ggplot2 Tutorial

Jianhui Gao

2023-07-13

Prepare the data

```
# Install the package
install.packages("palmerpenguins")
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.4.2     v purrr   1.0.1
## v tibble  3.2.1     v dplyr   1.1.2
## v tidyr   1.3.0     v stringr 1.5.0
## v readr   2.1.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(palmerpenguins)
```

```
# check the data
?penguins
```

```
# look at first few rows
head(penguins)
```

```
## # A tibble: 6 x 8
##   species island    bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
##   <fct>   <fct>              <dbl>         <dbl>             <int>       <int>
## 1 Adelie  Torgersen           39.1          18.7               181        3750
## 2 Adelie  Torgersen           39.5          17.4               186        3800
## 3 Adelie  Torgersen           40.3          18                 195        3250
## 4 Adelie  Torgersen           NA            NA                  NA          NA
## 5 Adelie  Torgersen           36.7          19.3               193        3450
## 6 Adelie  Torgersen           39.3          20.6               190        3650
## # i 2 more variables: sex <fct>, year <int>
```

```
View(penguins)
```

```
summary(penguins$species)
```

```
##    Adelie Chinstrap    Gentoo
##       152        68       124
```

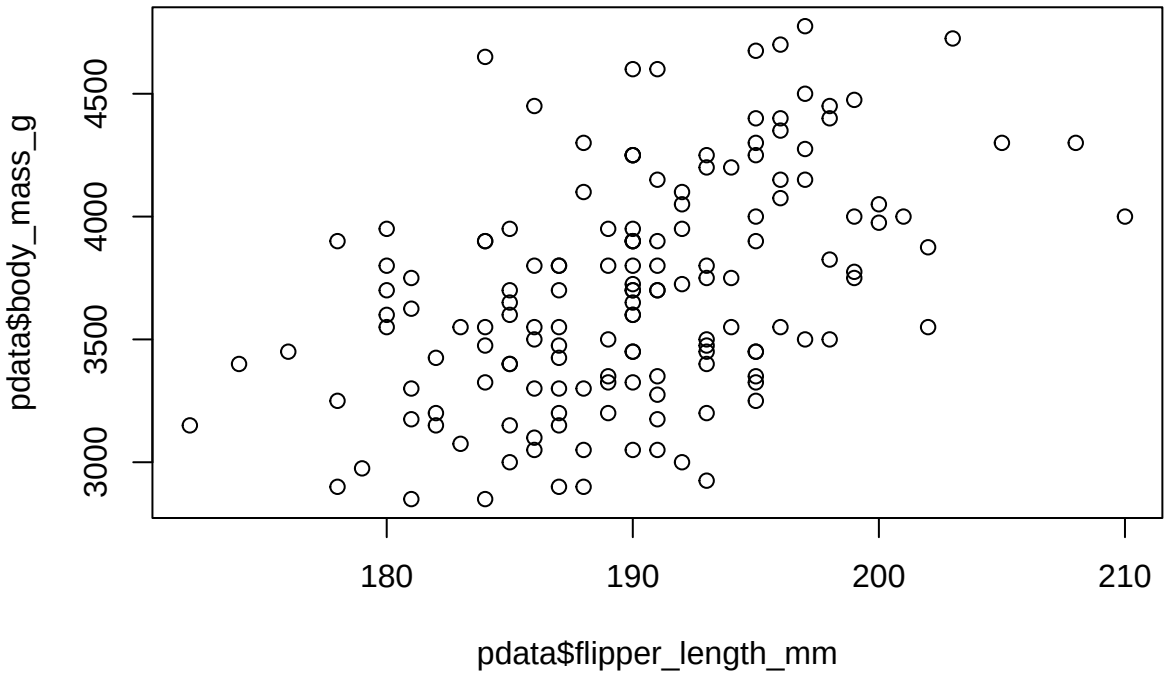
Scatter Plot

Task 1: A scatter plot of flipper length and body mass for species = "Adelie"

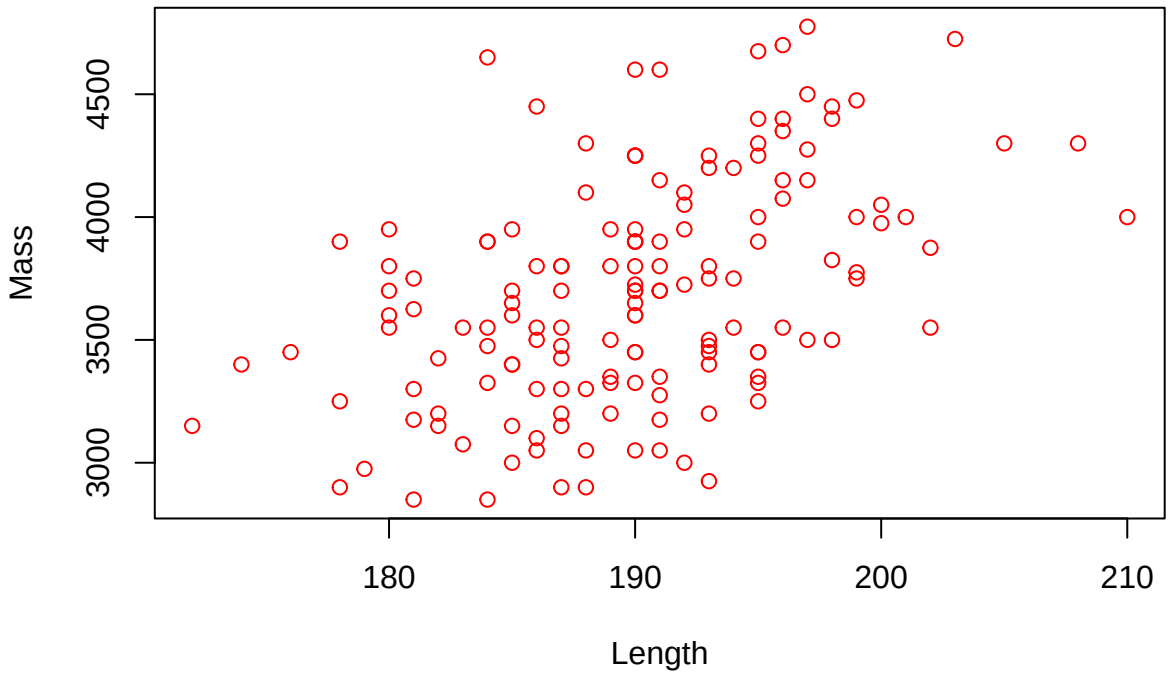
```r
# prepare the subset of data
## Generation X style
pdata <- penguins[penguins$species == "Adelie", ]

## Generation Z style
pdata <- penguins %>% filter(species == "Adelie")
```

```r
# Quick plot using basic R
plot(x = pdata$flipper_length_mm, y = pdata$body_mass_g)
```



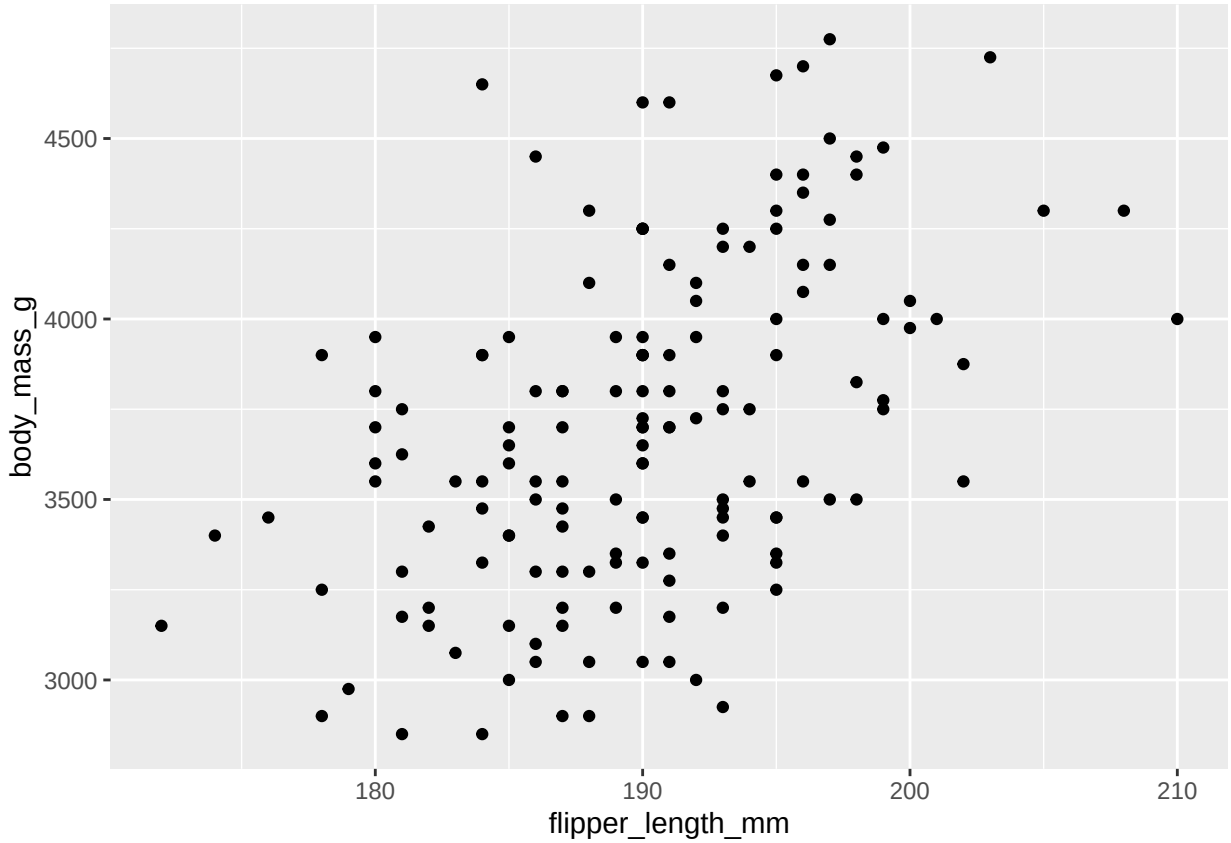

```r
# change x-axis and y-axis labels
plot(x = pdata$flipper_length_mm, y = pdata$body_mass_g, xlab = "Length", ylab = "Mass", col = 'red')
```




```
# Using ggplot instead
penguins %>%
  filter(species == "Adelie") %>%
  ggplot()
```

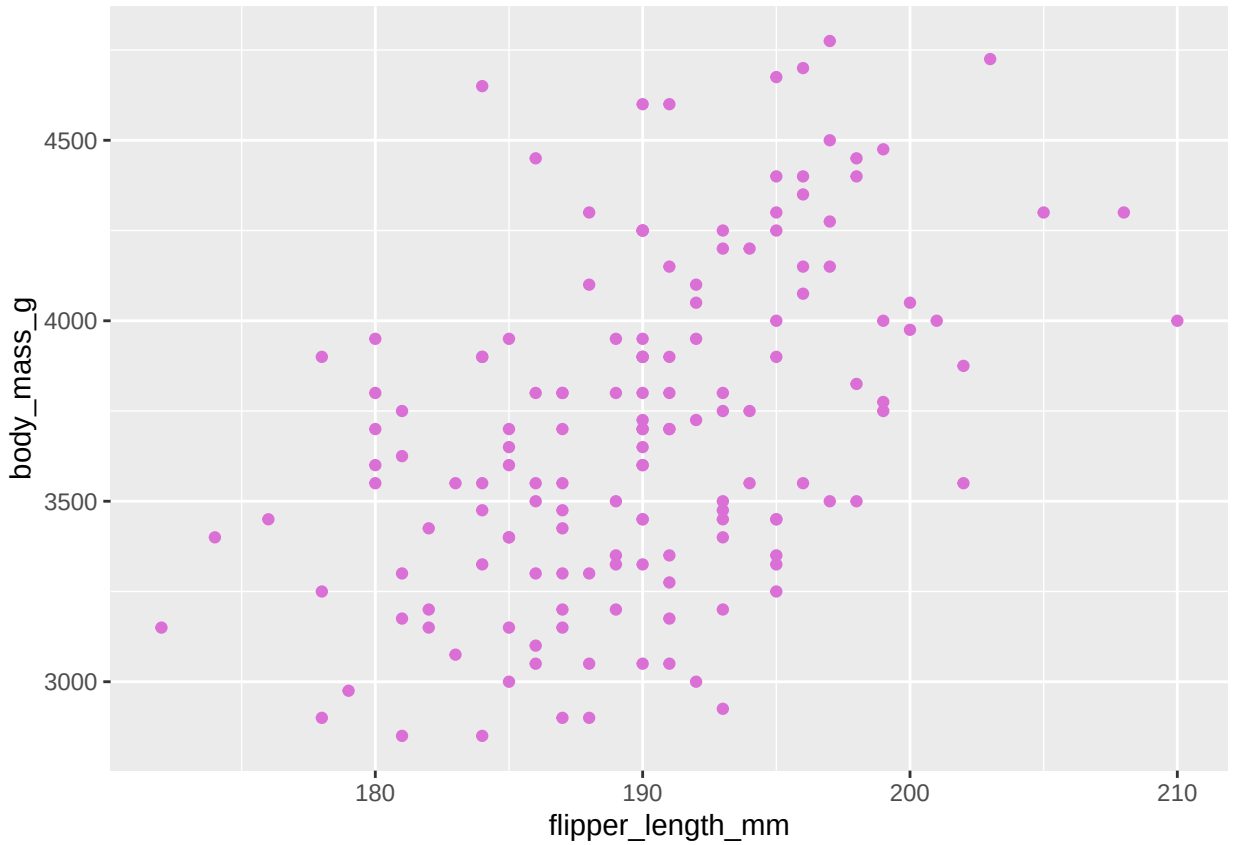
```r
# add points
penguins %>%
  filter(species == "Adelie") %>%
  ggplot() +
  geom_point(aes(x=flipper_length_mm, y = body_mass_g))
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```



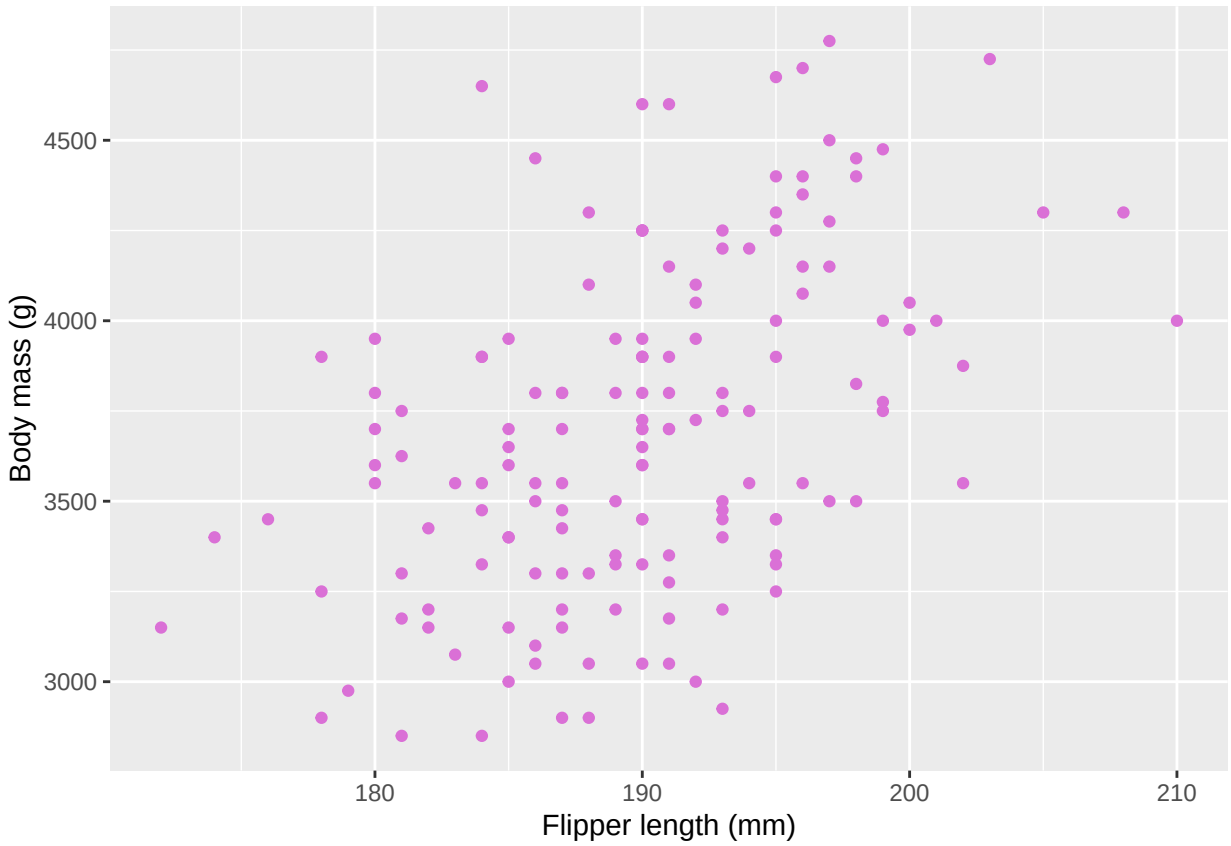

```r
# change color of point
penguins %>%
  filter(species == "Adelie") %>%
  ggplot() +
  geom_point(aes(x=flipper_length_mm, y = body_mass_g), color = "orchid")
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```



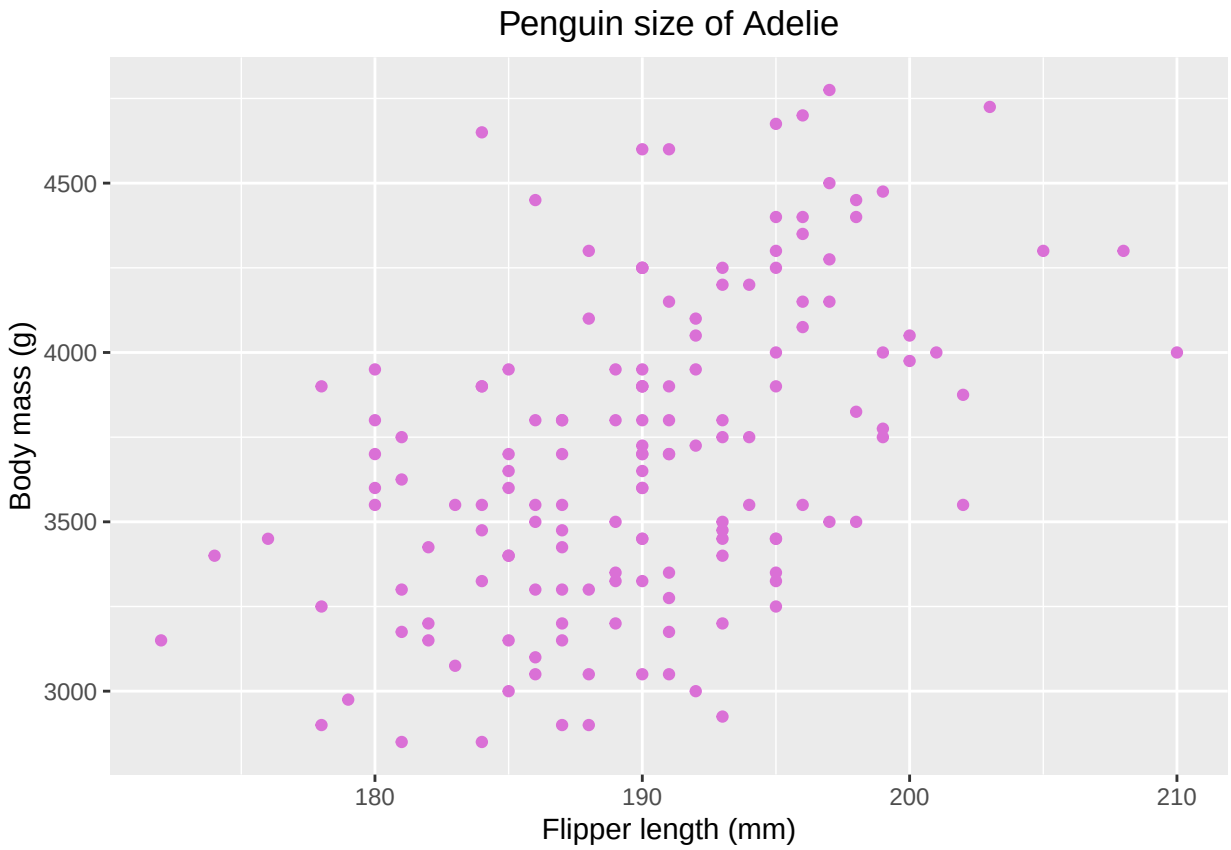

```
# change x and y labels
penguins %>%
  filter(species == "Adelie") %>%
  ggplot() +
  geom_point(aes(x = flipper_length_mm, y = body_mass_g), color = "orchid") +
  xlab("Flipper length (mm)") +
  ylab("Body mass (g)")
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```



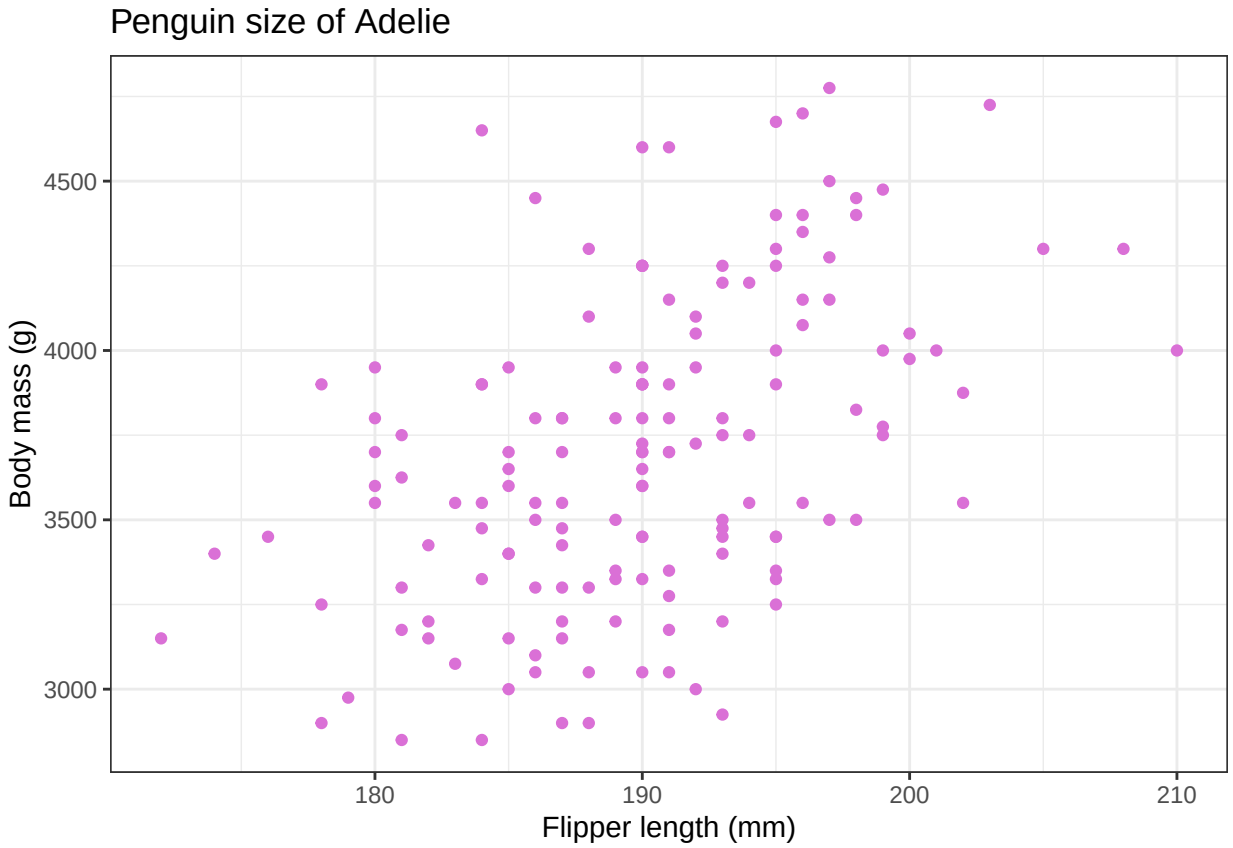

```
# change x and y labels
penguins %>%
  filter(species == "Adelie") %>%
  ggplot() +
  geom_point(aes(x = flipper_length_mm, y = body_mass_g), color = "orchid") +
  xlab("Flipper length (mm)") +
  ylab("Body mass (g)") +
  ggtitle("Penguin size of Adelie") +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```



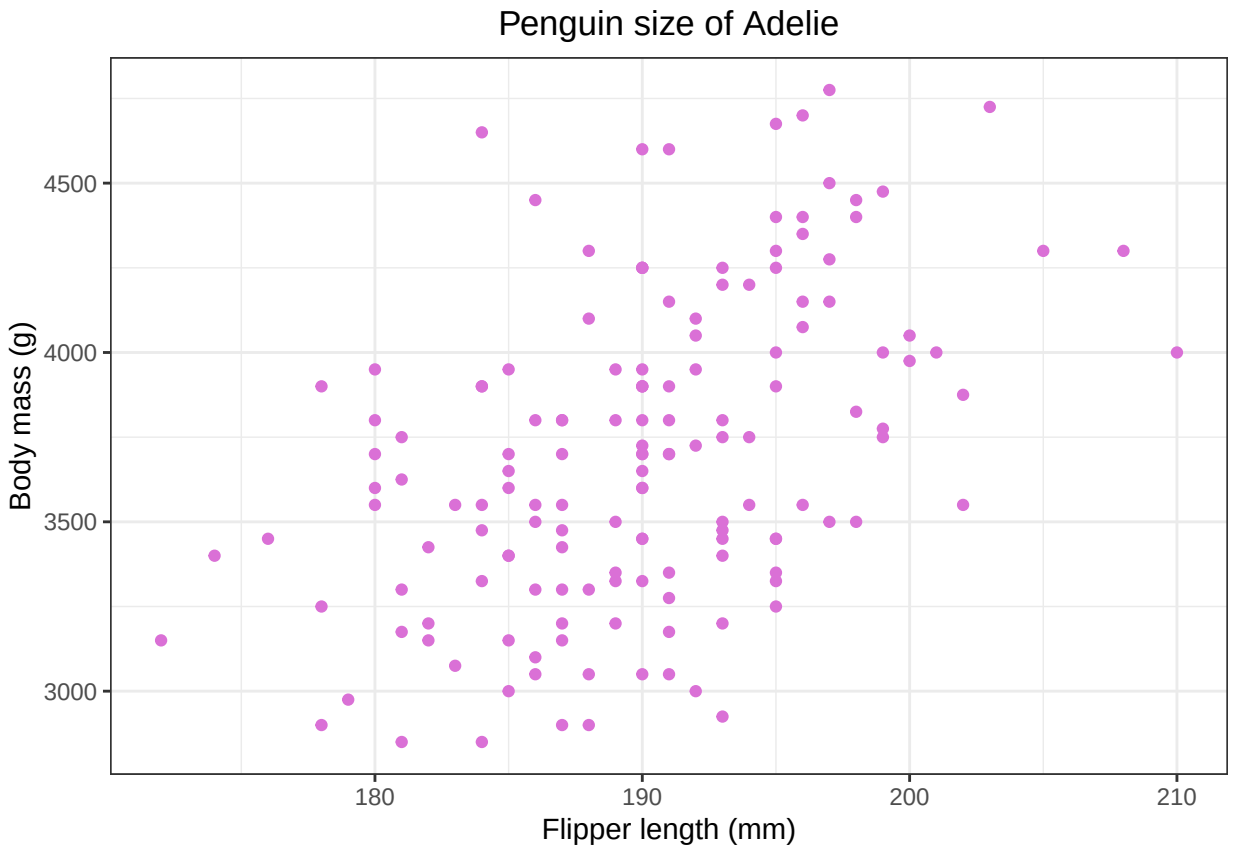

```
# change the background of the plot
penguins %>%
  filter(species == "Adelie") %>%
  ggplot() +
  geom_point(aes(x = flipper_length_mm, y = body_mass_g), color = "orchid") +
  xlab("Flipper length (mm)") +
  ylab("Body mass (g)") +
  ggtitle("Penguin size of Adelie") +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme_bw()
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```




```
# change the background of the plot
penguins %>%
  filter(species == "Adelie") %>%
  ggplot() +
  geom_point(aes(x = flipper_length_mm, y = body_mass_g), color = "orchid") +
  xlab("Flipper length (mm)") +
  ylab("Body mass (g)") +
  ggtitle("Penguin size of Adelie") +
  theme_bw() + # The order of theme_bw() and theme() matters
  theme(plot.title = element_text(hjust = 0.5))
```

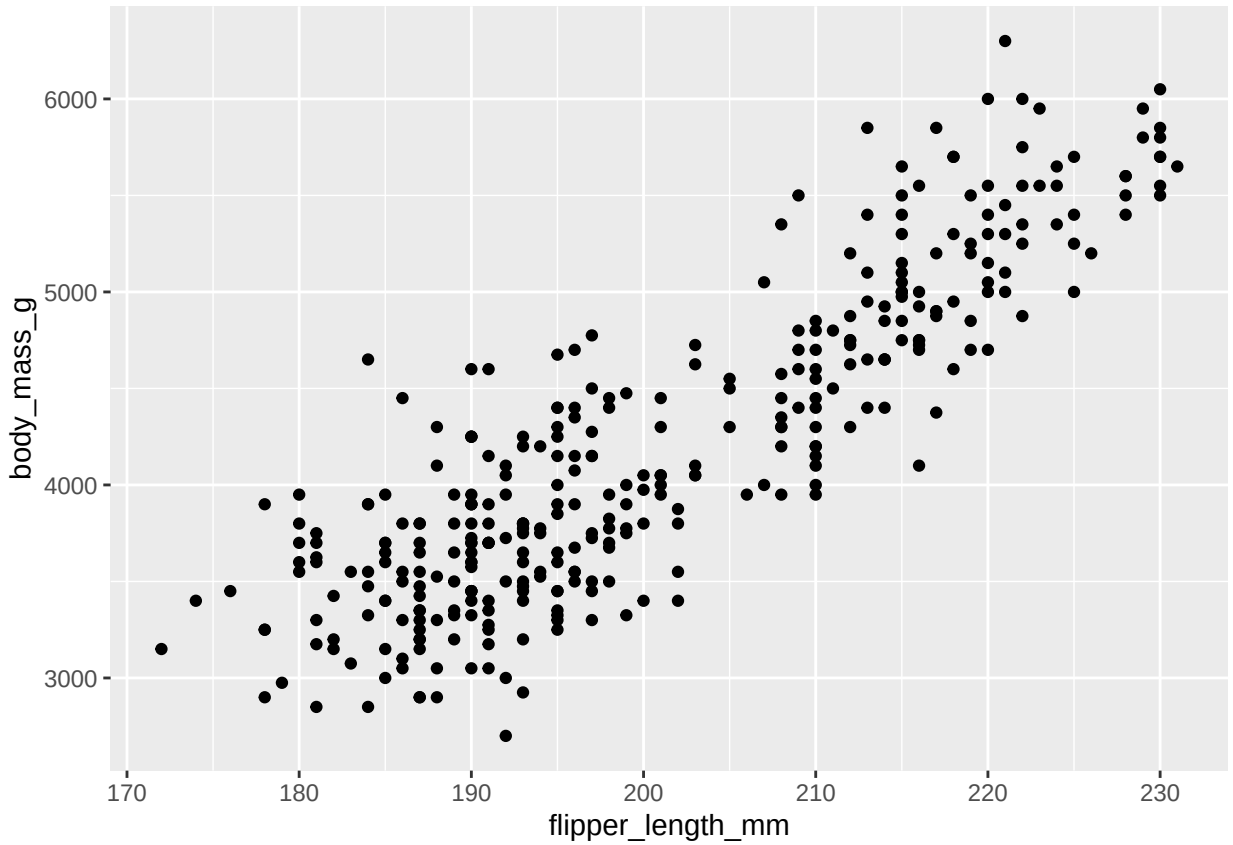
```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```



Task 2: A scatter plot of flipper length and body mass for ALL species

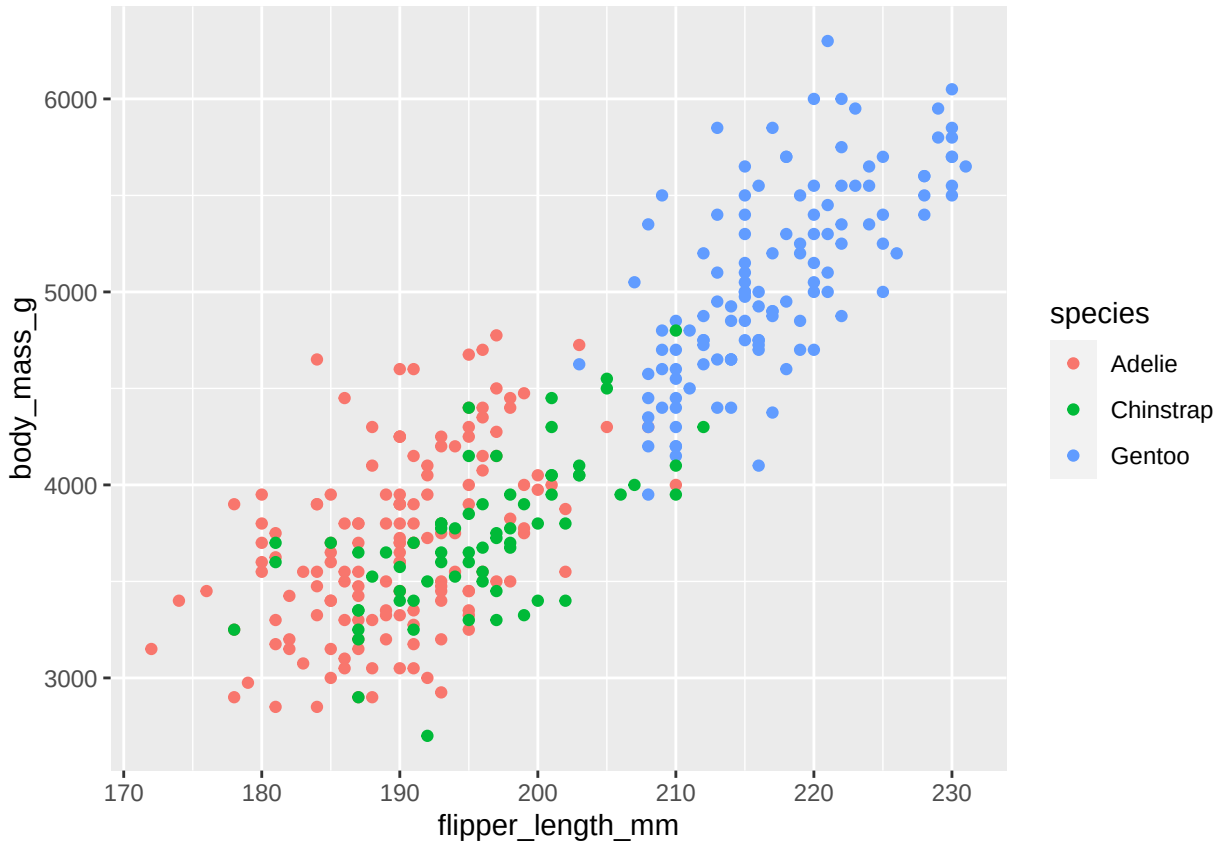
```
# Basic
# No need filter
penguins %>%
  ggplot() +
  geom_point(aes(x = flipper_length_mm, y = body_mass_g))
```

```
## Warning: Removed 2 rows containing missing values (`geom_point()`).
```



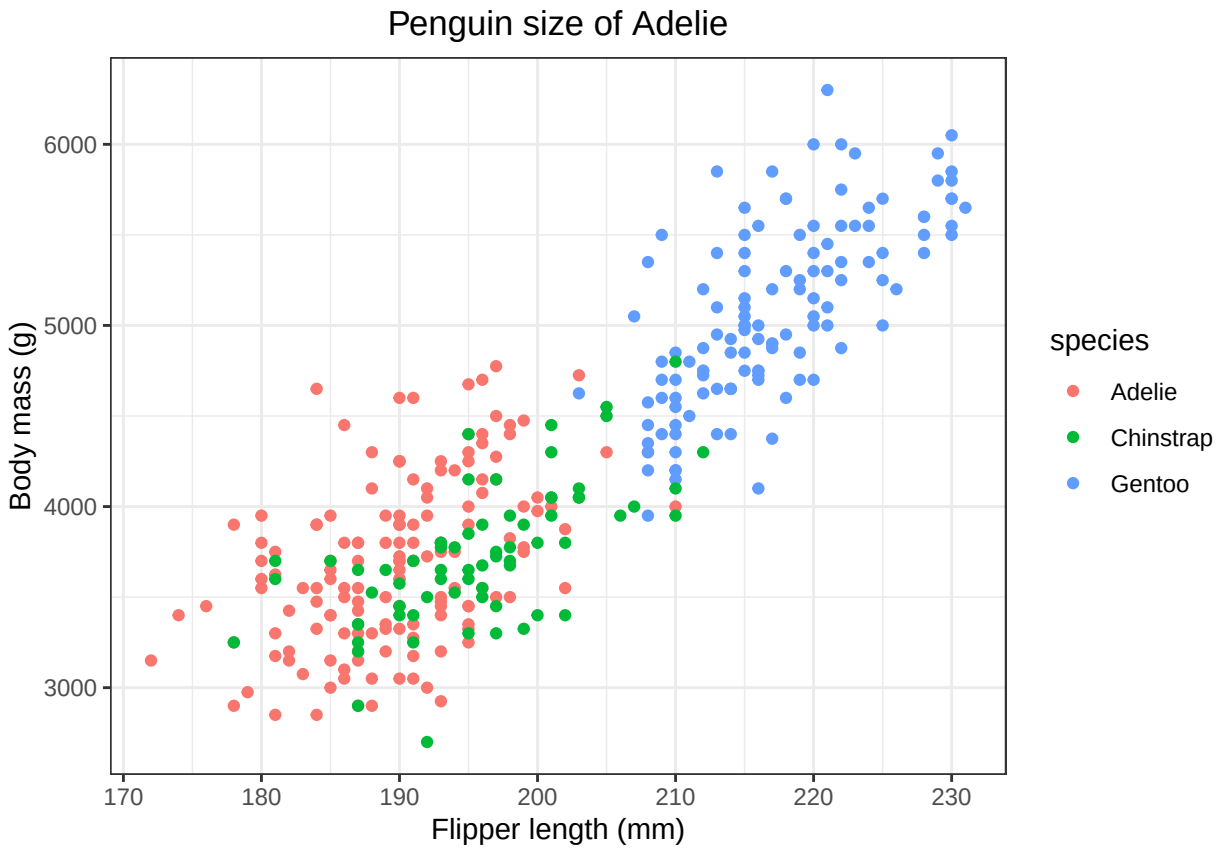

```
# This is bad as I dont know what specieis a point is from
penguins %>%
  ggplot() +
  geom_point(aes(x = flipper_length_mm, y = body_mass_g, color = species))
```

```
## Warning: Removed 2 rows containing missing values (`geom_point()`).
```



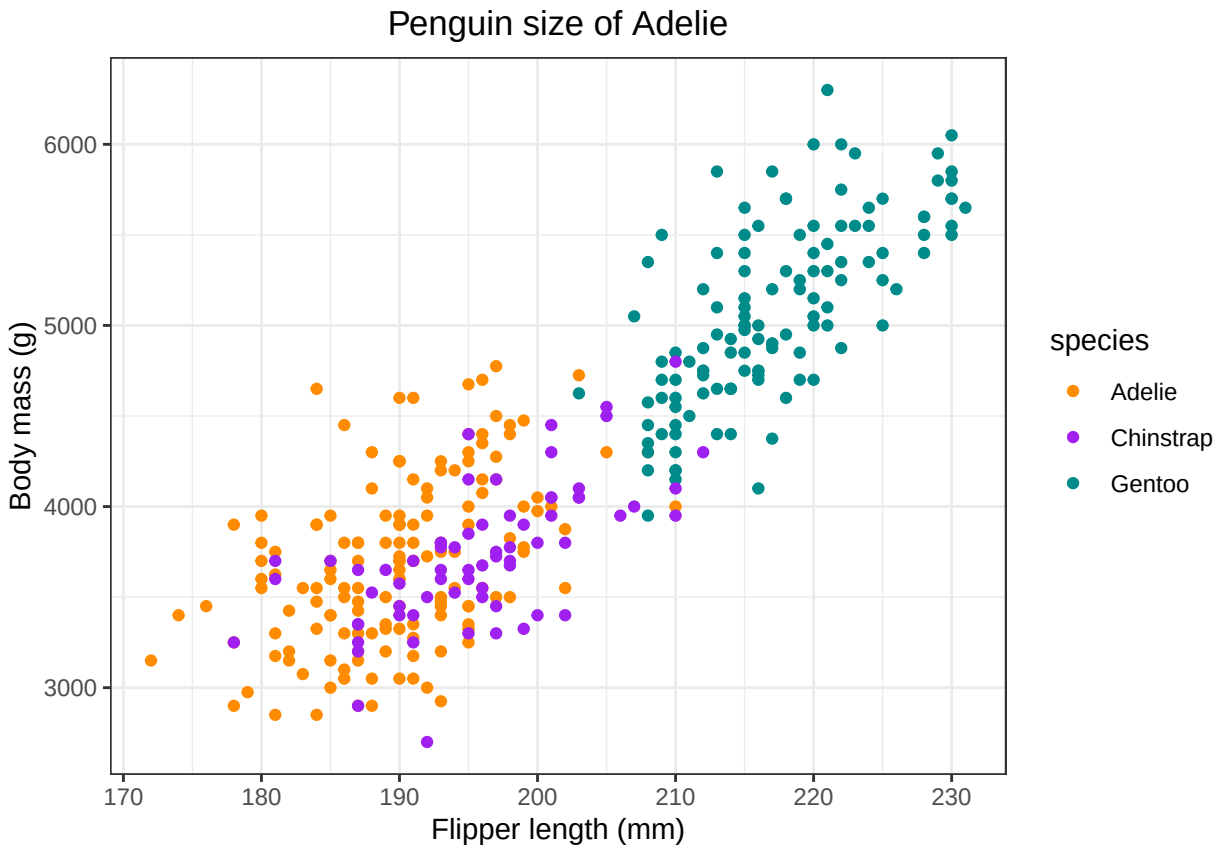

```
# You should take a moment to see now the color is inside the aes()
# Change x, y axis labels as before
penguins %>%
  ggplot() +
  geom_point(aes(x = flipper_length_mm, y = body_mass_g, color = species))  +
  xlab("Flipper length (mm)") +
  ylab("Body mass (g)") +
  ggtitle("Penguin size of Adelie") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 2 rows containing missing values (`geom_point()`).
```



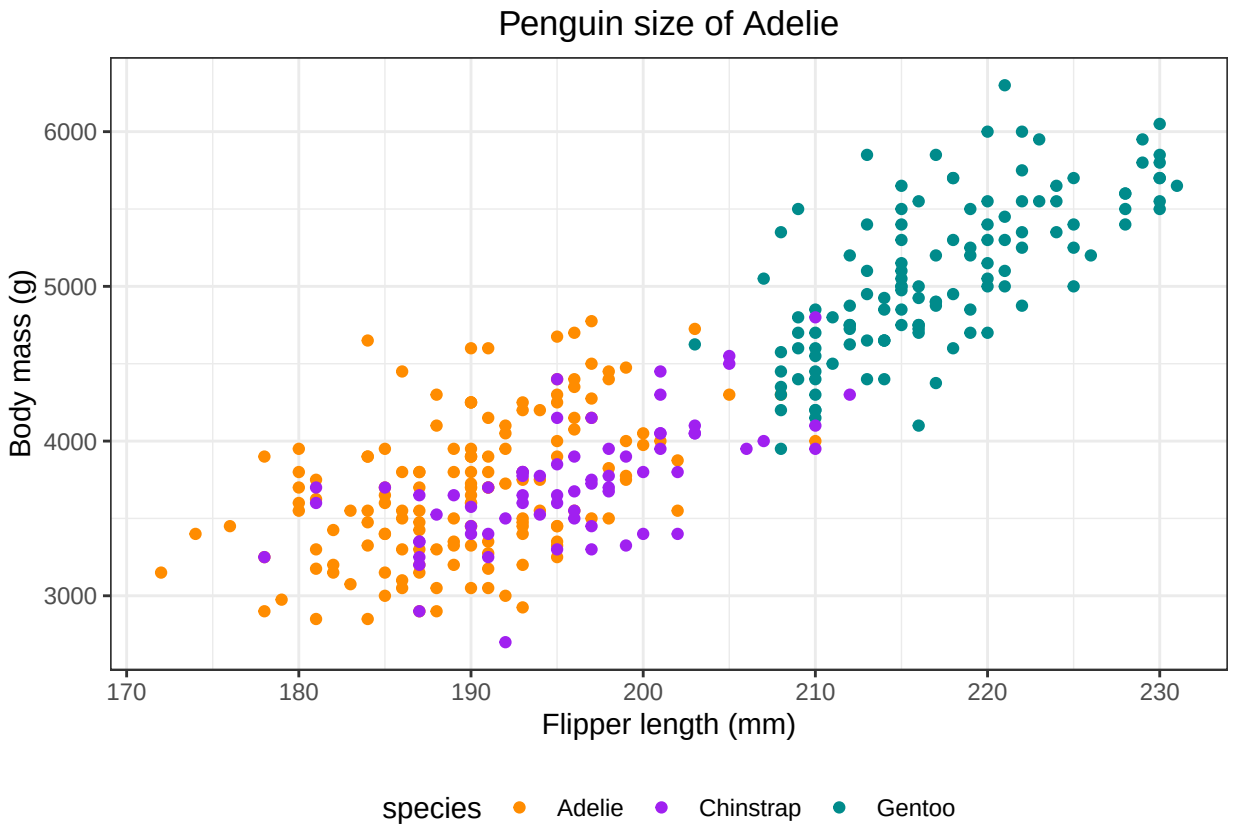

```
# Change the color manually for each group
penguins %>%
  ggplot() +
  geom_point(aes(x = flipper_length_mm, y = body_mass_g, color = species))  +
  xlab("Flipper length (mm)") +
  ylab("Body mass (g)") +
  ggtitle("Penguin size of Adelie") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_color_manual(values = c("darkorange","purple","cyan4"))
```

```
## Warning: Removed 2 rows containing missing values (`geom_point()`).
```



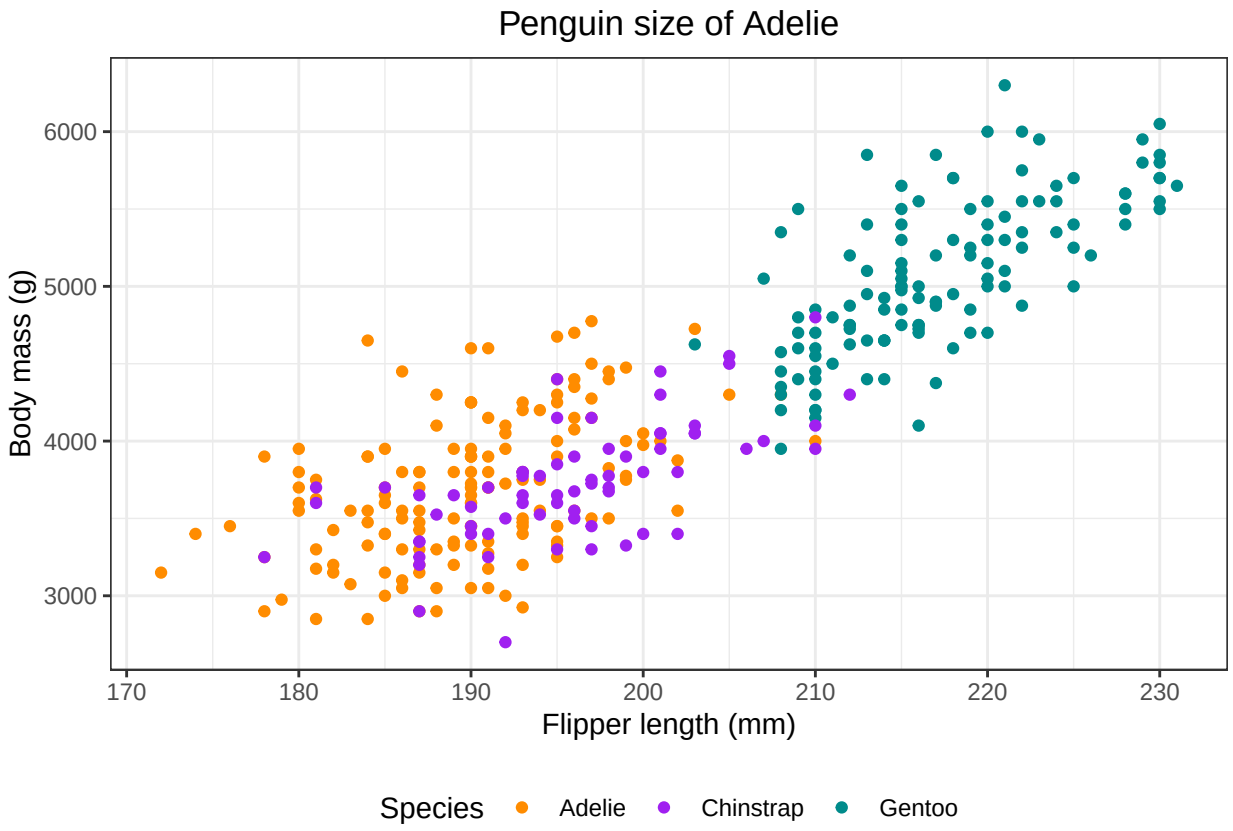

```
# Change legend position
penguins %>%
  ggplot() +
  geom_point(aes(x = flipper_length_mm, y = body_mass_g, color = species))  +
  xlab("Flipper length (mm)") +
  ylab("Body mass (g)") +
  ggtitle("Penguin size of Adelie") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_color_manual(values = c("darkorange","purple","cyan4")) +
  theme(legend.position = "bottom")
```

```
## Warning: Removed 2 rows containing missing values (`geom_point()`).
```




```
# Change legend title
penguins %>%
  ggplot() +
  geom_point(aes(x = flipper_length_mm, y = body_mass_g, color = species))  +
  xlab("Flipper length (mm)") +
  ylab("Body mass (g)") +
  ggtitle("Penguin size of Adelie") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5))+
  scale_color_manual(values = c("darkorange","purple","cyan4")) +
  theme(legend.position = "bottom") +
  labs(color = "Species")
```

```
## Warning: Removed 2 rows containing missing values (`geom_point()`).
```



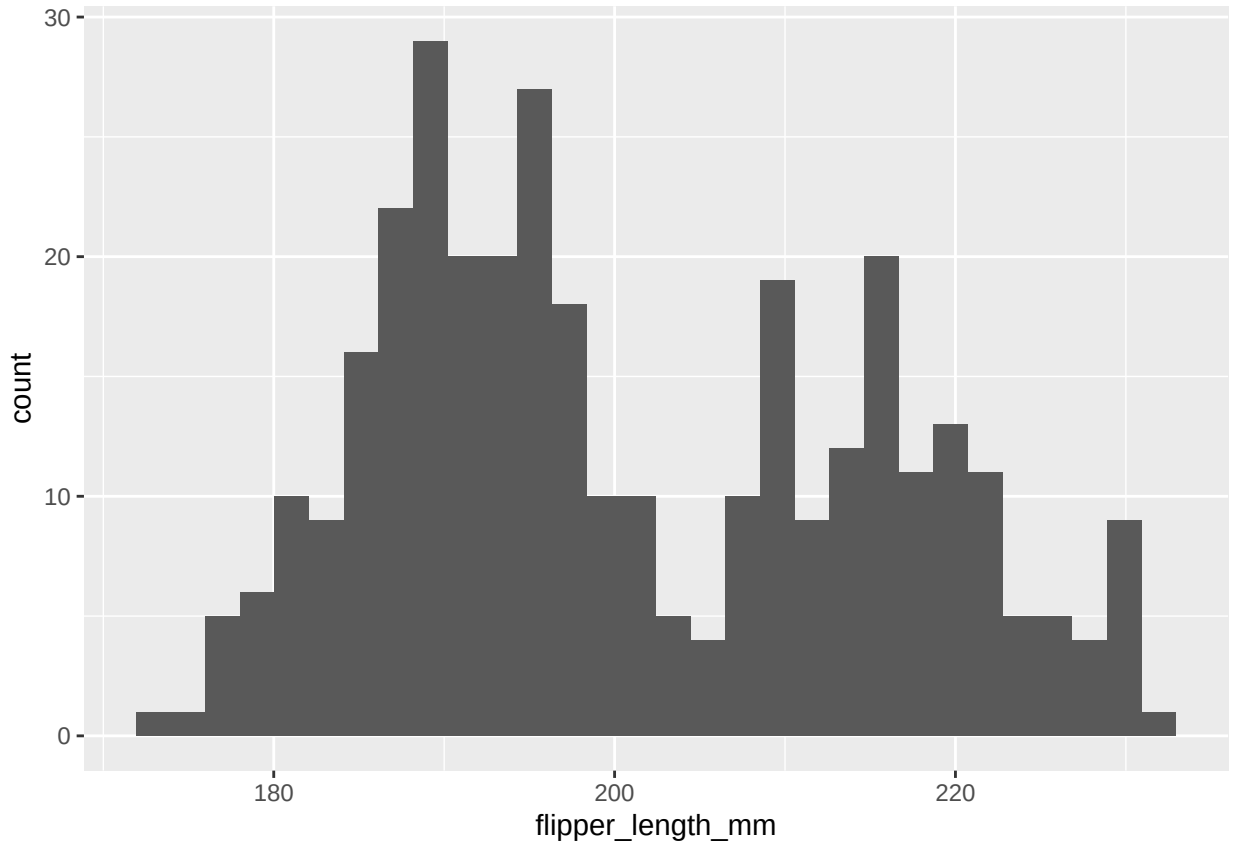
Histograms

Task 1: Plot a histgoram of flipper length

```
# Baisc
penguins %>%
   ggplot() +
   geom_histogram(aes(x = flipper_length_mm))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

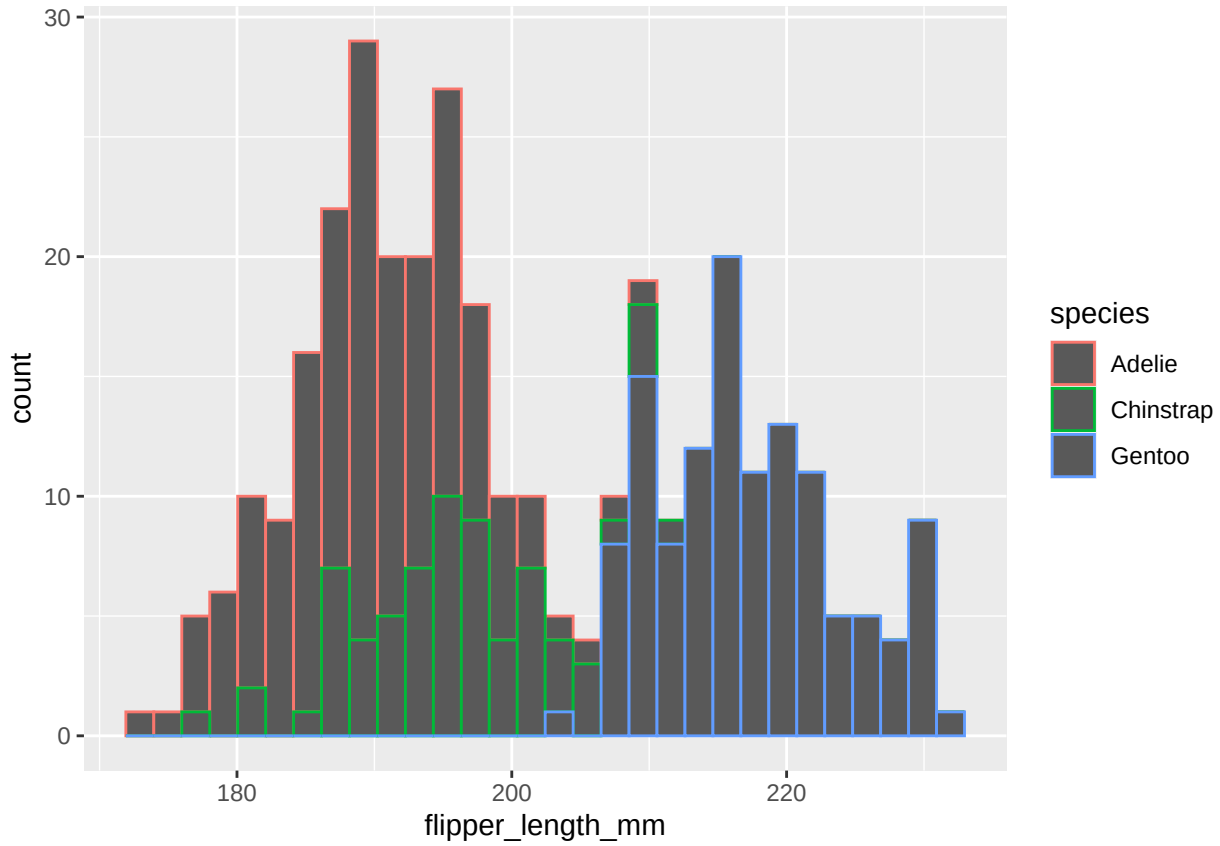
```
## Warning: Removed 2 rows containing non-finite values (`stat_bin()`).
```




```
# Same problem as before
# Cant distinguish between species
penguins %>%
   ggplot() +
   geom_histogram(aes(x = flipper_length_mm, color = species))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

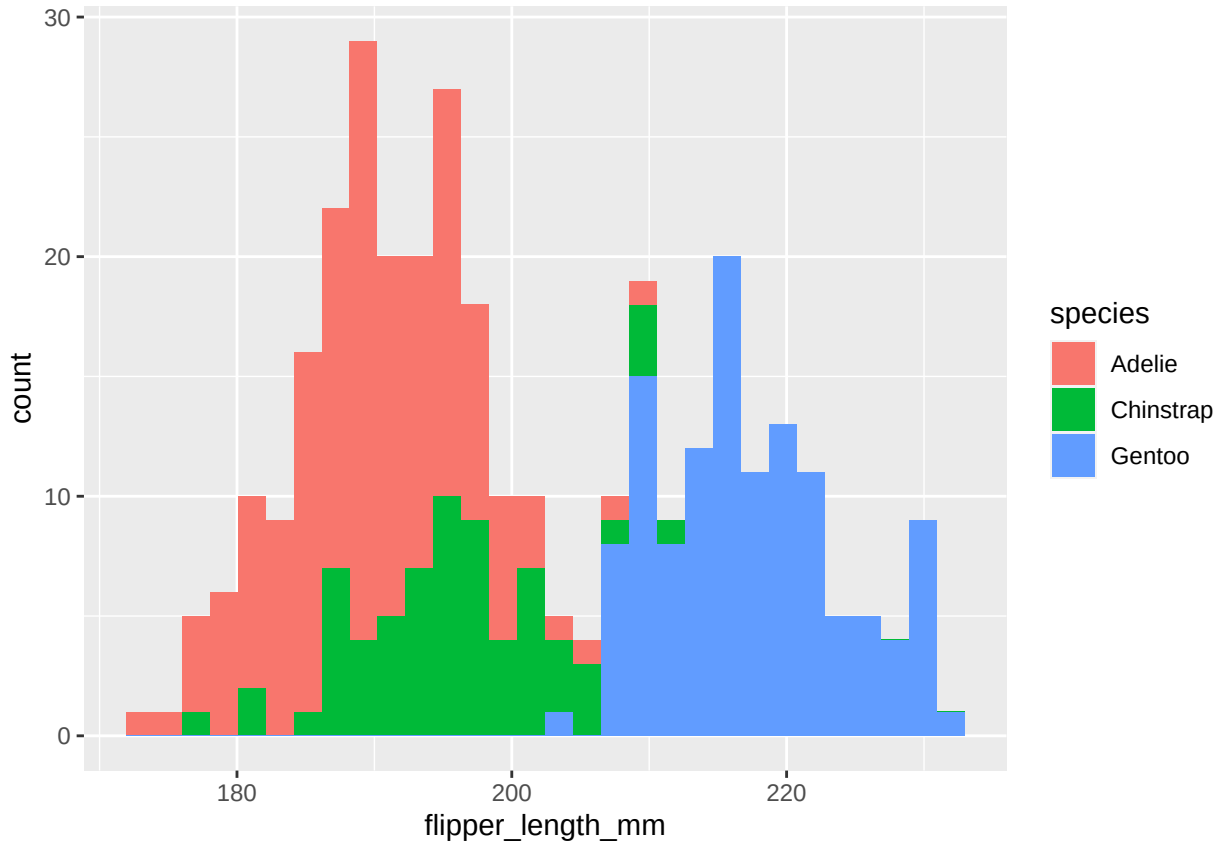
```
## Warning: Removed 2 rows containing non-finite values (`stat_bin()`).
```




```
# This looks not good
penguins %>%
   ggplot() +
   geom_histogram(aes(x = flipper_length_mm, fill = species))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

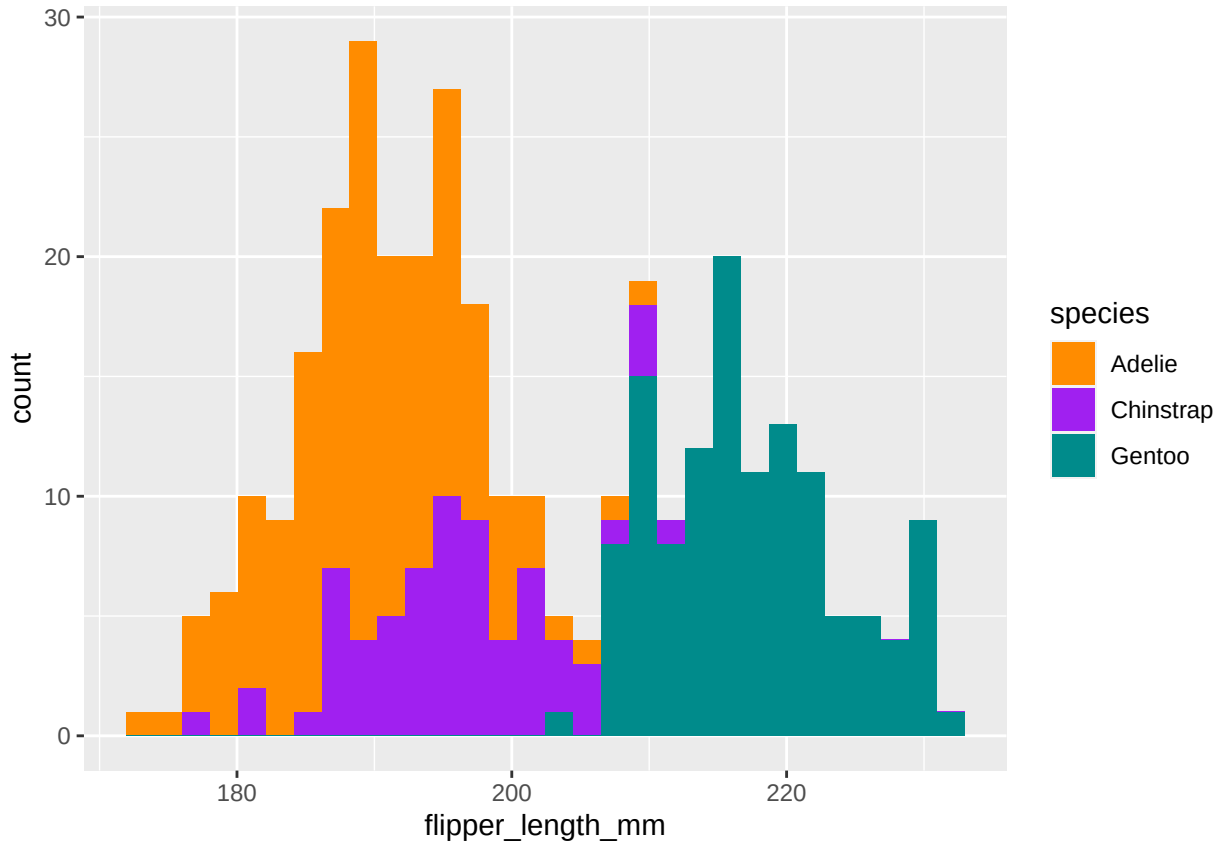
```
## Warning: Removed 2 rows containing non-finite values (`stat_bin()`).
```




```
# Manually change color
# Notice that now we are using fill
# so we should use scale_fill_manual() instead of scale_color_manual()
penguins %>%
   ggplot() +
   geom_histogram(aes(x = flipper_length_mm, fill = species)) +
   scale_fill_manual(values = c("darkorange","purple","cyan4"))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

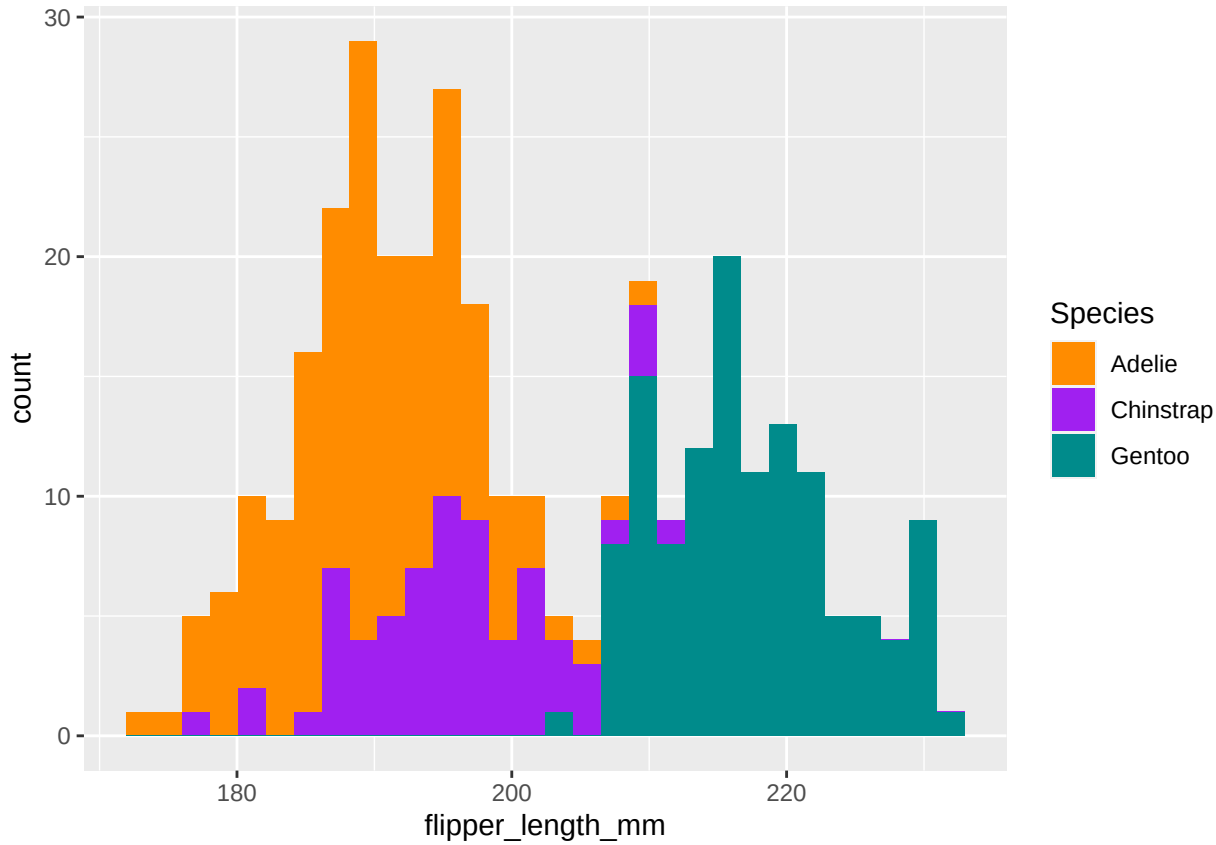
```
## Warning: Removed 2 rows containing non-finite values (`stat_bin()`).
```




```
# Similarly if we want to change the title of the legend
penguins %>%
   ggplot() +
   geom_histogram(aes(x = flipper_length_mm, fill = species)) +
   scale_fill_manual(values = c("darkorange","purple","cyan4")) +
   labs(fill = "Species")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

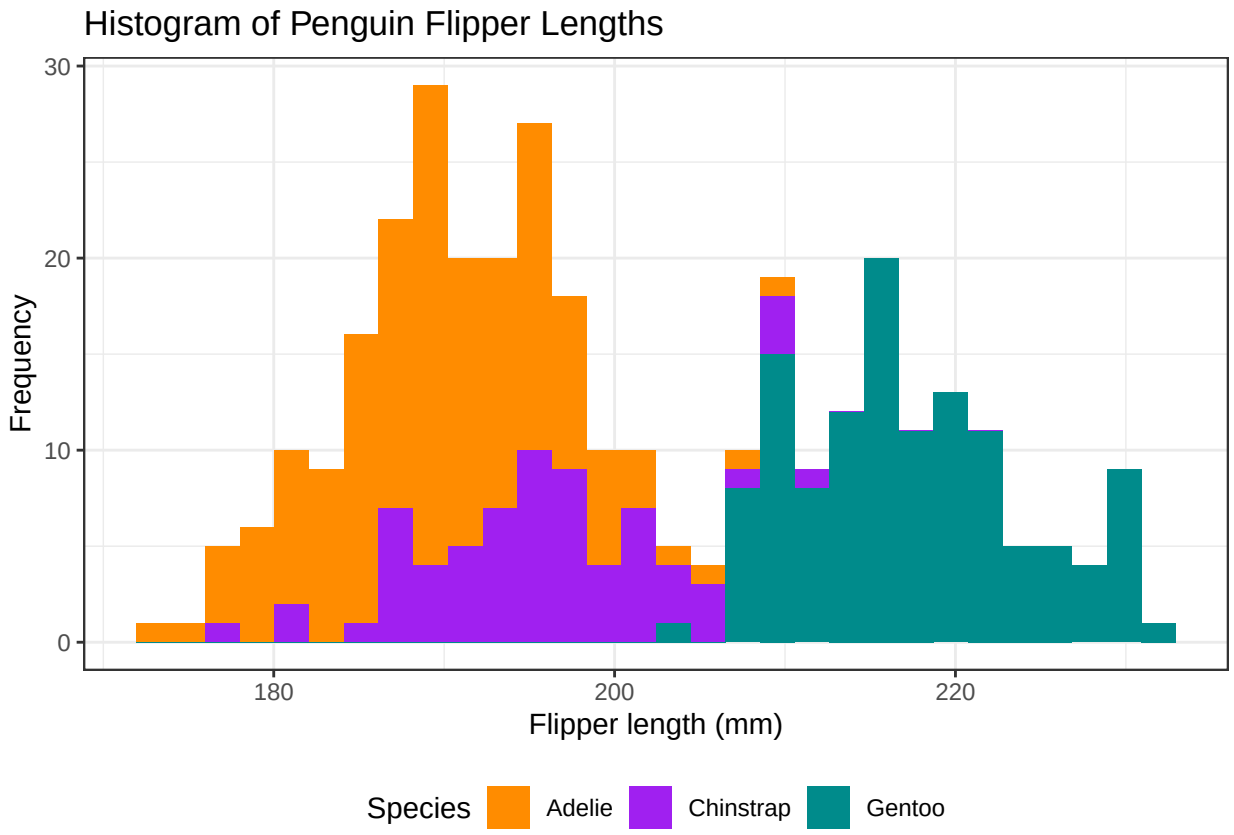
```
## Warning: Removed 2 rows containing non-finite values (`stat_bin()`).
```




```
# Changing theme, x, y axis labels and add titles are same as before
penguins %>%
   ggplot() +
   geom_histogram(aes(x = flipper_length_mm, fill = species)) +
   scale_fill_manual(values = c("darkorange","purple","cyan4")) +
   labs(fill = "Species") +
   theme_bw() +
   theme(legend.position = "bottom")  +
   xlab("Flipper length (mm)") +
   ylab("Frequency") +
   ggtitle("Histogram of Penguin Flipper Lengths")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

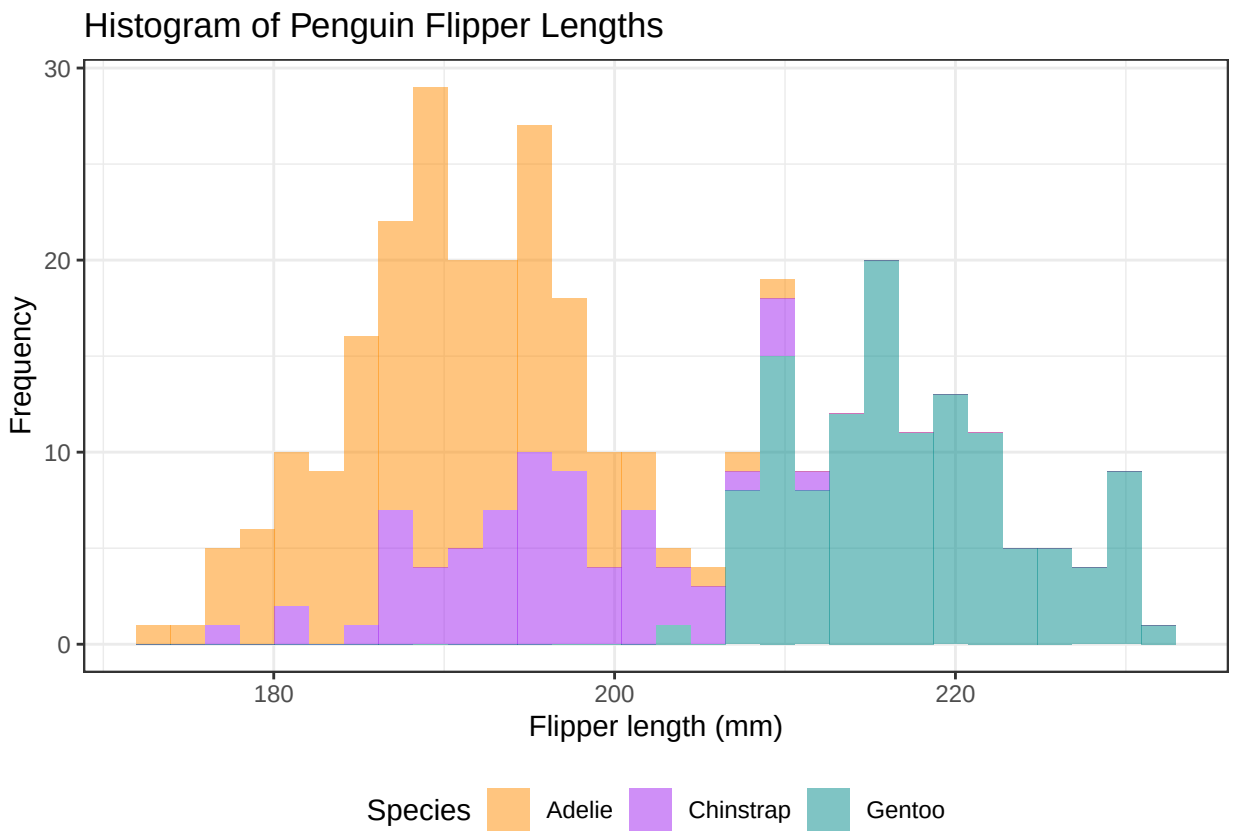
```
## Warning: Removed 2 rows containing non-finite values (`stat_bin()`).
```




```
# this is hard to see as the histograms are overlapping.
# one potential remedy is to use alpha, which changes the transparency of the color
penguins %>%
   ggplot() +
   geom_histogram(aes(x = flipper_length_mm, fill = species), alpha = 0.5) +
   scale_fill_manual(values = c("darkorange","purple","cyan4")) +
   labs(fill = "Species") +
   theme_bw() +
   theme(legend.position = "bottom")  +
   xlab("Flipper length (mm)") +
   ylab("Frequency") +
   ggtitle("Histogram of Penguin Flipper Lengths")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

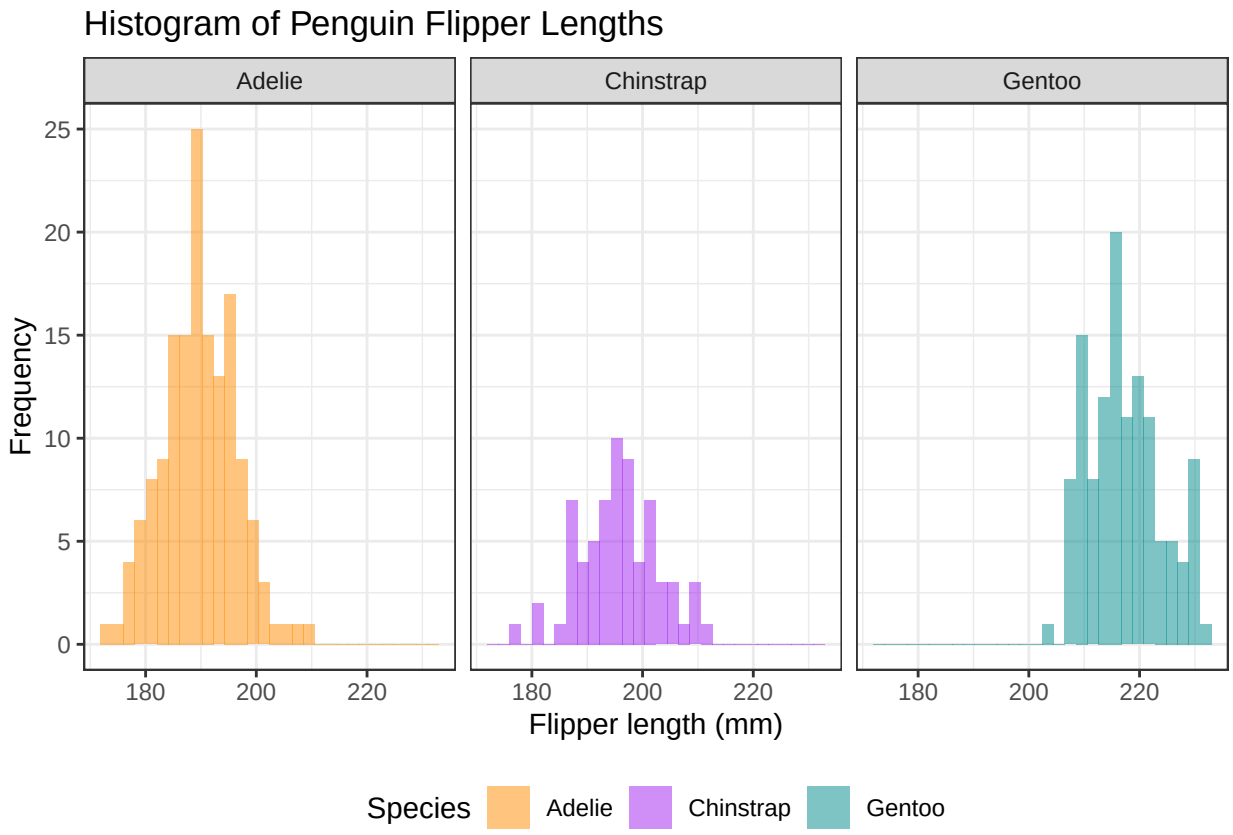
```
## Warning: Removed 2 rows containing non-finite values (`stat_bin()`).
```




```
# But this is still hard to see in this case
# Use Facet Plot
penguins %>%
   ggplot() +
   geom_histogram(aes(x = flipper_length_mm, fill = species), alpha = 0.5) +
   scale_fill_manual(values = c("darkorange","purple","cyan4")) +
   labs(fill = "Species") +
   theme_bw() +
   theme(legend.position = "bottom")  +
   xlab("Flipper length (mm)") +
   ylab("Frequency") +
   ggtitle("Histogram of Penguin Flipper Lengths") +
   facet_wrap(. ~ species)
```

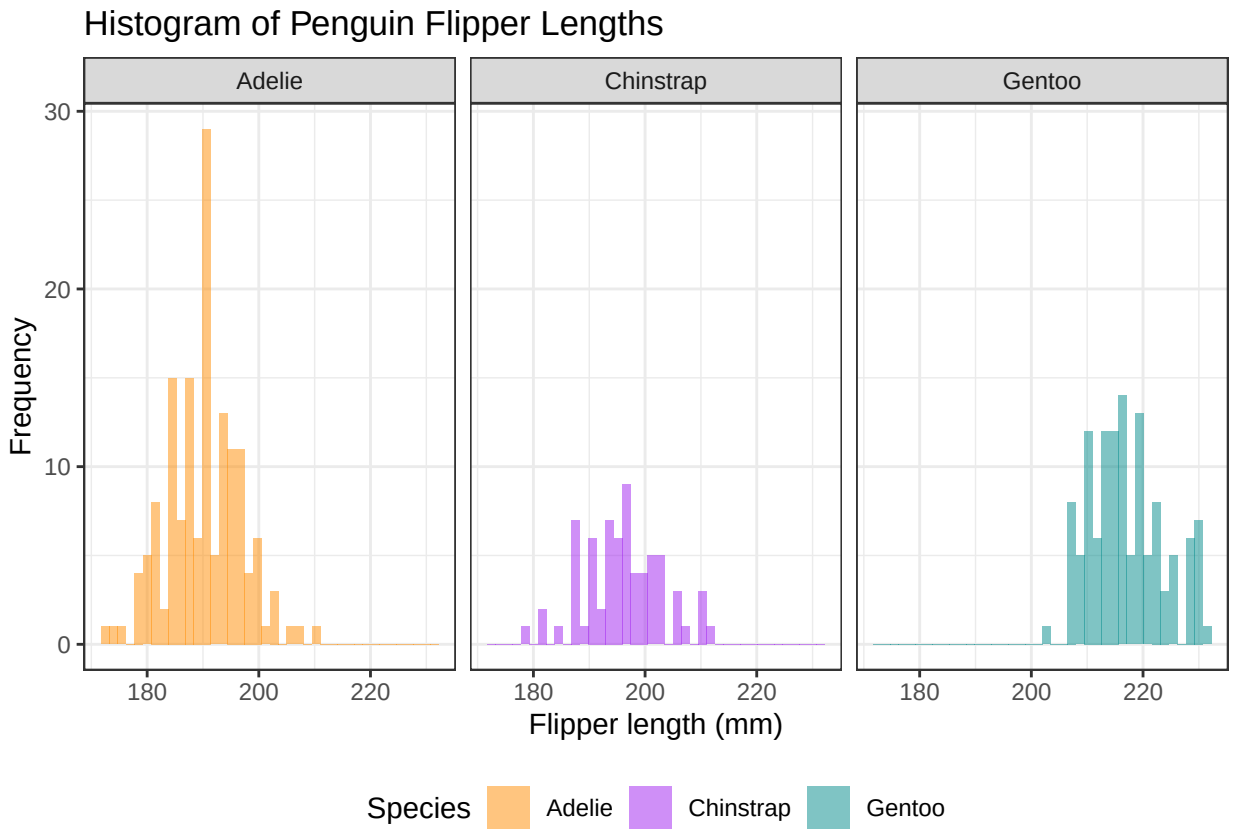
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 2 rows containing non-finite values (`stat_bin()`).
```




```
# change bins
penguins %>%
   ggplot() +
   geom_histogram(aes(x = flipper_length_mm, fill = species), alpha = 0.5, bins = 40) +
   scale_fill_manual(values = c("darkorange","purple","cyan4")) +
   labs(fill = "Species") +
   theme_bw() +
   theme(legend.position = "bottom")  +
   xlab("Flipper length (mm)") +
   ylab("Frequency") +
   ggtitle("Histogram of Penguin Flipper Lengths") +
   facet_wrap(. ~ species)
```

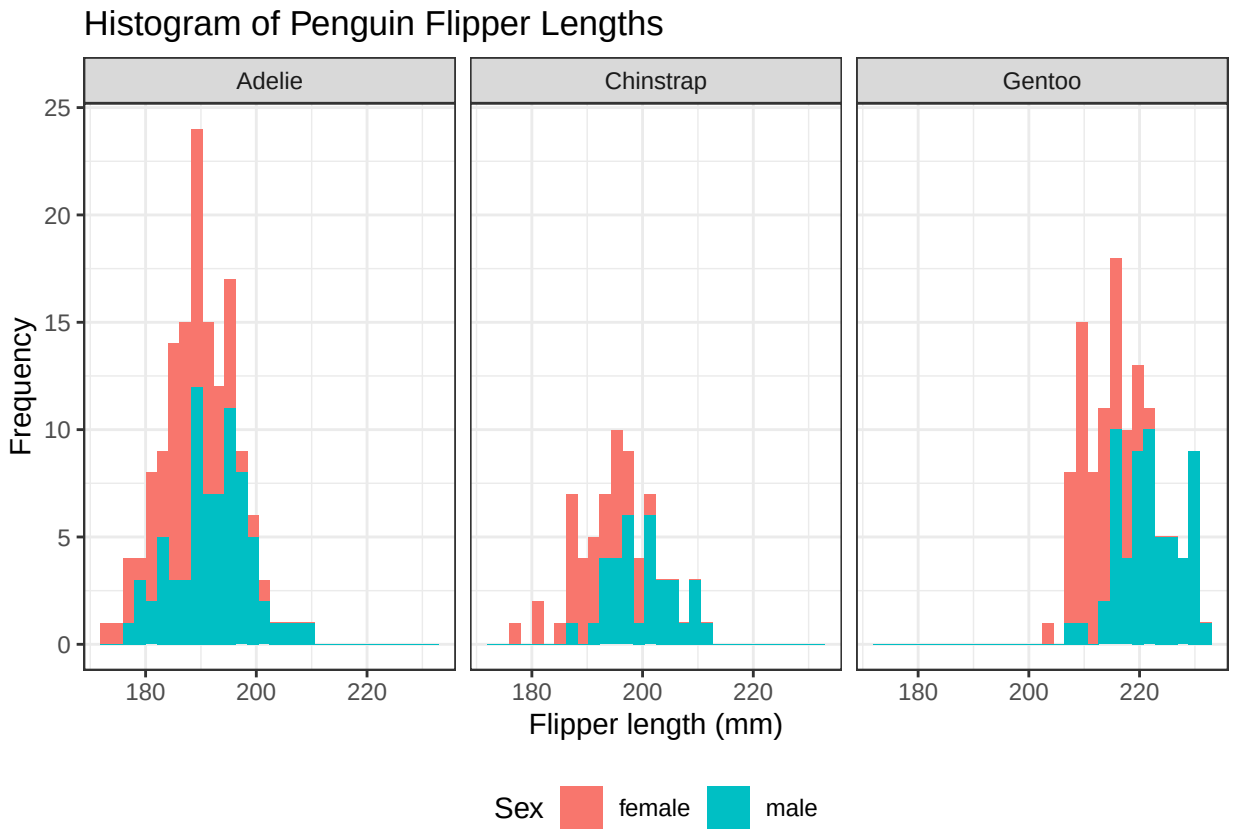
```
## Warning: Removed 2 rows containing non-finite values (`stat_bin()`).
```



Task 2 Stratify it by sex

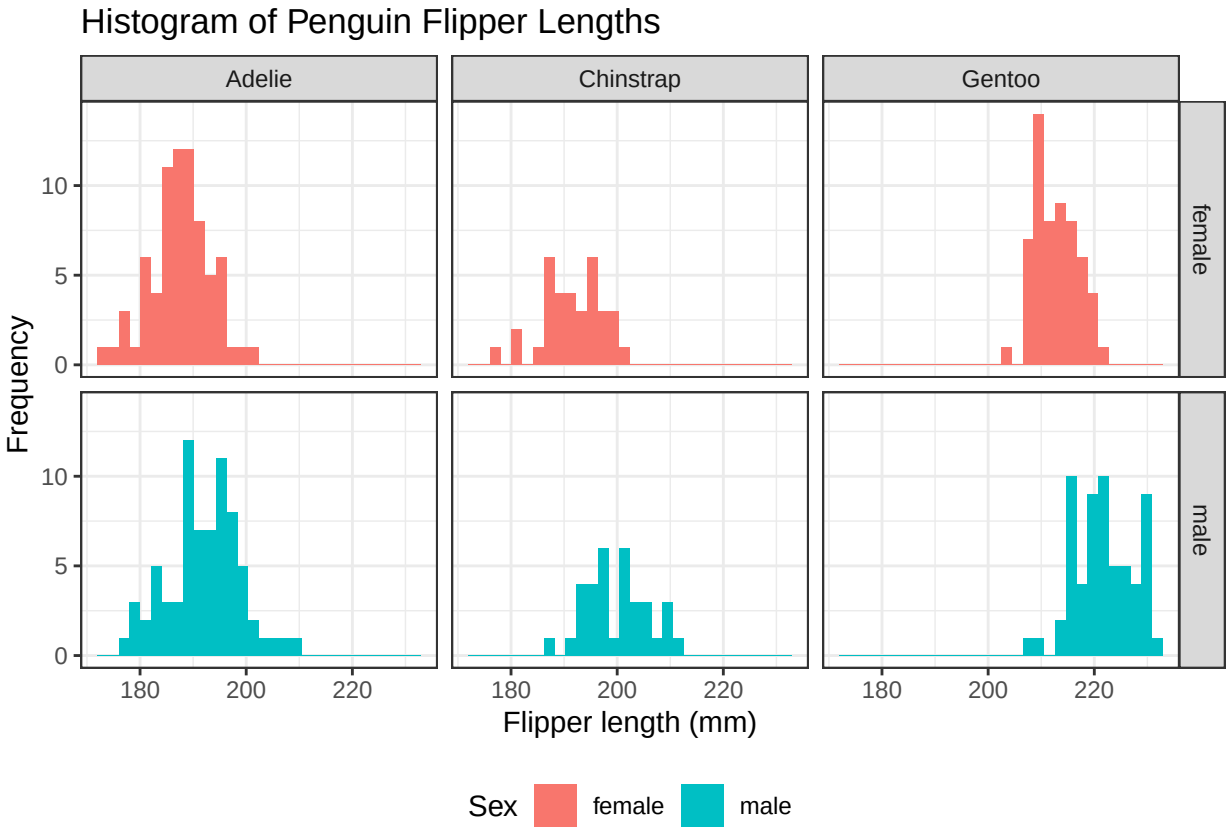
```
penguins %>%
  filter(!is.na(sex)) %>%
   ggplot() +
   geom_histogram(aes(x = flipper_length_mm, fill = sex)) +
   labs(fill = "Sex") +
   theme_bw() +
   theme(legend.position = "bottom")  +
   xlab("Flipper length (mm)") +
   ylab("Frequency") +
   ggtitle("Histogram of Penguin Flipper Lengths") +
   facet_wrap(. ~ species)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
penguins %>%
  filter(!is.na(sex)) %>%
   ggplot() +
   geom_histogram(aes(x = flipper_length_mm, fill = sex)) +
   labs(fill = "Sex") +
   theme_bw() +
   theme(legend.position = "bottom")  +
   xlab("Flipper length (mm)") +
   ylab("Frequency") +
   ggtitle("Histogram of Penguin Flipper Lengths") +
   facet_grid(sex ~ species)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


Histogram of Penguin Flipper Lengths
Adelie
Chinstrap
Gentoo
female
male
Frequency
10
5
0
180
200
220
Flipper length (mm)
Sex
female
male