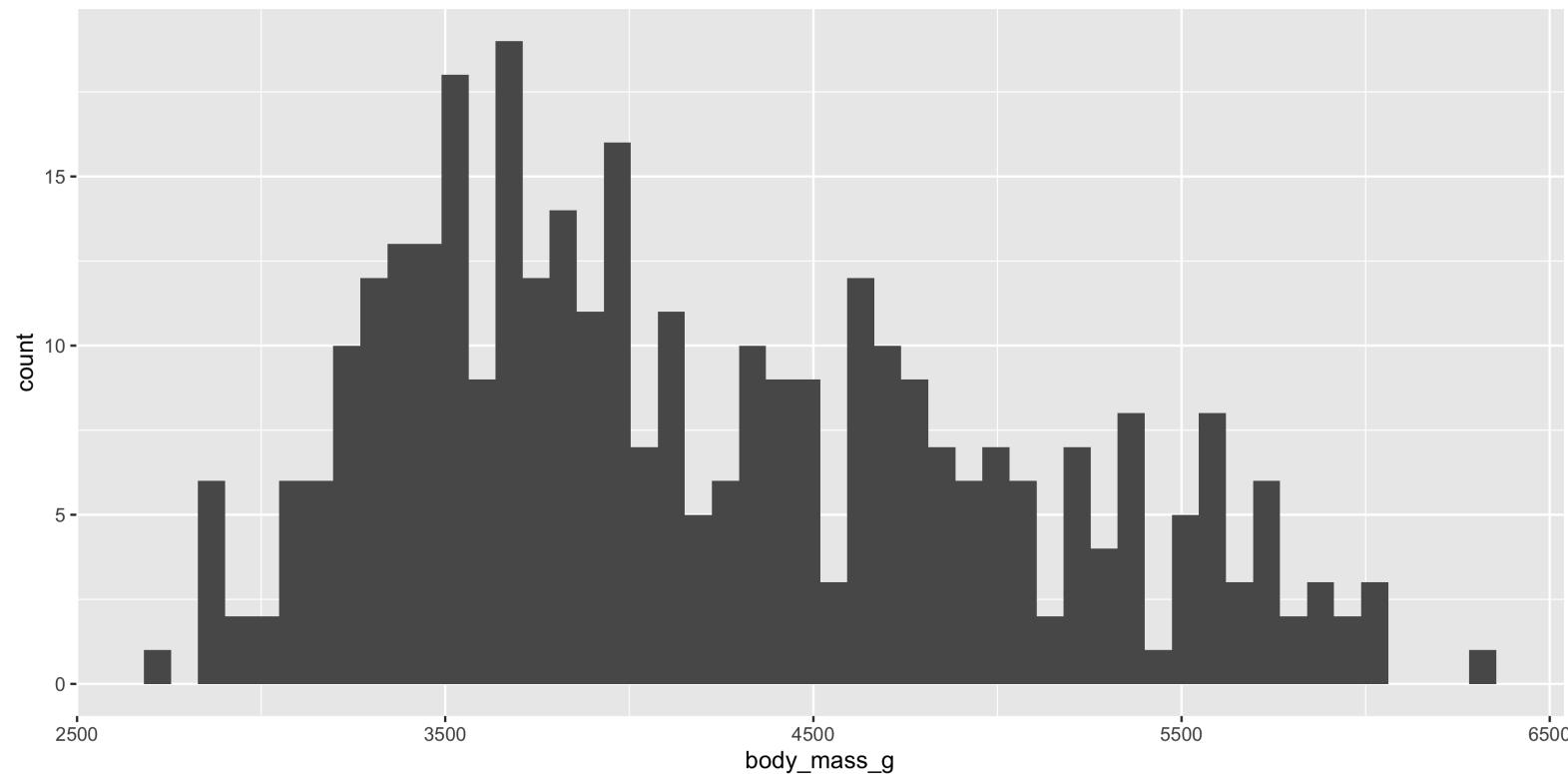
# Histograms

```
1 ggplot(  
2   penguins,  
3   aes(x = body_mass_g)  
4 ) +  
5   geom_histogram()
```



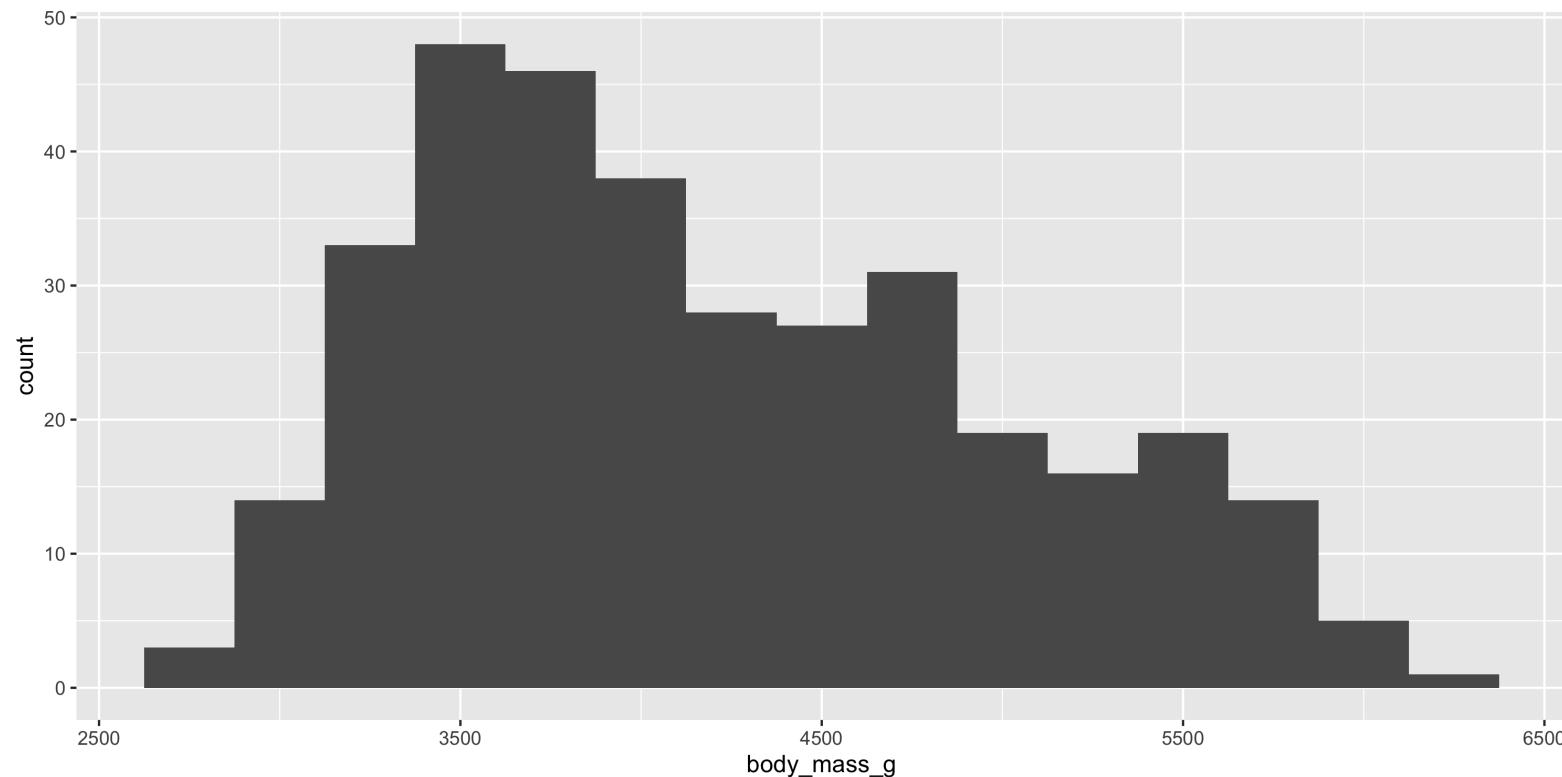
# Histograms - bins

```
1 ggplot(  
2   penguins,  
3   aes( x = body_mass_g )  
4 ) +  
5   geom_histogram(bins = 50)
```



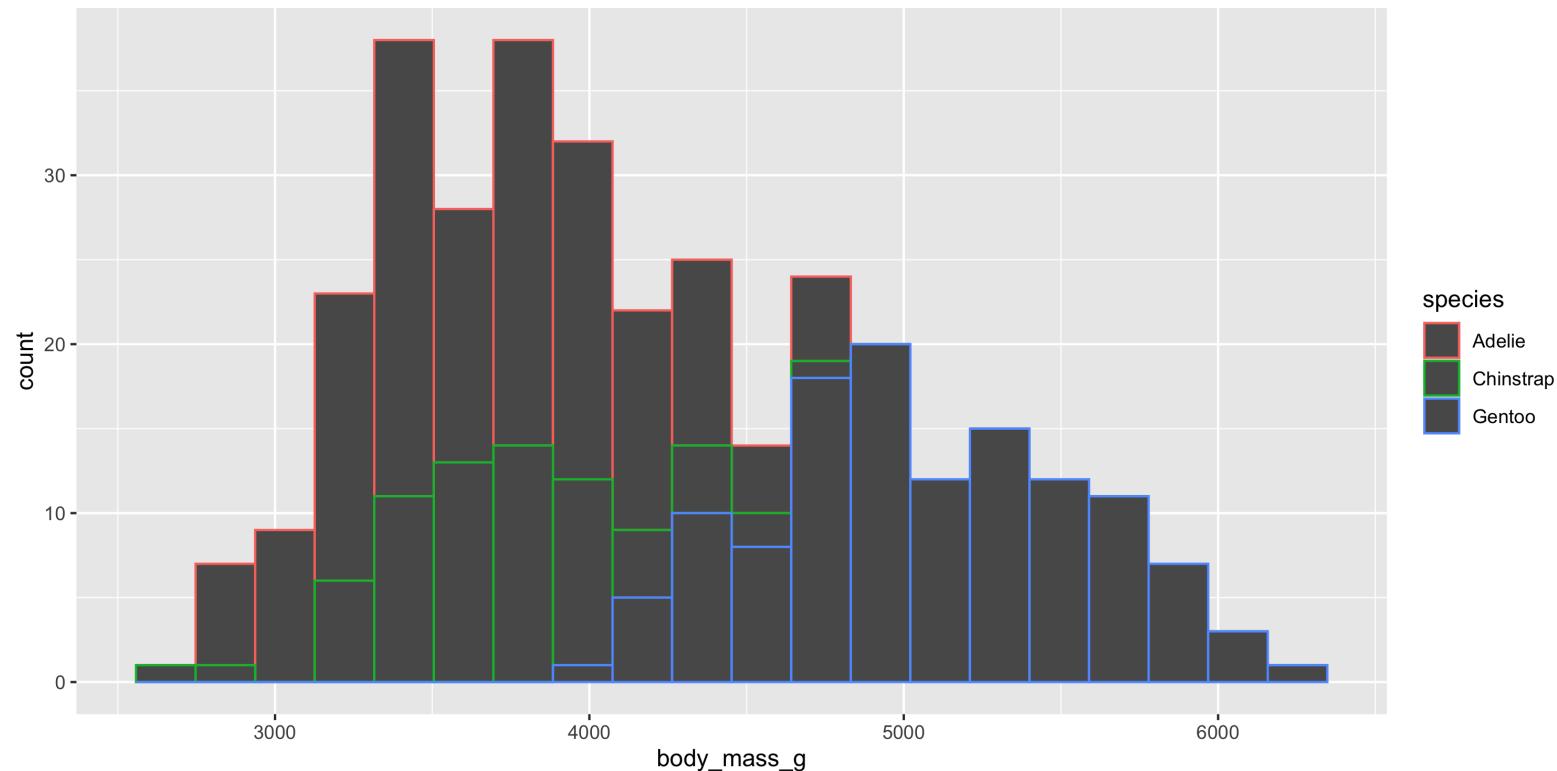
# Histograms - binwidth

```
1 ggplot(  
2   penguins,  
3   aes( x = body_mass_g )  
4 ) +  
5   geom_histogram(binwidth = 250)
```



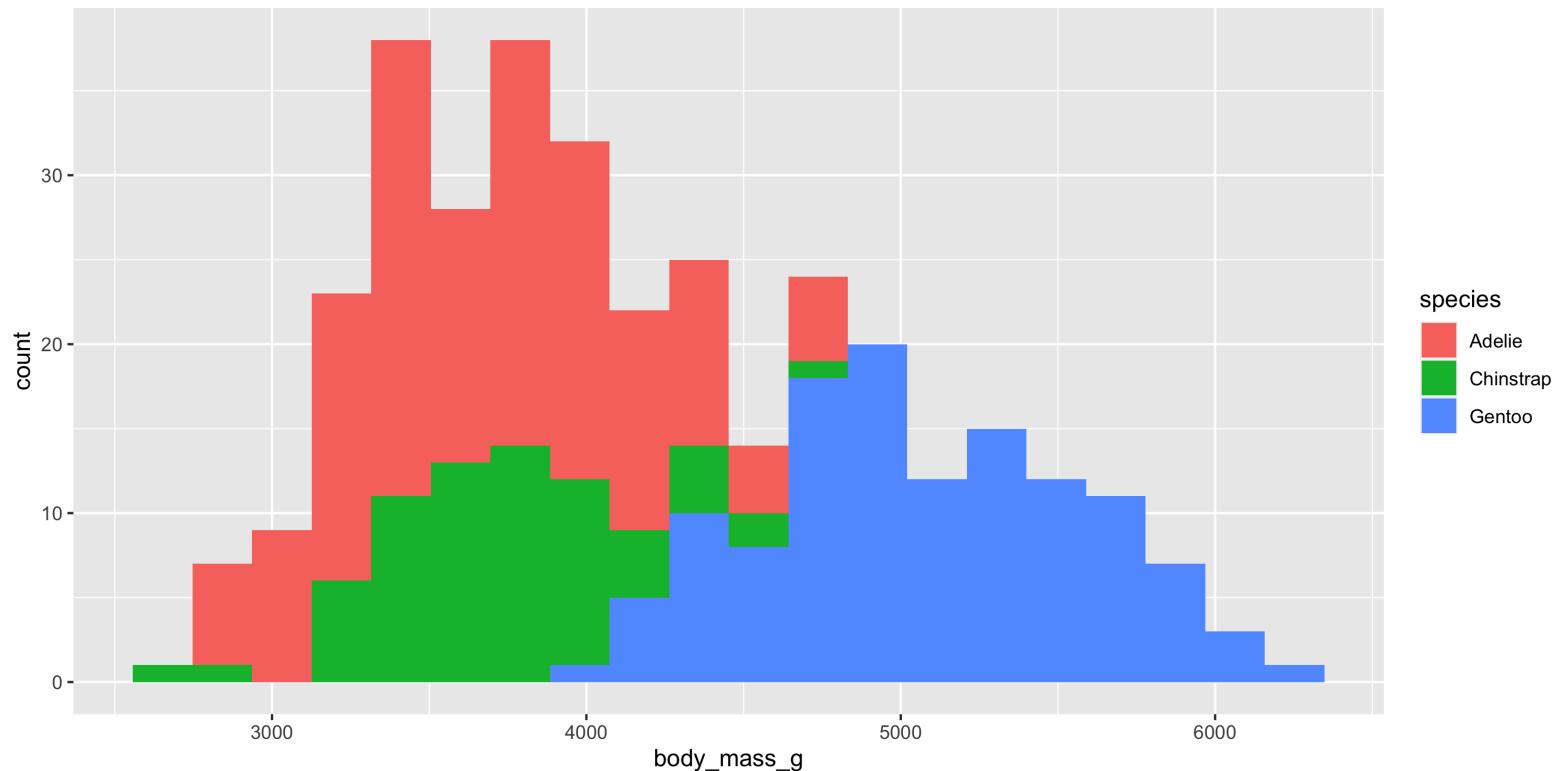
# Histograms - color

```
1 ggplot(  
2   penguins,  
3   aes(x = body_mass_g,  
4       color = species )  
5 ) +  
6   geom_histogram(bins = 20)
```



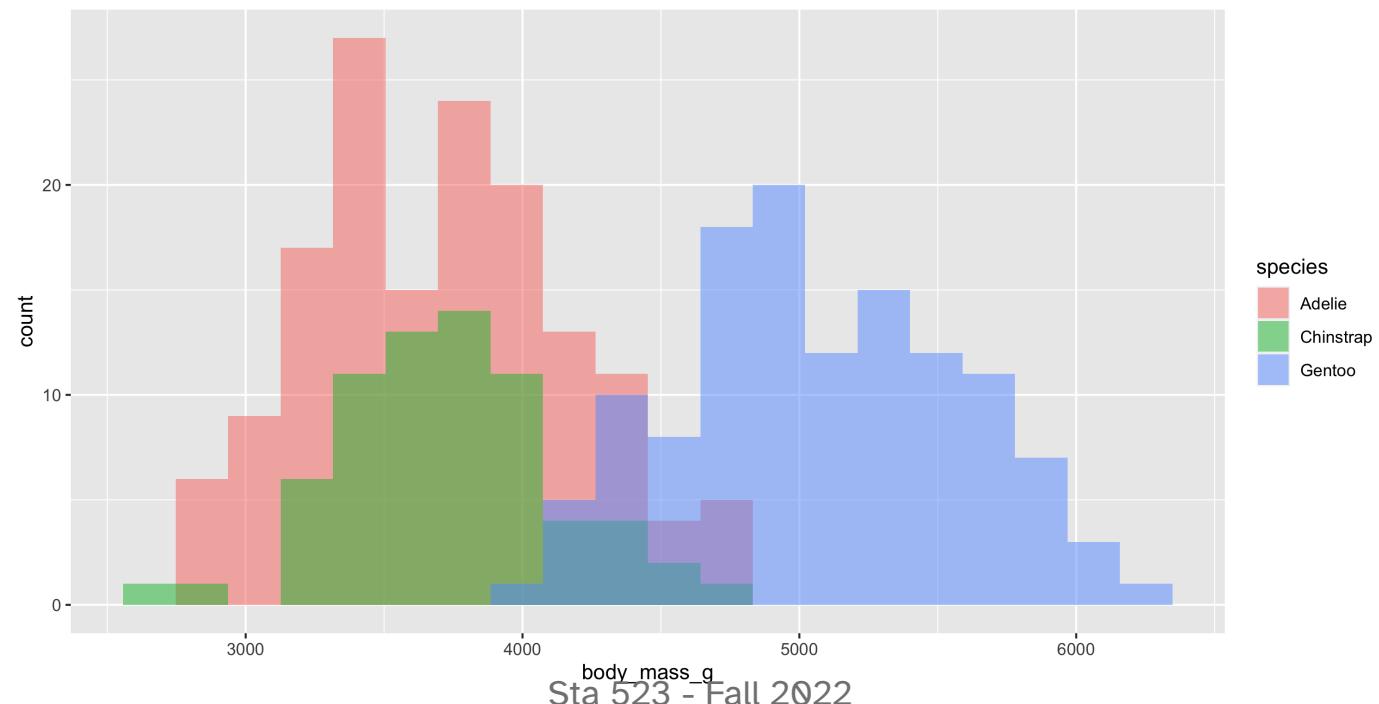
# Histograms - fill

```
1 ggplot(  
2   penguins,  
3   aes(x = body_mass_g,  
4       fill = species )  
5 ) +  
6   geom_histogram(bins = 20)
```



# Histograms - position

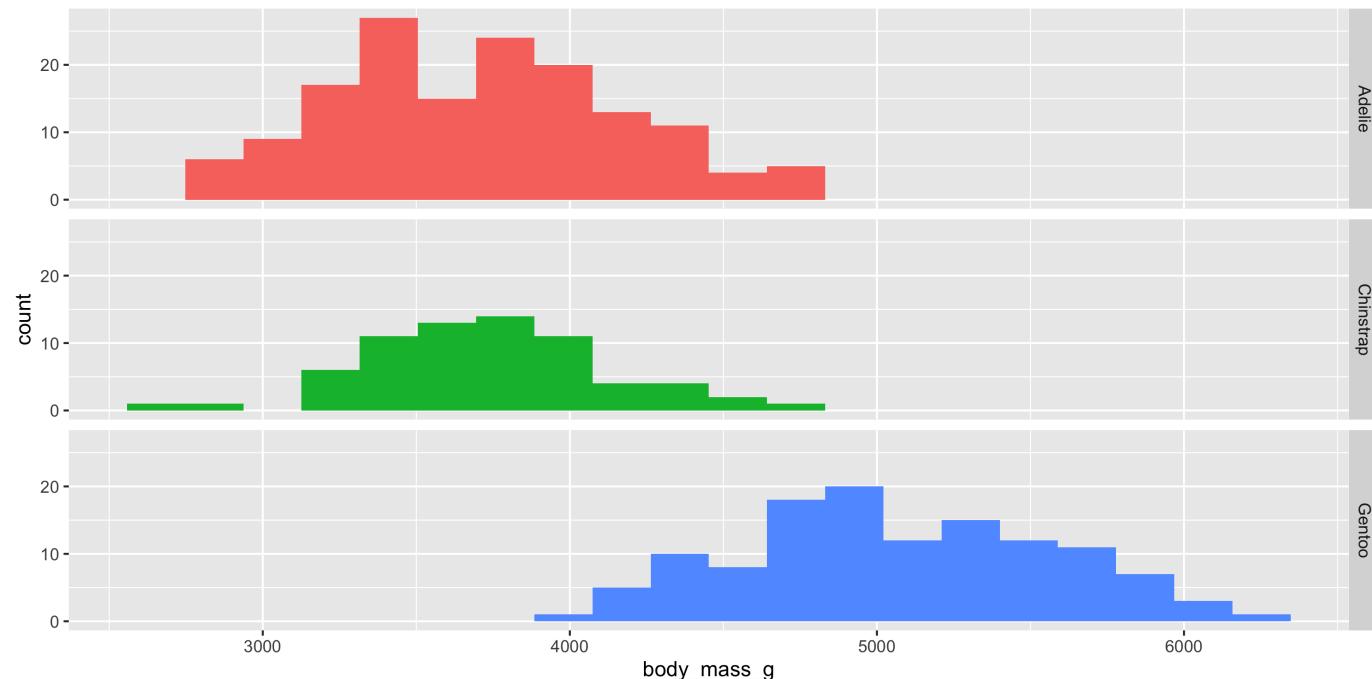
```
1 ggplot(  
2   penguins,  
3   aes(x = body_mass_g,  
4       fill = species ))  
5 ) +  
6   geom_histogram(  
7     bins = 20, alpha = 0.5,  
8     position = "identity"  
9   )
```





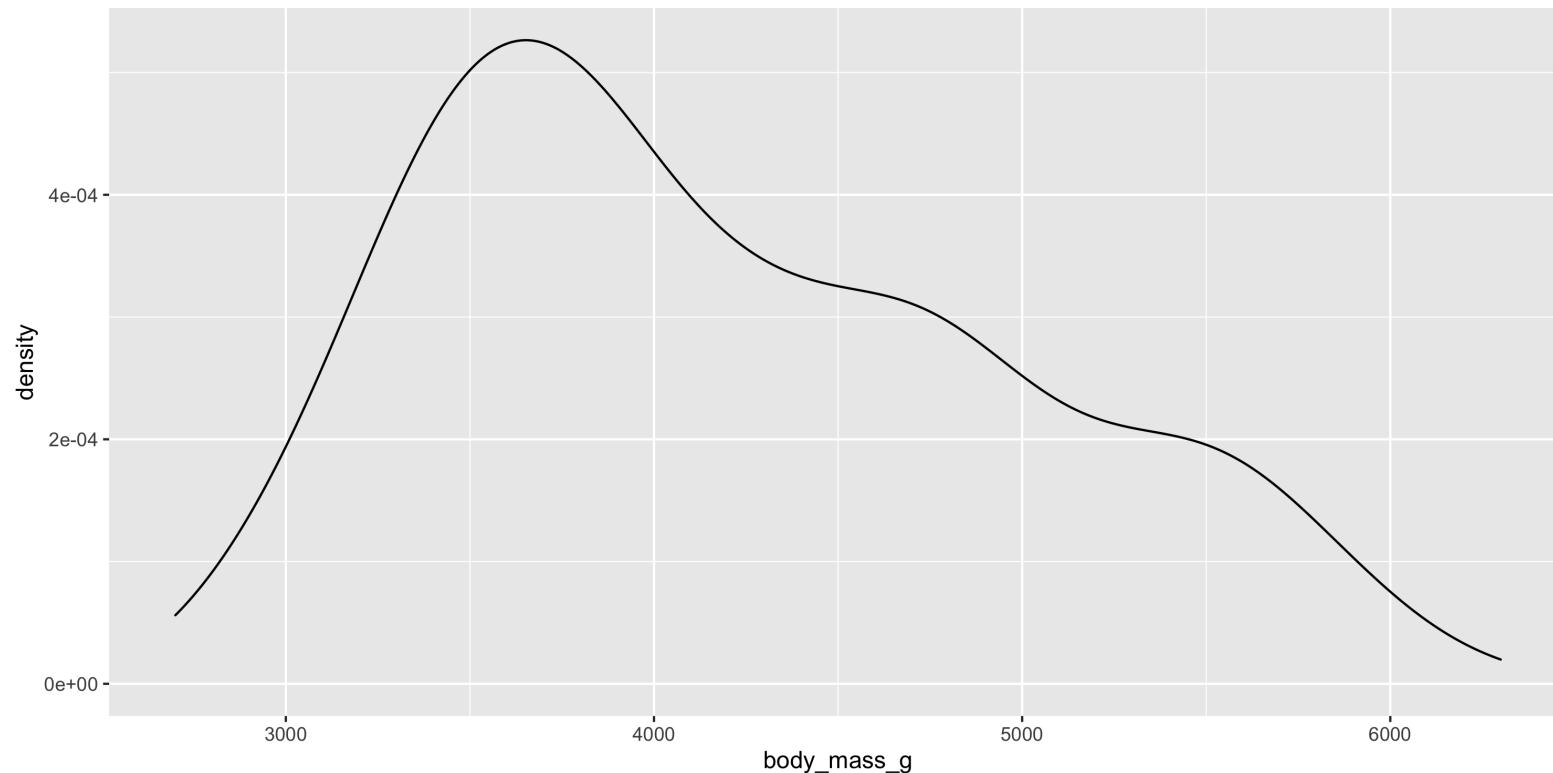
# Histograms - facets

```
1 ggplot(  
2   penguins,  
3   aes(x = body_mass_g,  
4         fill = species )  
5 ) +  
6   geom_histogram(bins = 20) +  
7   facet_grid(species ~ .) +  
8   guides(fill = "none")
```



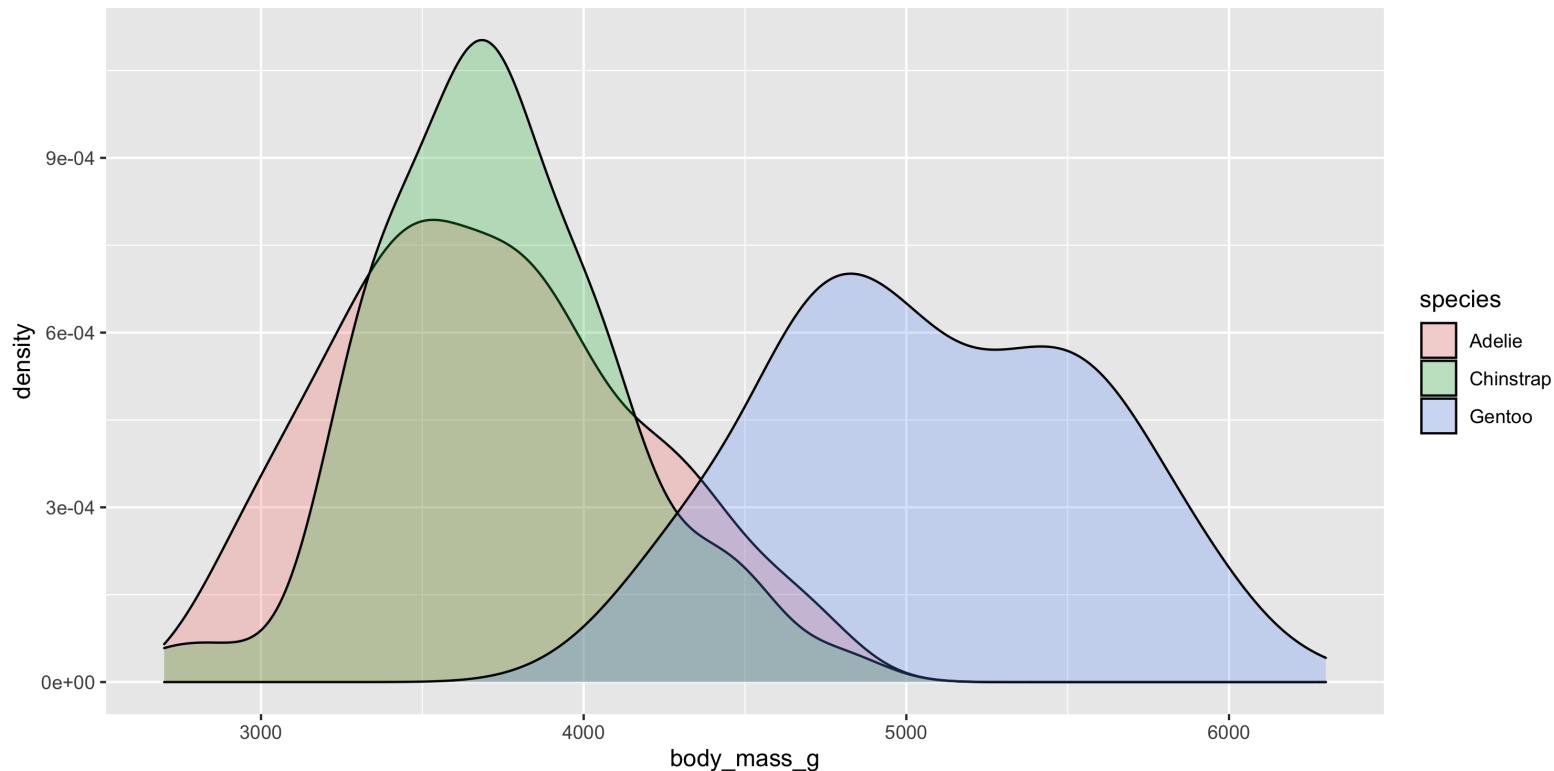
# Density plot

```
1 ggplot(  
2   penguins,  
3   aes(x = body_mass_g)  
4 ) +  
5   geom_density()
```



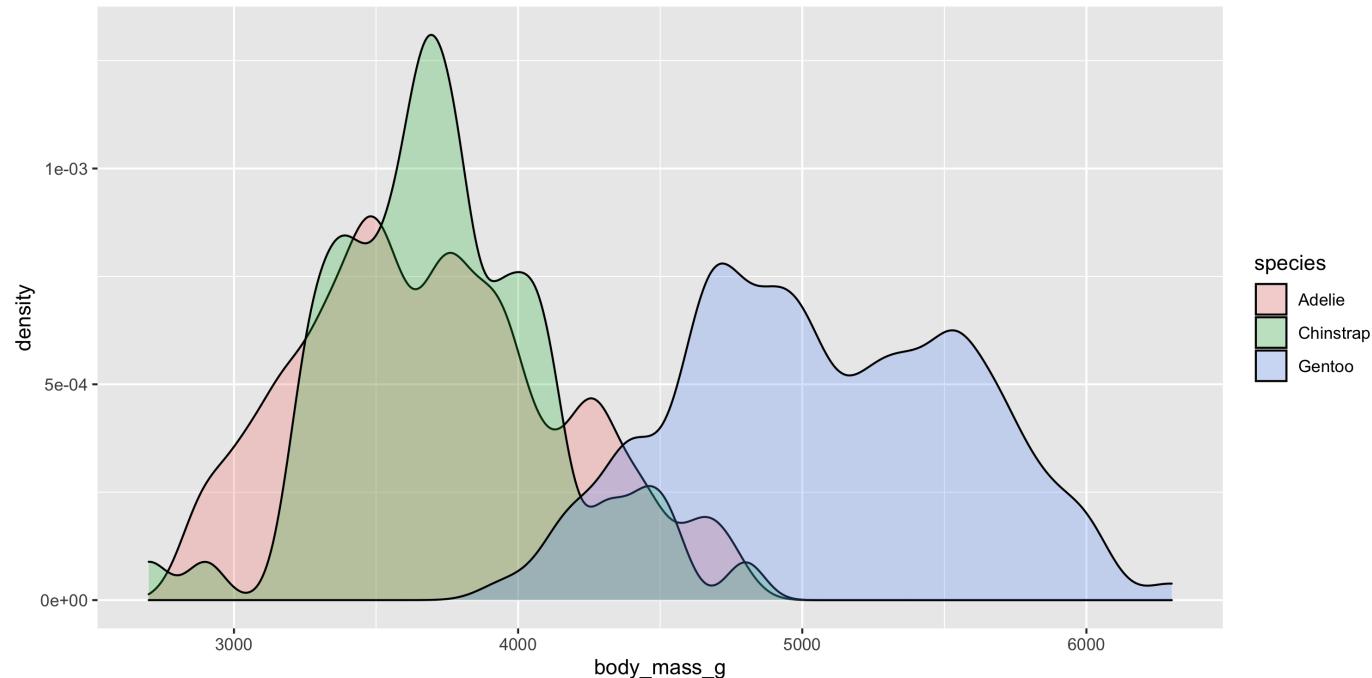
# Density plot - fill

```
1 ggplot(  
2   penguins,  
3   aes(x = body_mass_g,  
4       fill = species ))  
5 ) +  
6 geom_density(alpha = 0.25)
```



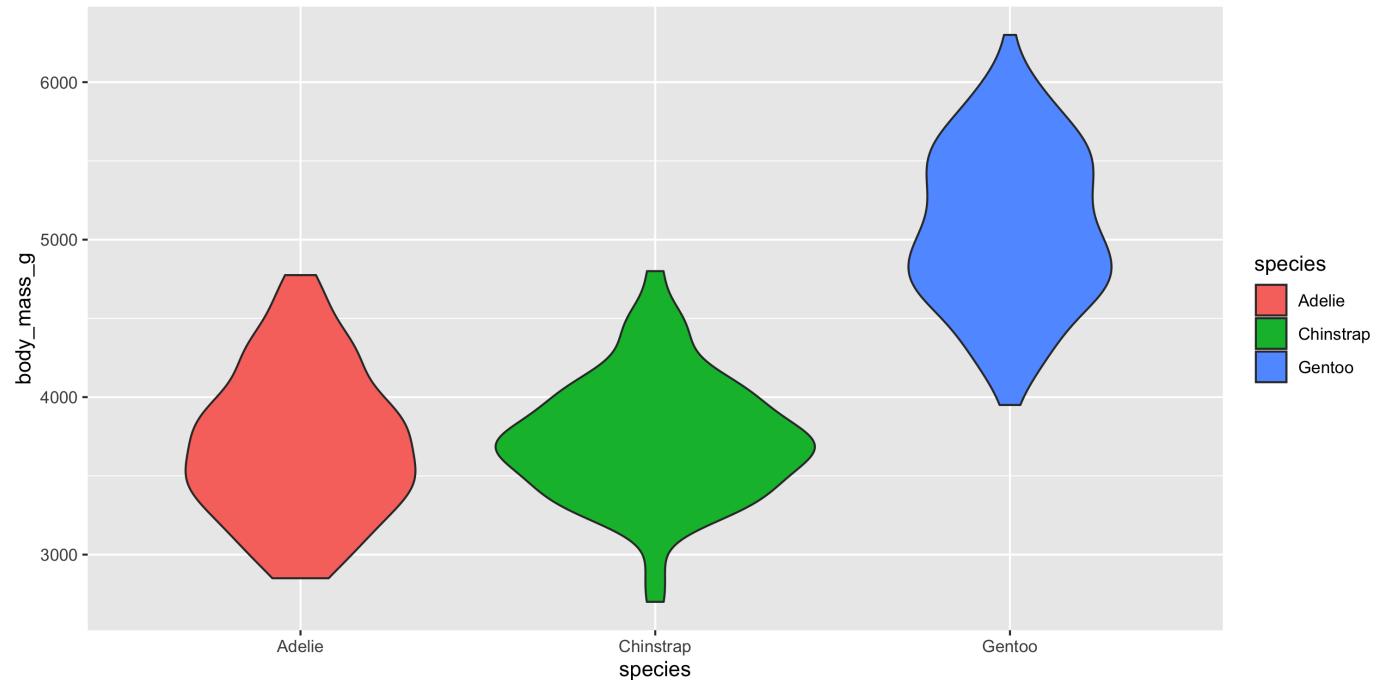
# Density plot - adjust

```
1 ggplot(  
2   penguins,  
3   aes(x = body_mass_g,  
4       fill = species ))  
5 ) +  
6   geom_density(  
7     adjust = 0.5, alpha = 0.25  
8 )
```



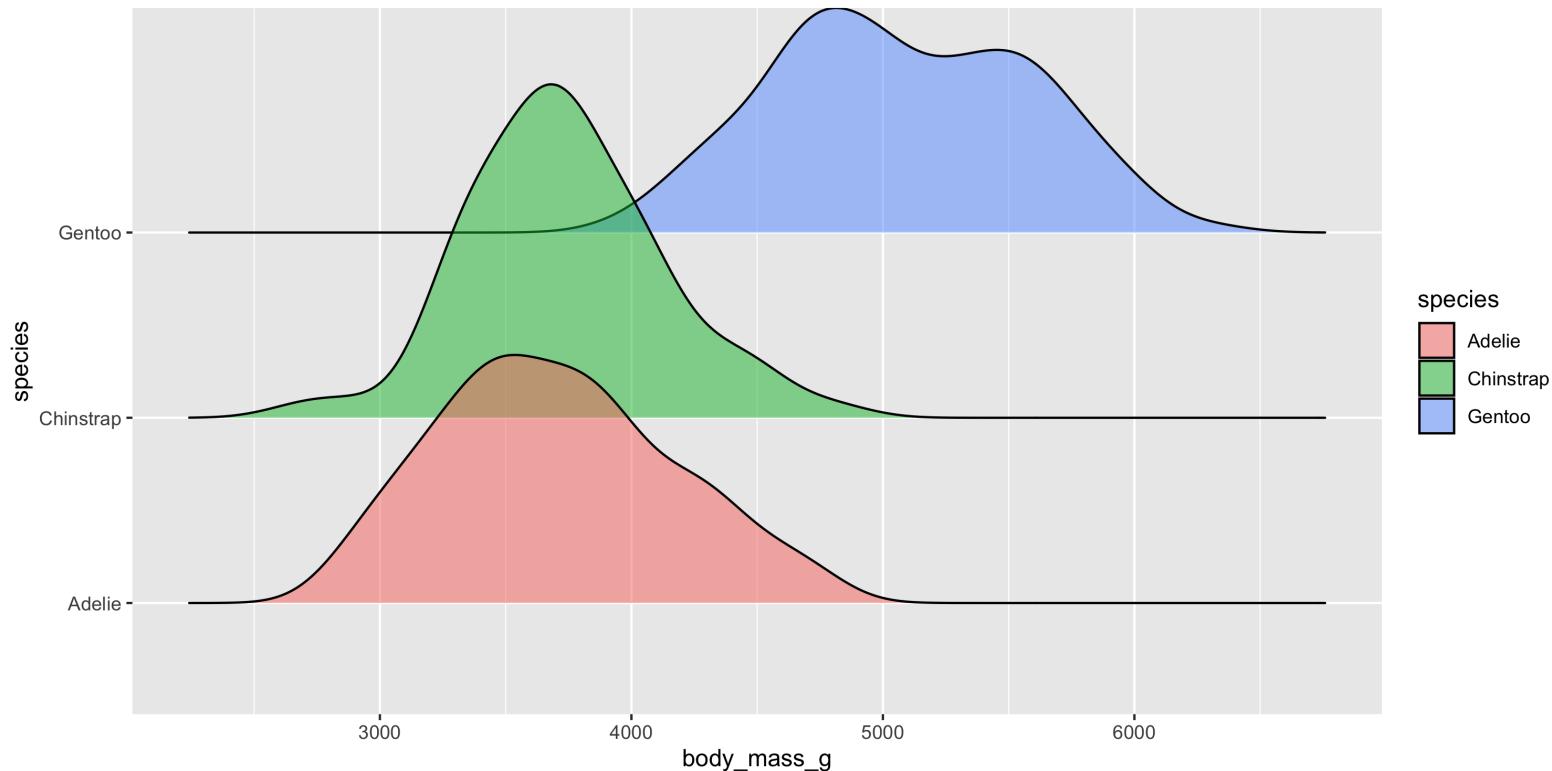
# Violin plot

```
1 ggplot(penguins,
2   aes(x = species, fill = species,
3       y = body_mass_g )
4 ) +
5   geom_violin()
```



# Ridge plot

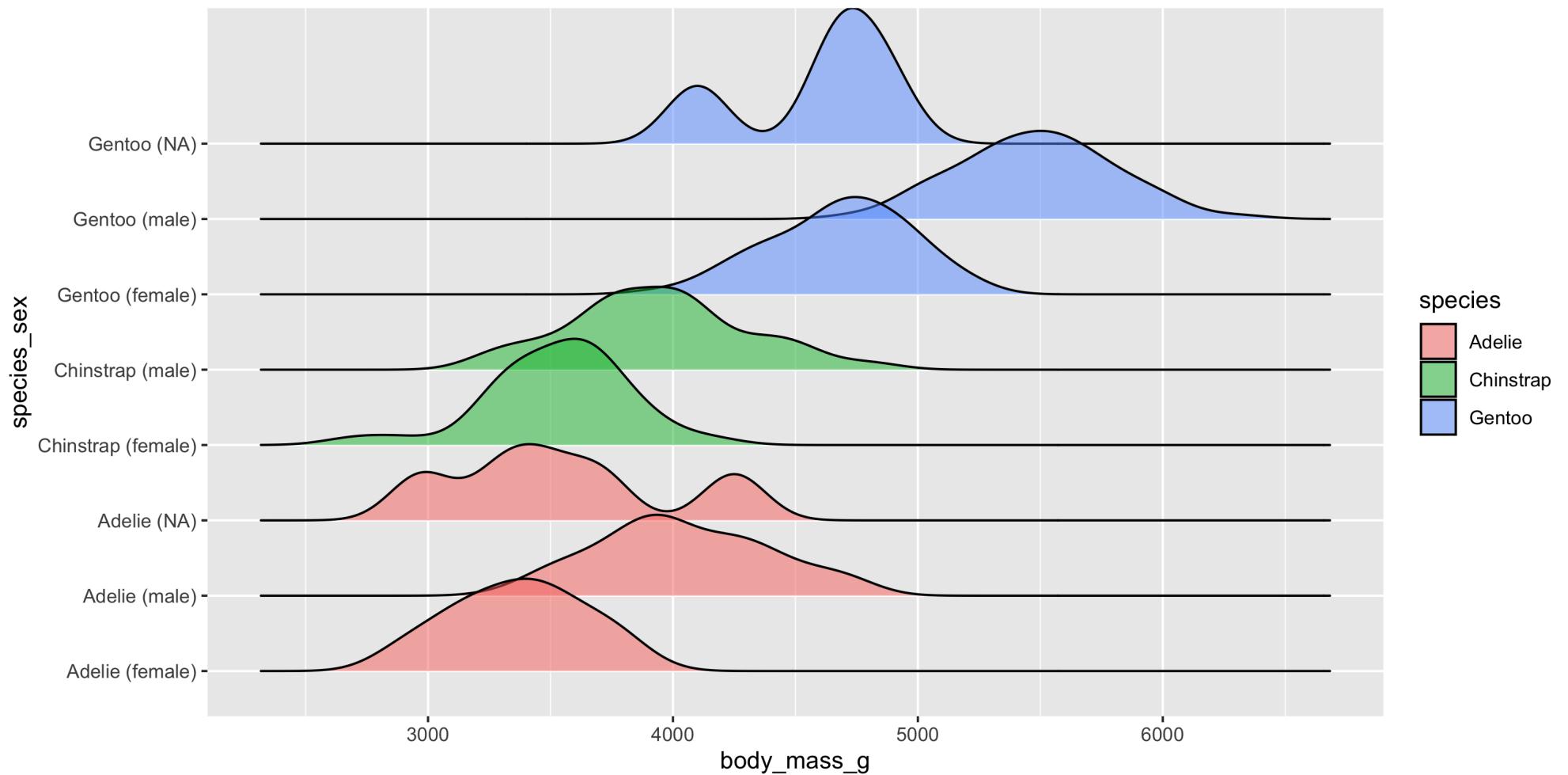
```
1 ggplot(  
2   penguins,  
3   aes(x = body_mass_g,  
4       y = species, fill = species ))  
5 ) +  
6   ggridges::geom_density_ridges(alpha = 0.5)
```



# Ridge plot - more categories + dplyr

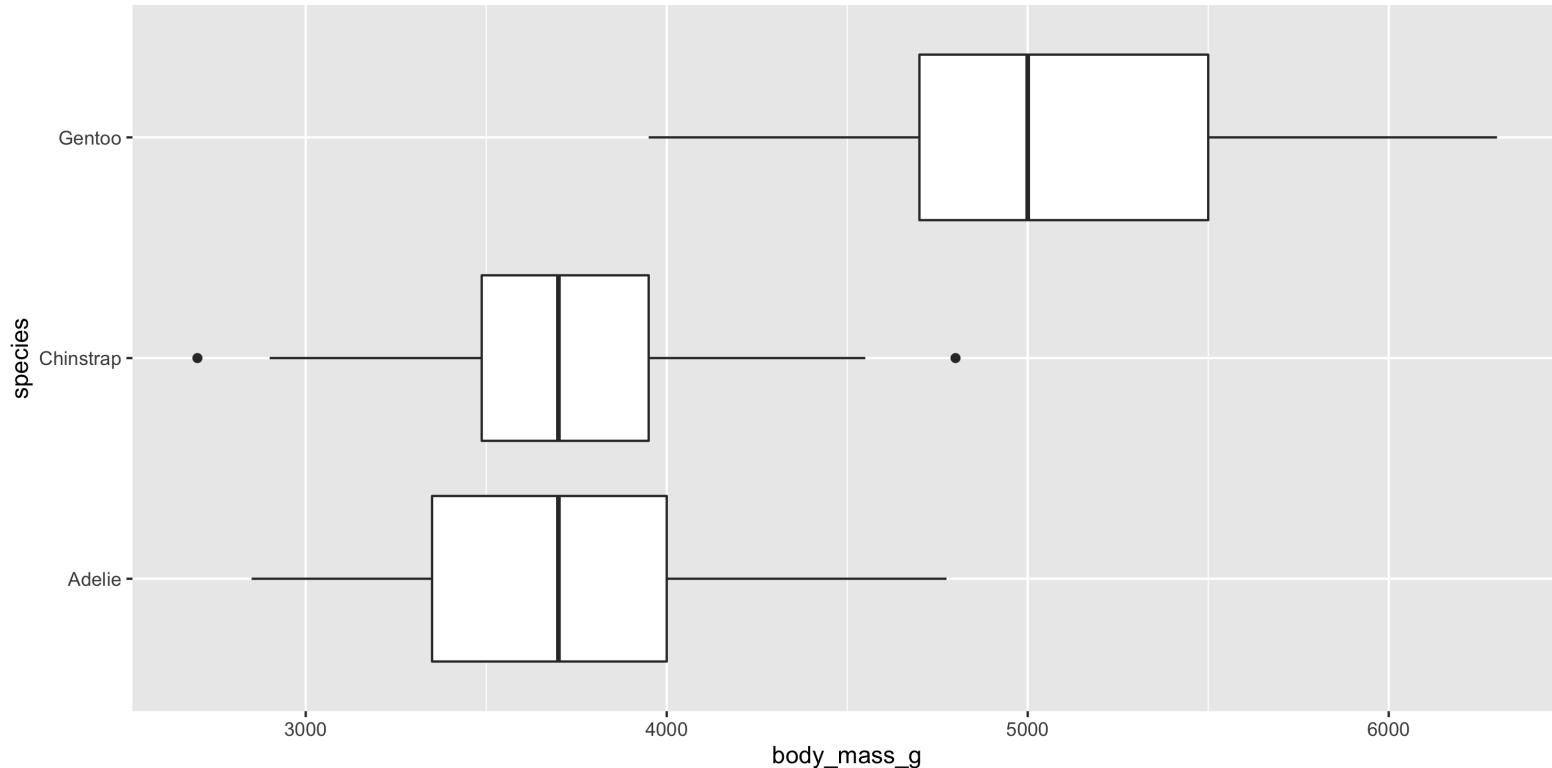
```
1 penguins %>%
2   mutate(
3     species_sex = paste0(
4       species, " (", sex, ")"
5     )
6   ) %>%
7   ggplot(
8     aes(
9       x = body_mass_g,
10      y = species_sex,
11      fill = species
12    )
13  ) +
14  ggridges::geom_density_ridges(
15    alpha = 0.5
16  )
```

# Ridge plot - more categories + dplyr



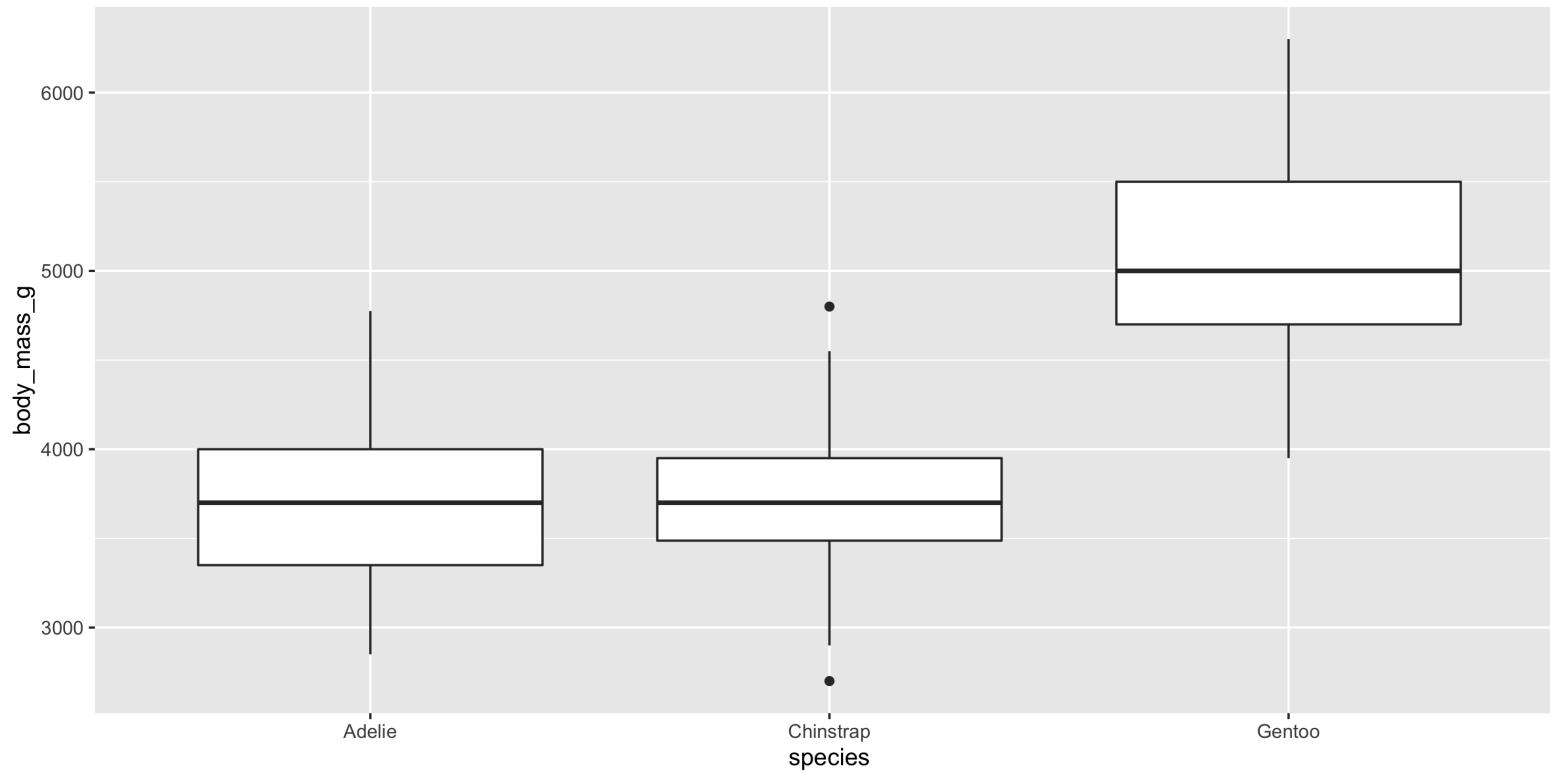
# Box plot

```
1 ggplot(  
2   penguins,  
3   aes(x = body_mass_g,  
4       y = species )  
5 ) +  
6   geom_boxplot()
```



# Box plot - coord\_flip

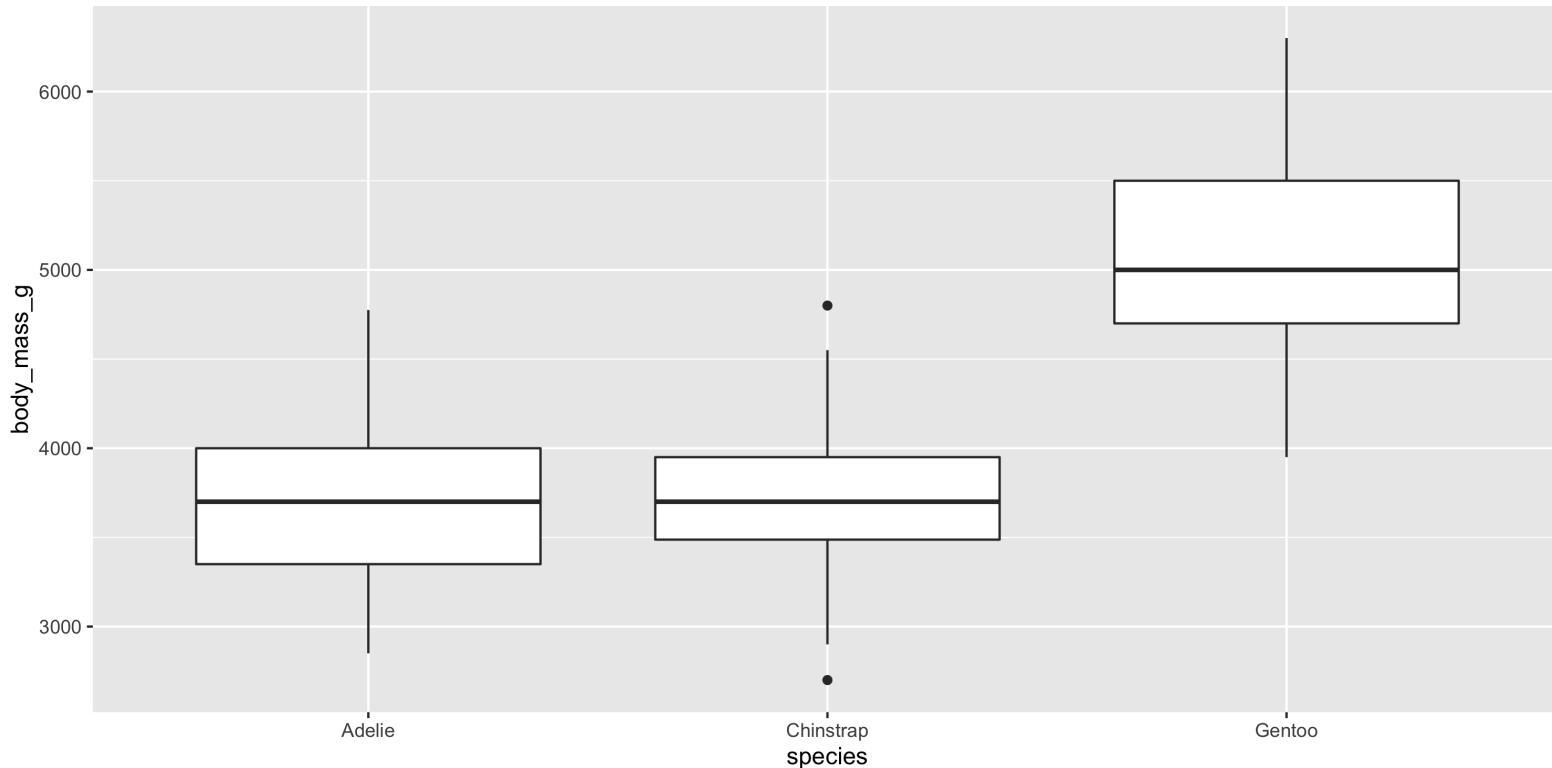
```
1 ggplot(  
2   penguins,  
3   aes(x = body_mass_g,  
4       y = species )  
5 ) +  
6   geom_boxplot() +  
7   coord_flip()
```





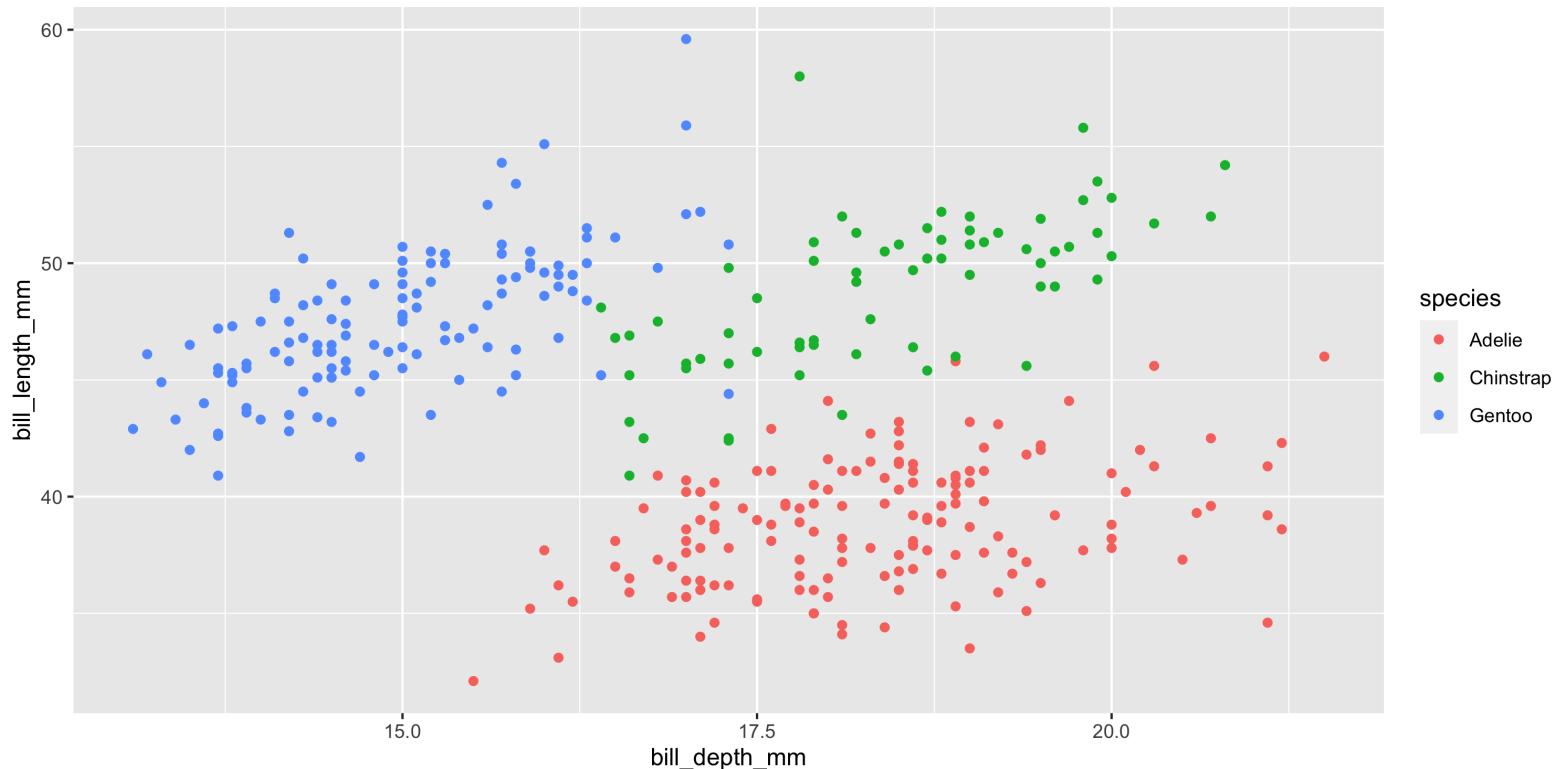
# Box plot - swap coords

```
1 ggplot(  
2   penguins,  
3   aes(x = species,  
4       y = body_mass_g )  
5 ) +  
6   geom_boxplot()
```



# Scatter plot

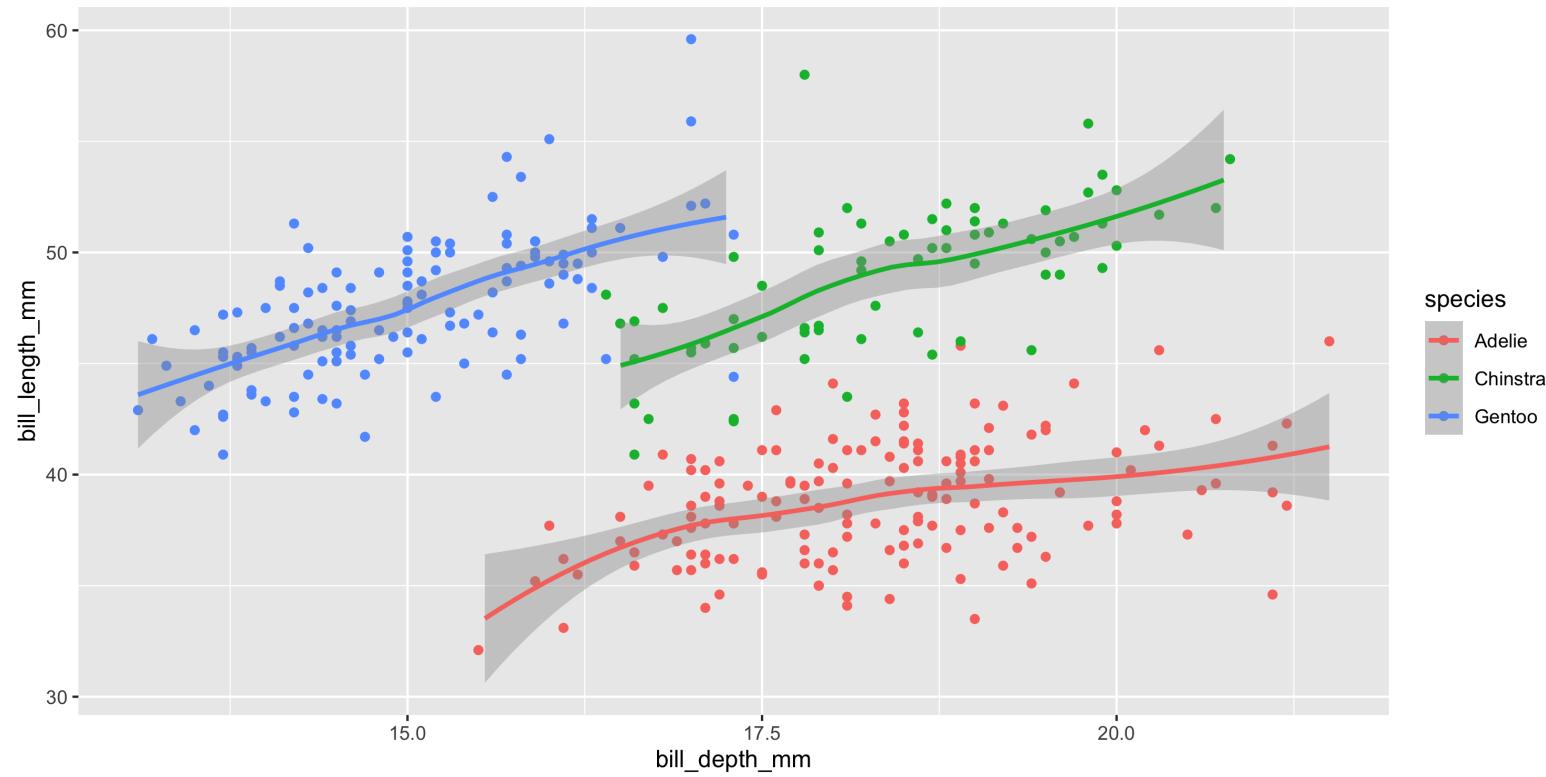
```
1 ggplot(  
2   penguins,  
3   aes(x = bill_depth_mm,  
4       y = bill_length_mm,  
5       color = species )  
6 ) +  
7   geom_point()
```





# Scatter plot - geom\_smooth

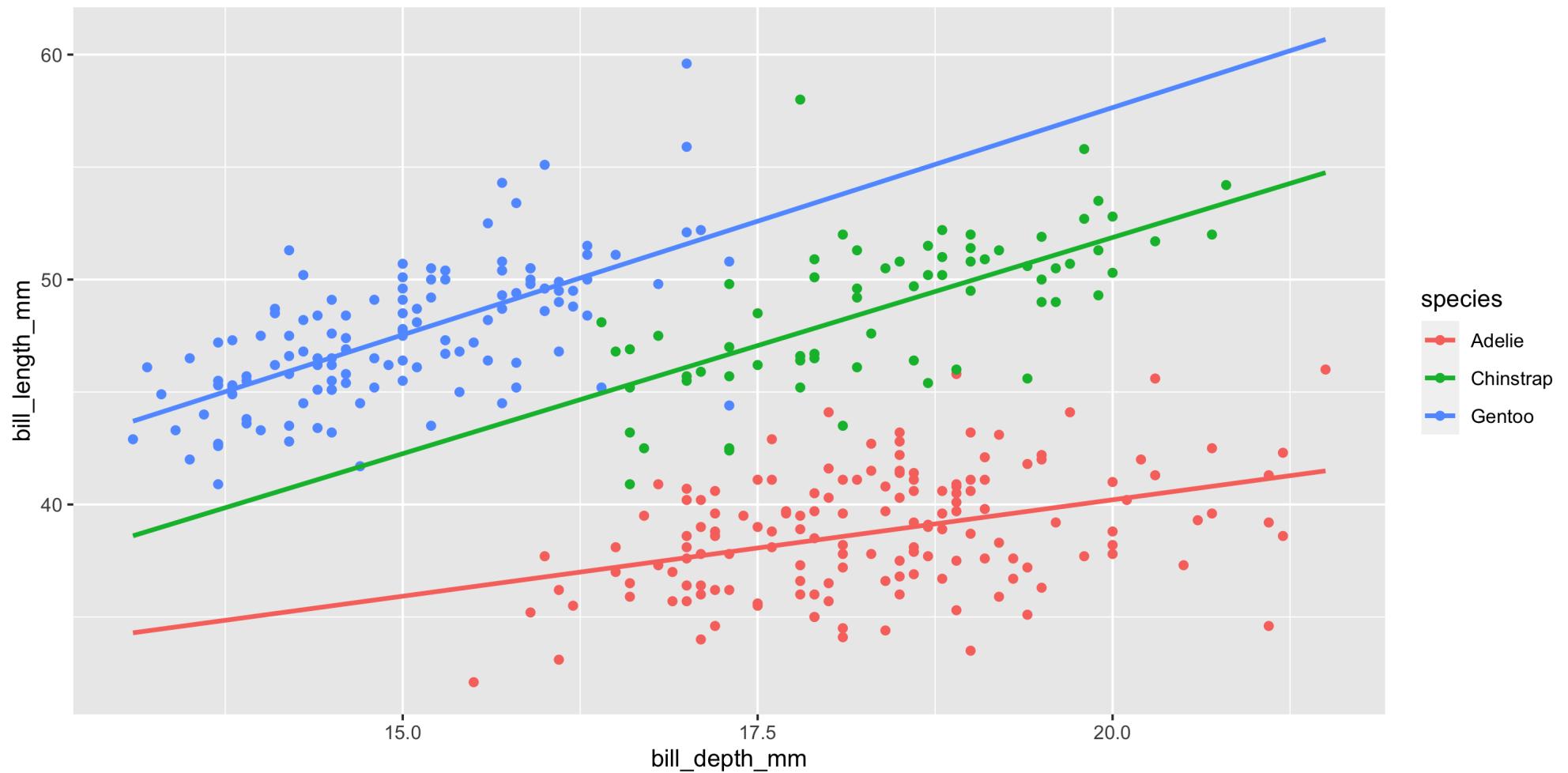
```
1 ggplot(  
2   penguins,  
3   aes(x = bill_depth_mm,  
4       y = bill_length_mm,  
5       color = species )  
6 ) +  
7   geom_point() +  
8   geom_smooth( fullrange = TRUE )
```



# Scatter plot - geom\_smooth w/ lm

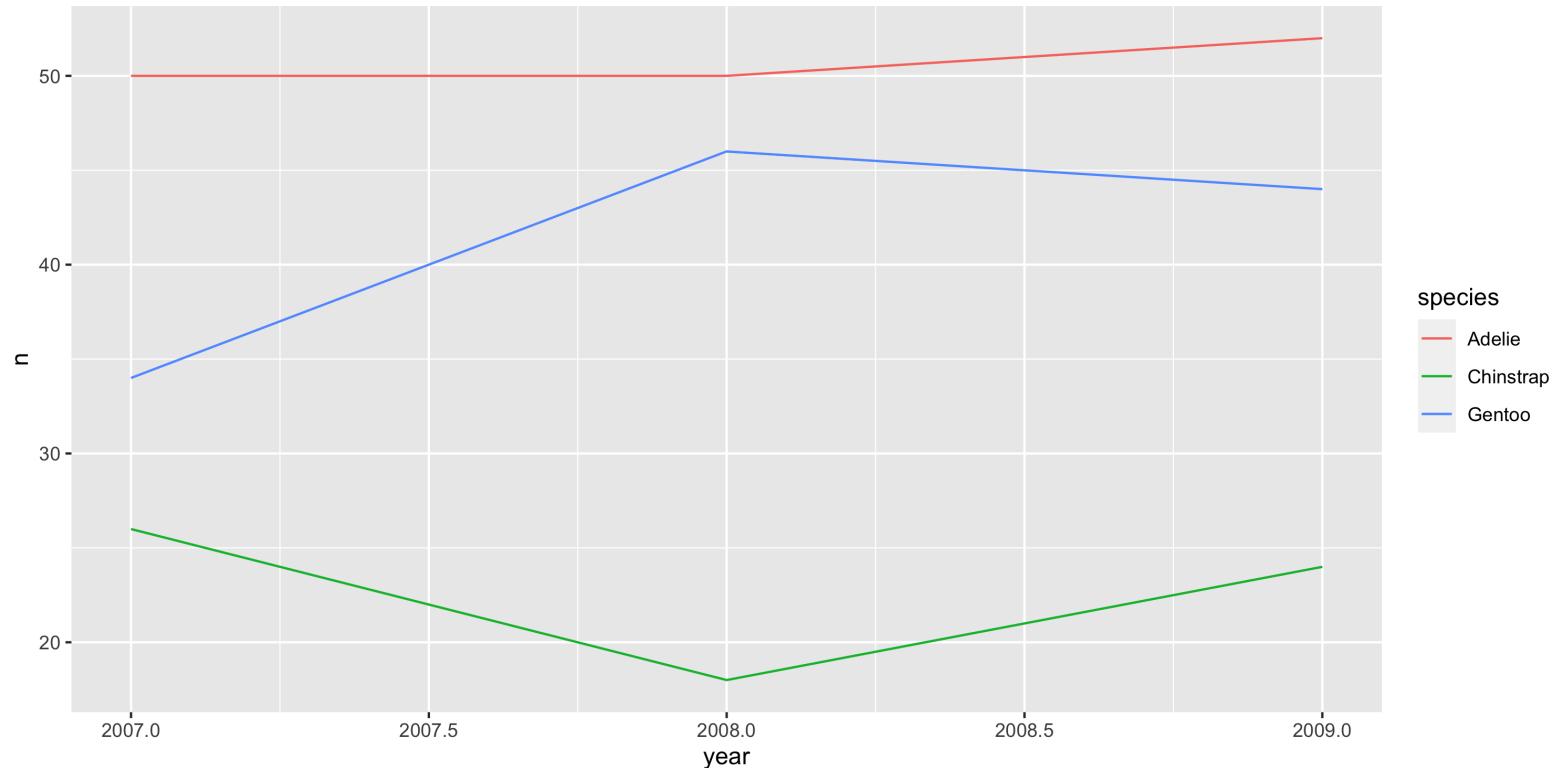
```
1 ggplot(  
2   penguins,  
3   aes(  
4     x = bill_depth_mm,  
5     y = bill_length_mm,  
6     color = species  
7   )  
8 ) +  
9   geom_point() +  
10  geom_smooth(  
11    method = "lm",  
12    se = FALSE,  
13    fullrange = TRUE  
14 )
```

# Scatter plot - geom\_smooth w/ lm



# Line plot

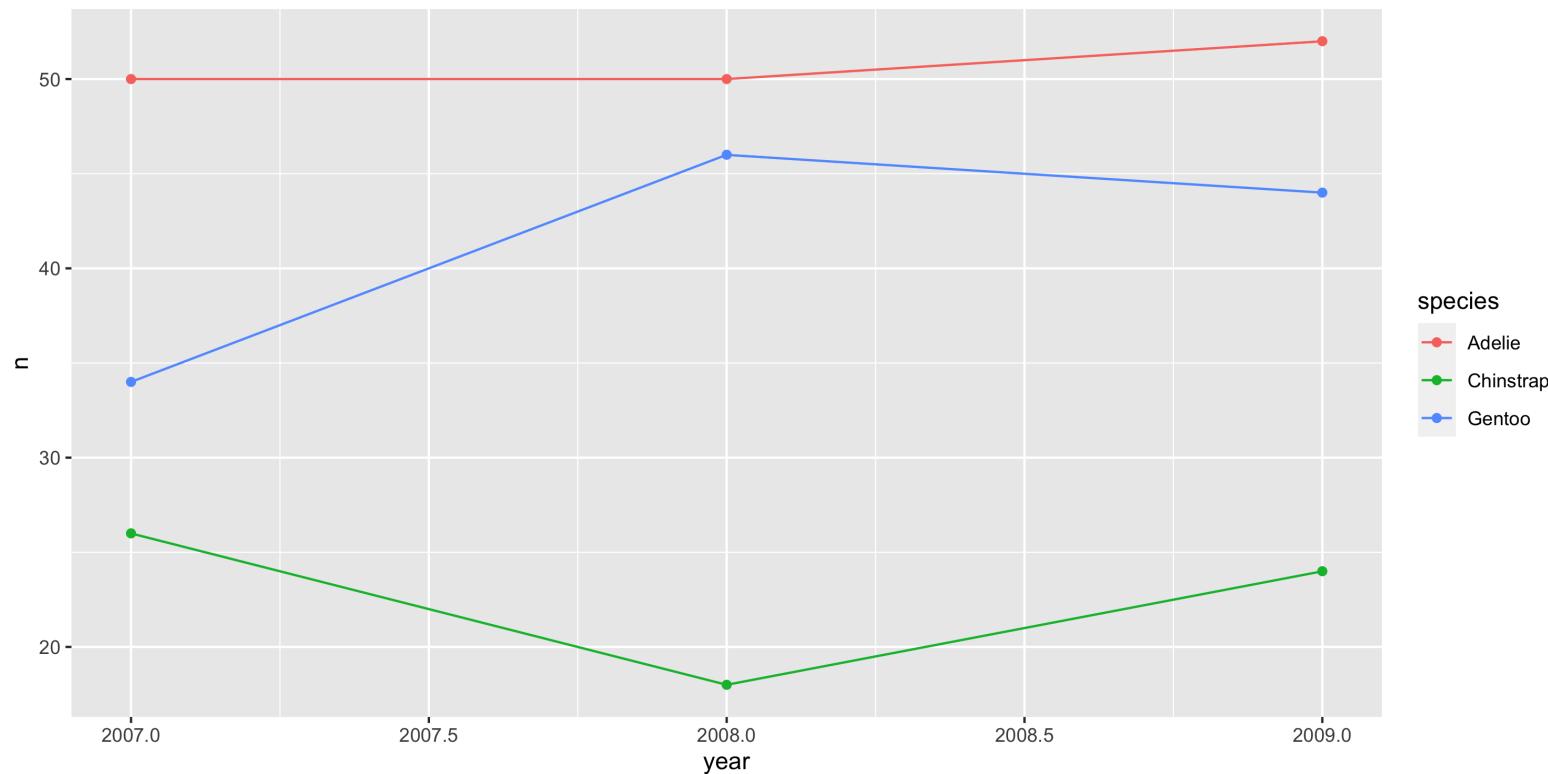
```
1 penguins %>%
2   count(species, year) %>%
3   ggplot(
4     aes(x = year, y = n,
5       color = species, group = species )
6   ) +
7   geom_line()
```





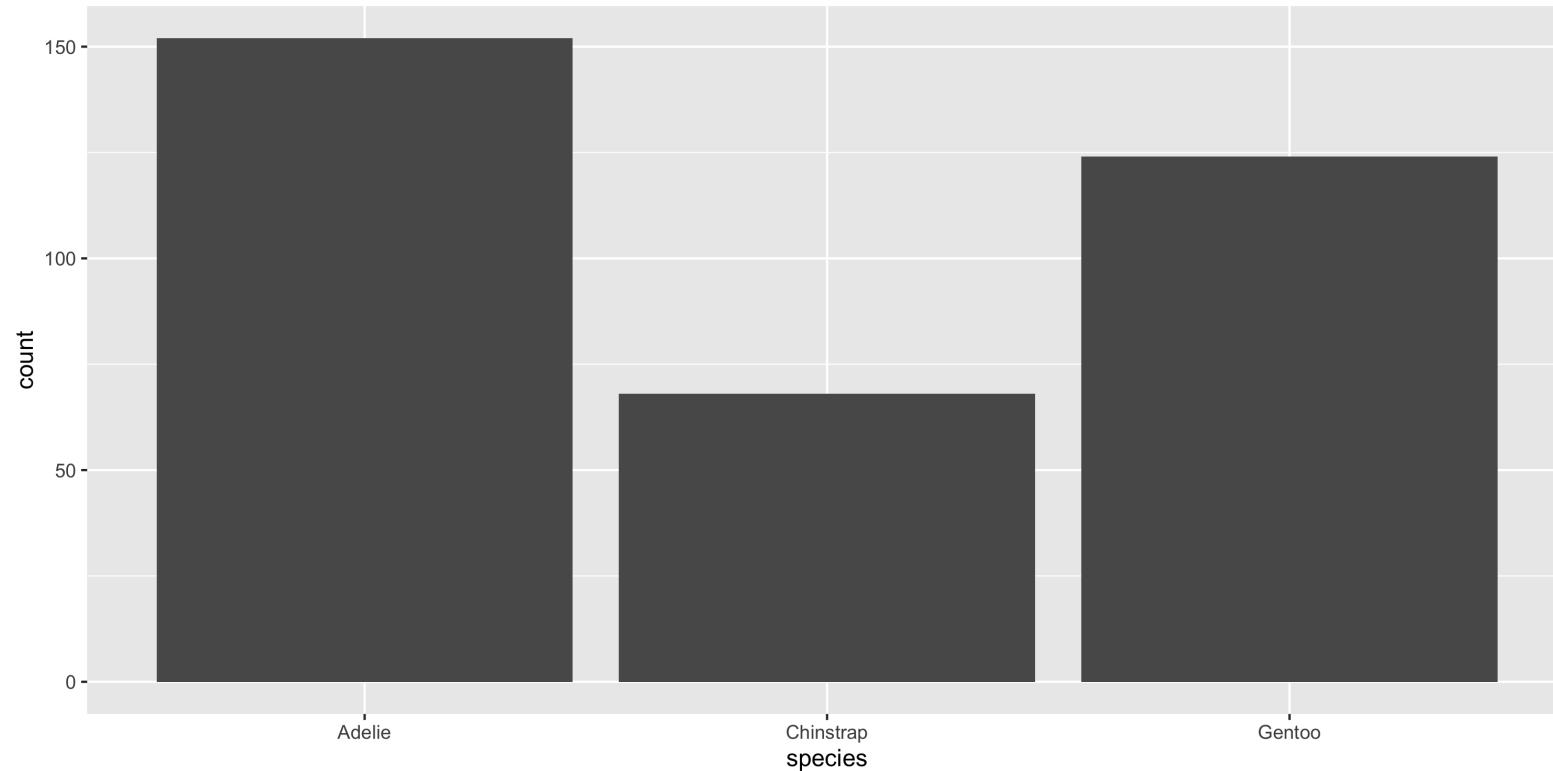
# Line plot - with points

```
1 penguins %>%
2   count(species, year) %>%
3   ggplot(
4     aes(x = year, y = n,
5          color = species, group = species )
6   ) +
7   geom_line() +
8   geom_point()
```



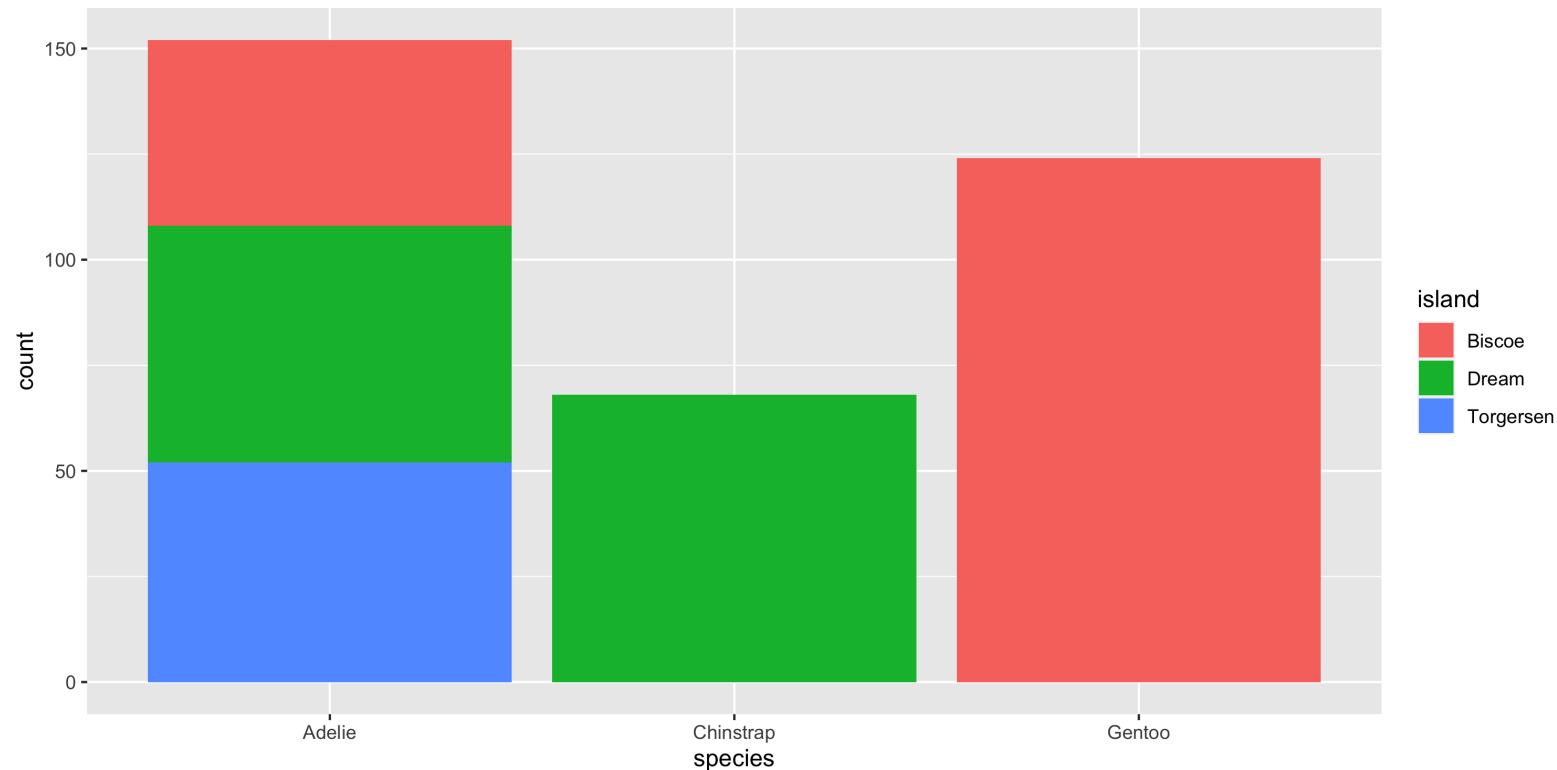
# Bar plot

```
1 ggplot(  
2   penguins,  
3   aes( x = species )  
4 ) +  
5   geom_bar()
```



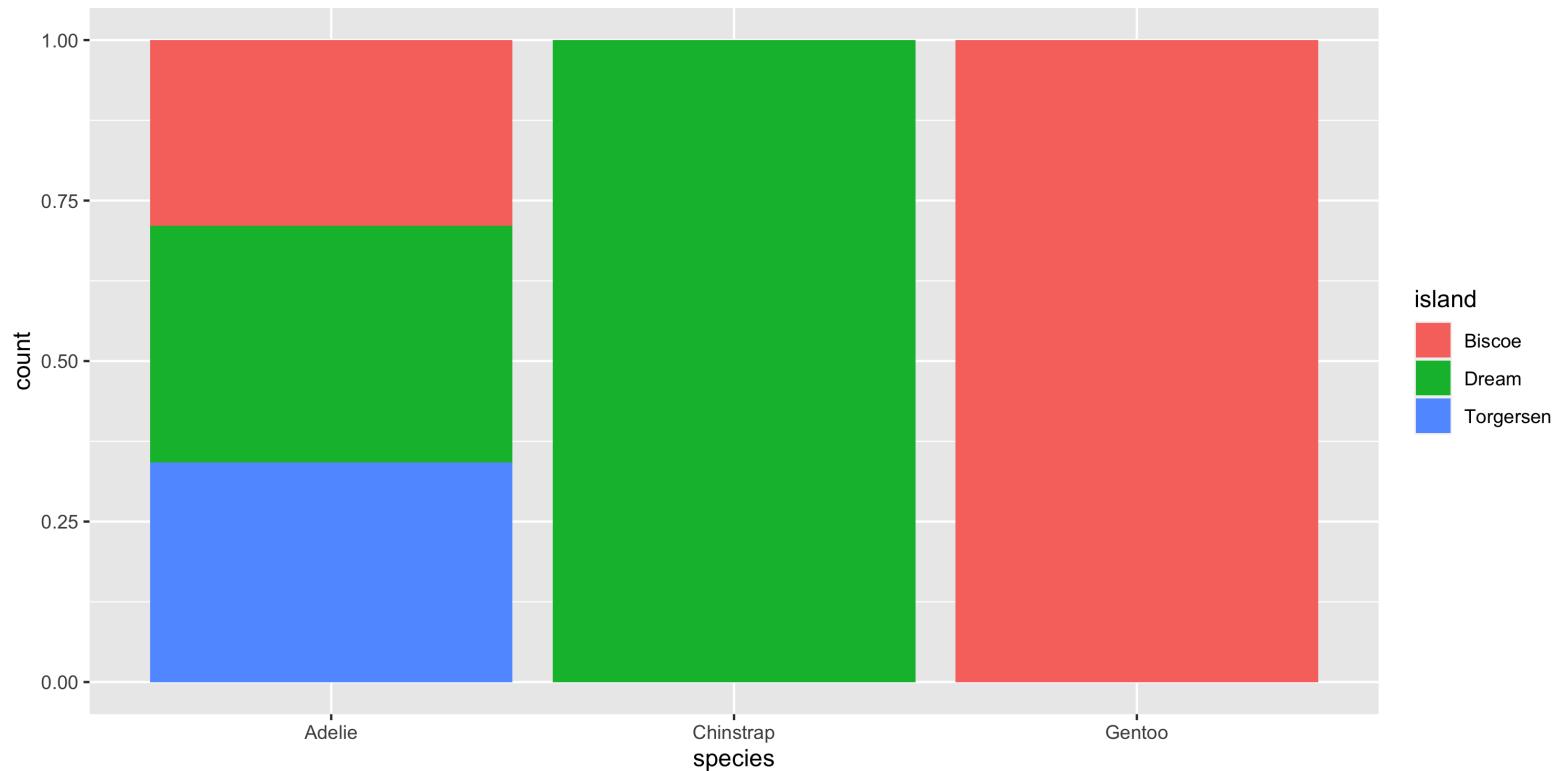
# Stacked bar plot

```
1 ggplot(  
2   penguins,  
3   aes( x = species,  
4         fill = island )  
5 ) +  
6   geom_bar()
```



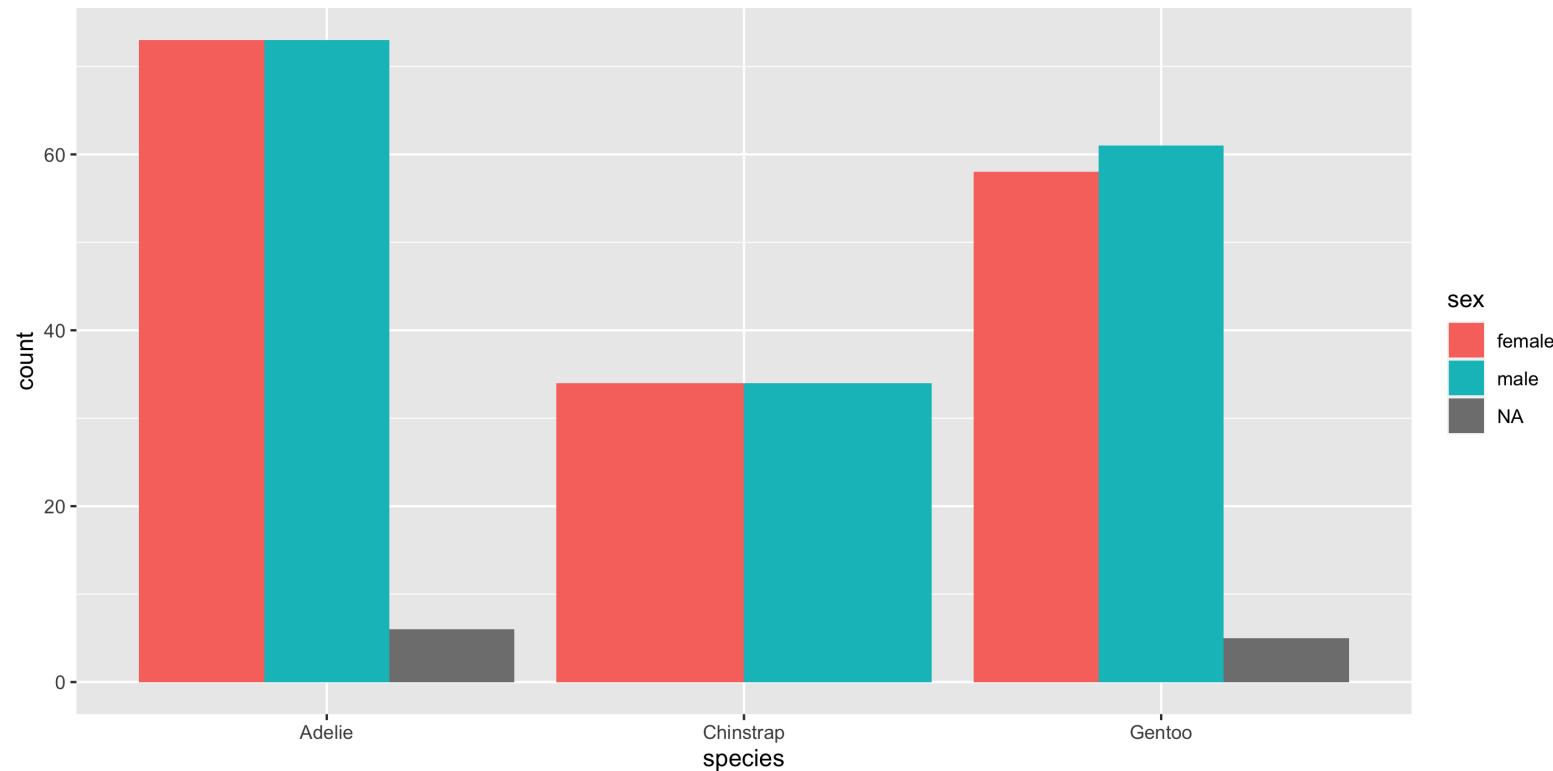
# Stacked relative frequency bar plot

```
1 ggplot(  
2   penguins,  
3   aes( x = species,  
4         fill = island )  
5 ) +  
6   geom_bar(position = "fill")
```



# Dodged bar plot

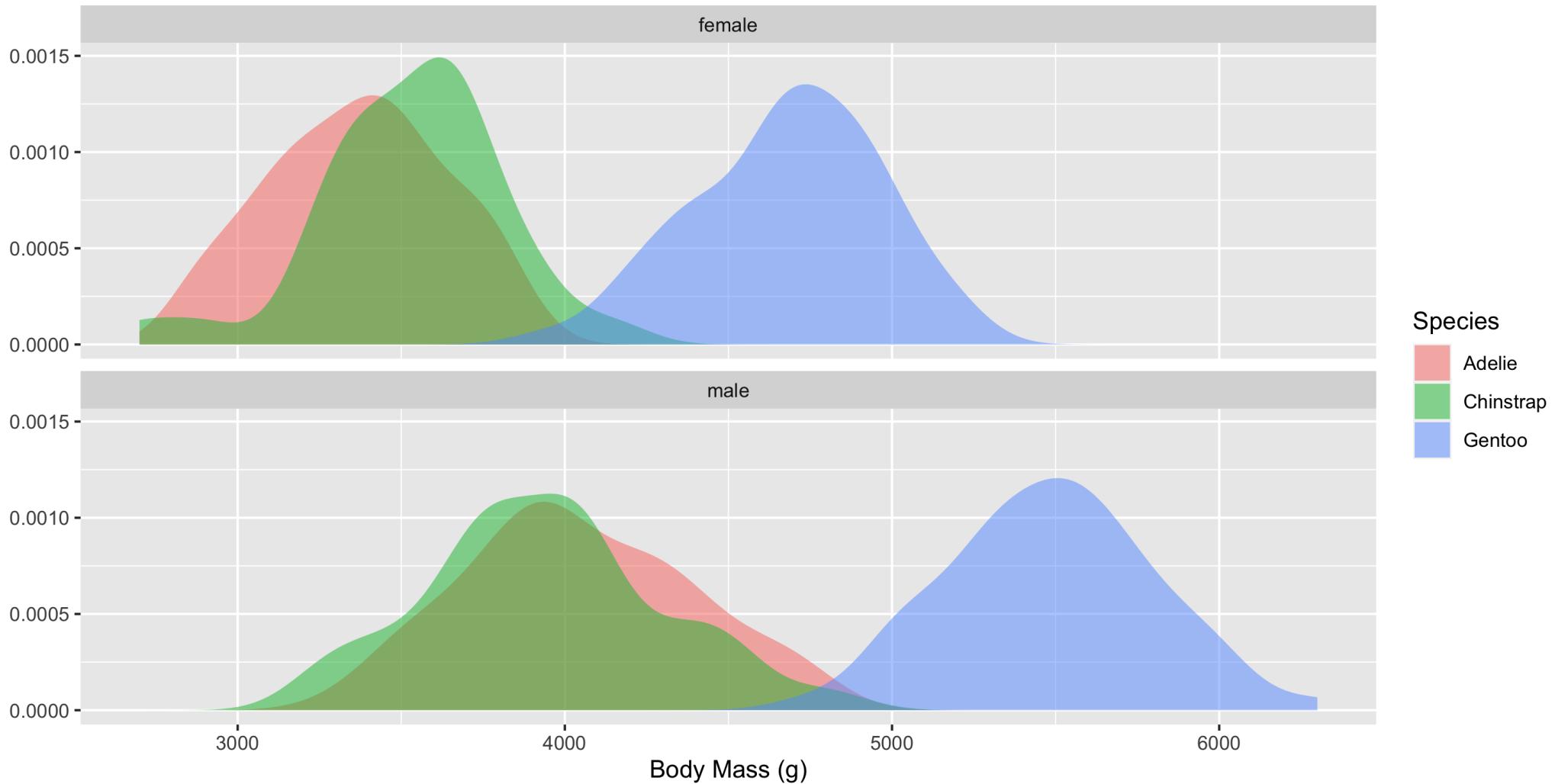
```
1 ggplot(  
2   penguins,  
3   aes( x = species,  
4         fill = sex )  
5 ) +  
6   geom_bar(position = "dodge")
```



# Exercises

# Exercise 1

Recreate, as faithfully as possible, the following plot using ggplot2 and the `penguins` data.



# Exercise 2

Recreate, as faithfully as possible, the following plot from the [palmerpenguin](#) package readme in ggplot2.

