

```
# Assignment: ASSIGNMENT 4
```

```
# Name: Sharma, Dipika
```

```
# Date: 2021-04-25
```

```
## Load the ggplot2 package
```

```
#install.packages("ggplot2")
```

```
> library(ggplot2)
```

```
> theme_set(theme_minimal())
```

```
>
```

```
> getwd()
```

```
[1] "/Users/dipikasharma/R_Projects/DSC520"
```

```
> ## Set the working directory to the root of your DSC 520 directory
```

```
> setwd("/Users/dipikasharma/R_Projects/DSC520")
```

```
## Load the `data/r4ds/heights.csv` to
```

```
heights_df <- read.csv("data/r4ds/heights.csv")
```

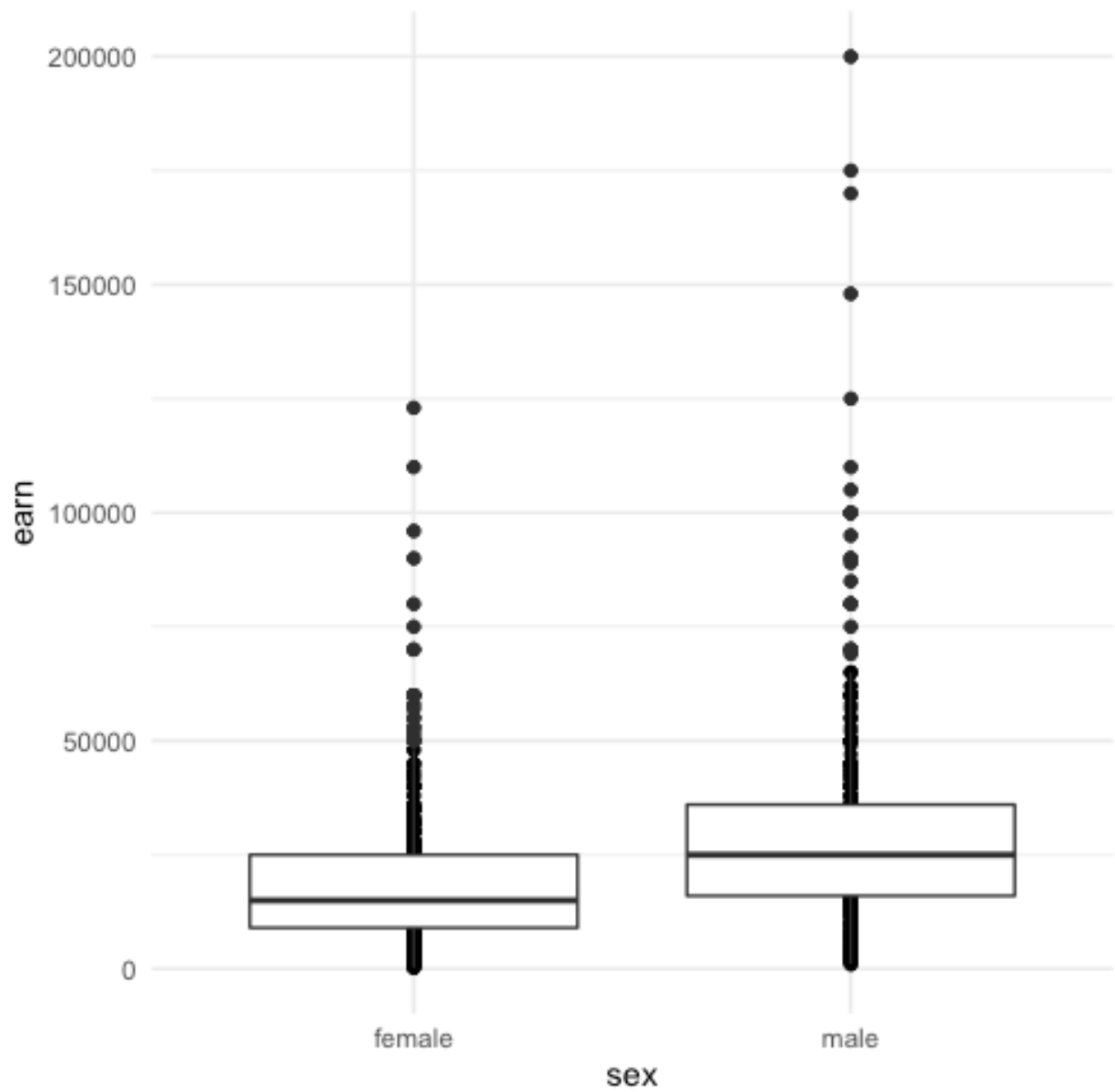
```
#heights_df
```

```
# https://ggplot2.tidyverse.org/reference/geom\_boxplot.html
```

```
## Create boxplots of sex vs. earn and race vs. earn using `geom_point()` and `geom_boxplot()`
```

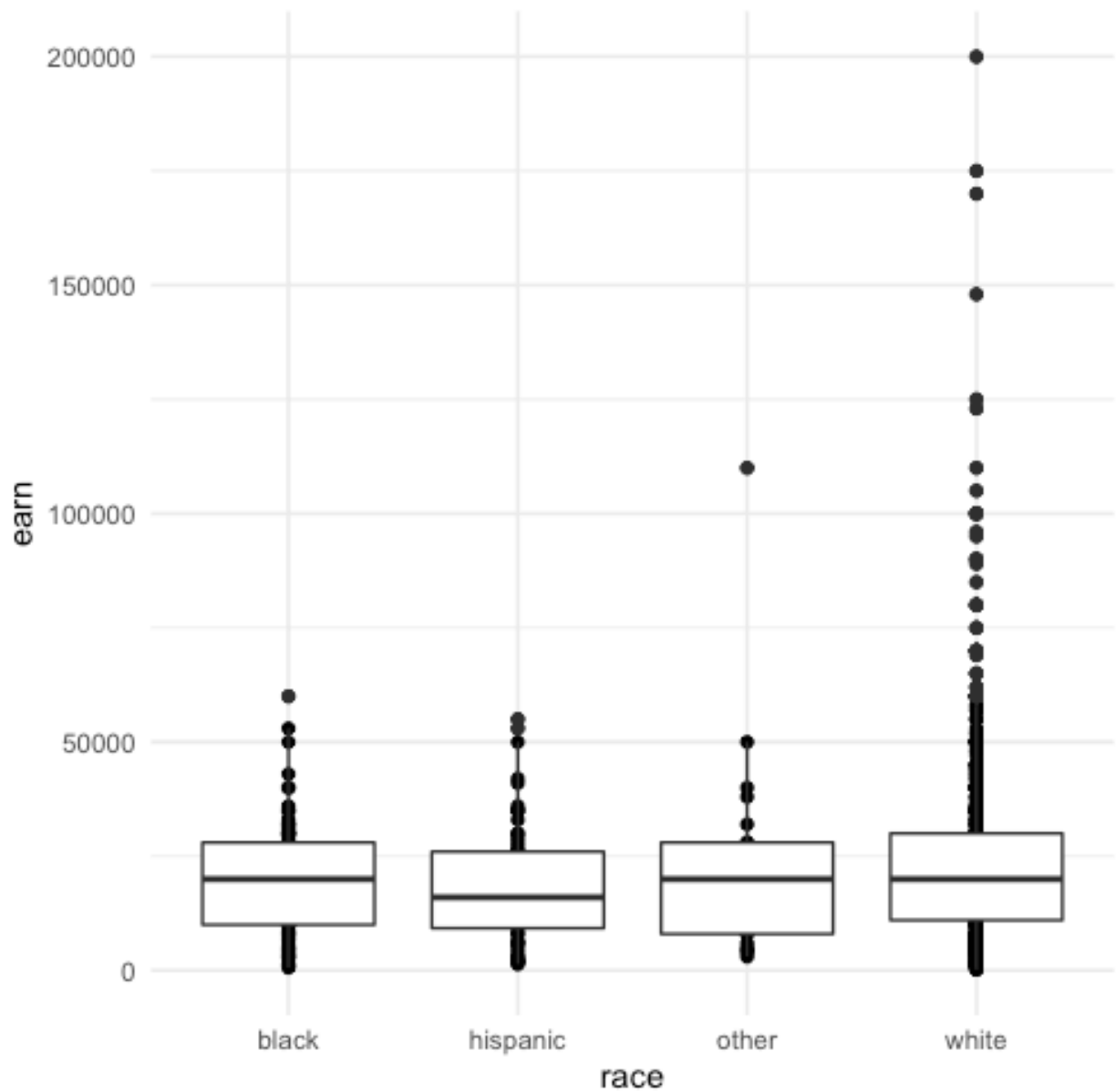
```
## sex vs. earn
```

```
ggplot(heights_df, aes(x=sex, y=earn)) + geom_point() + geom_boxplot()
```

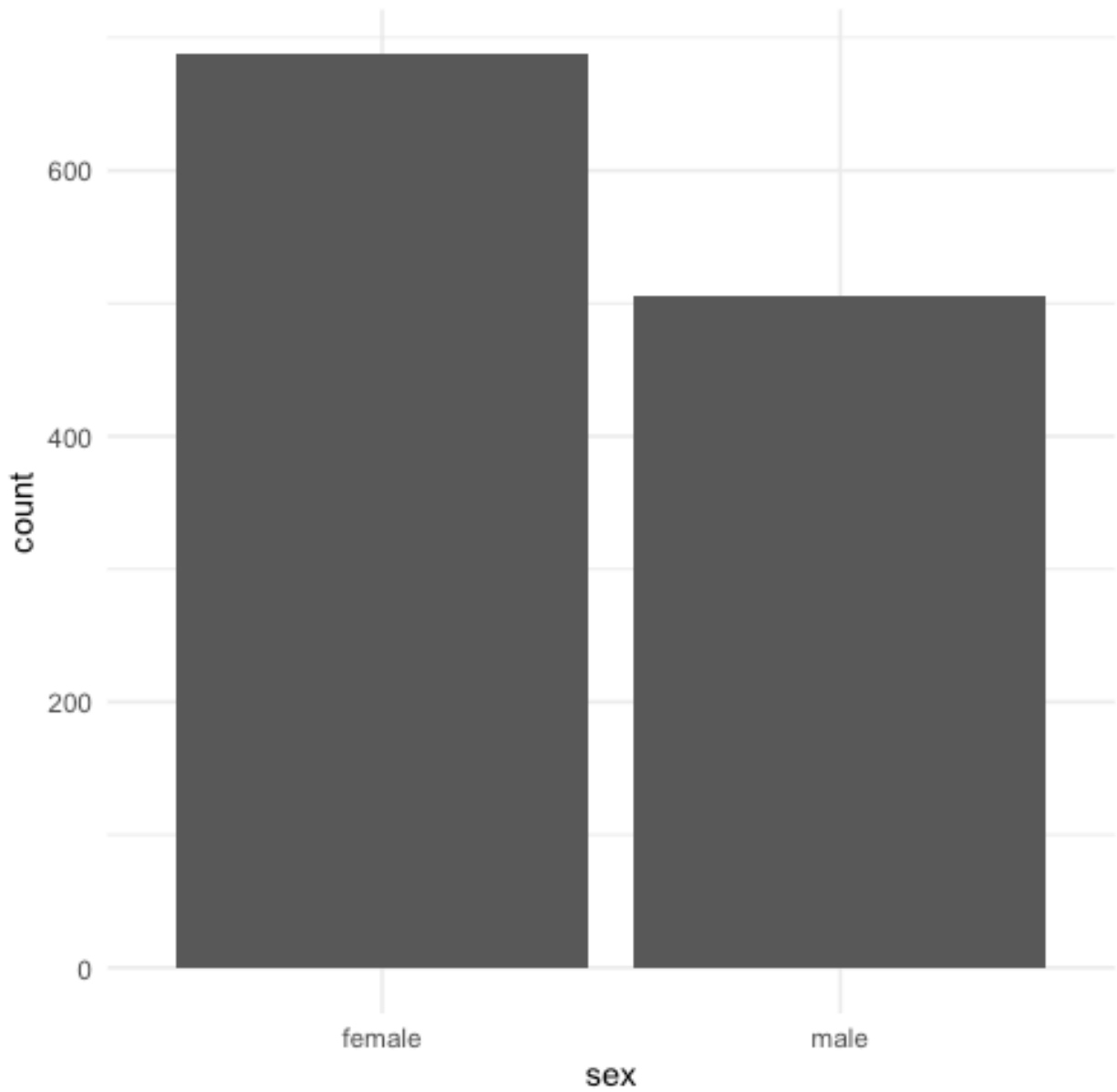


## race vs. earn

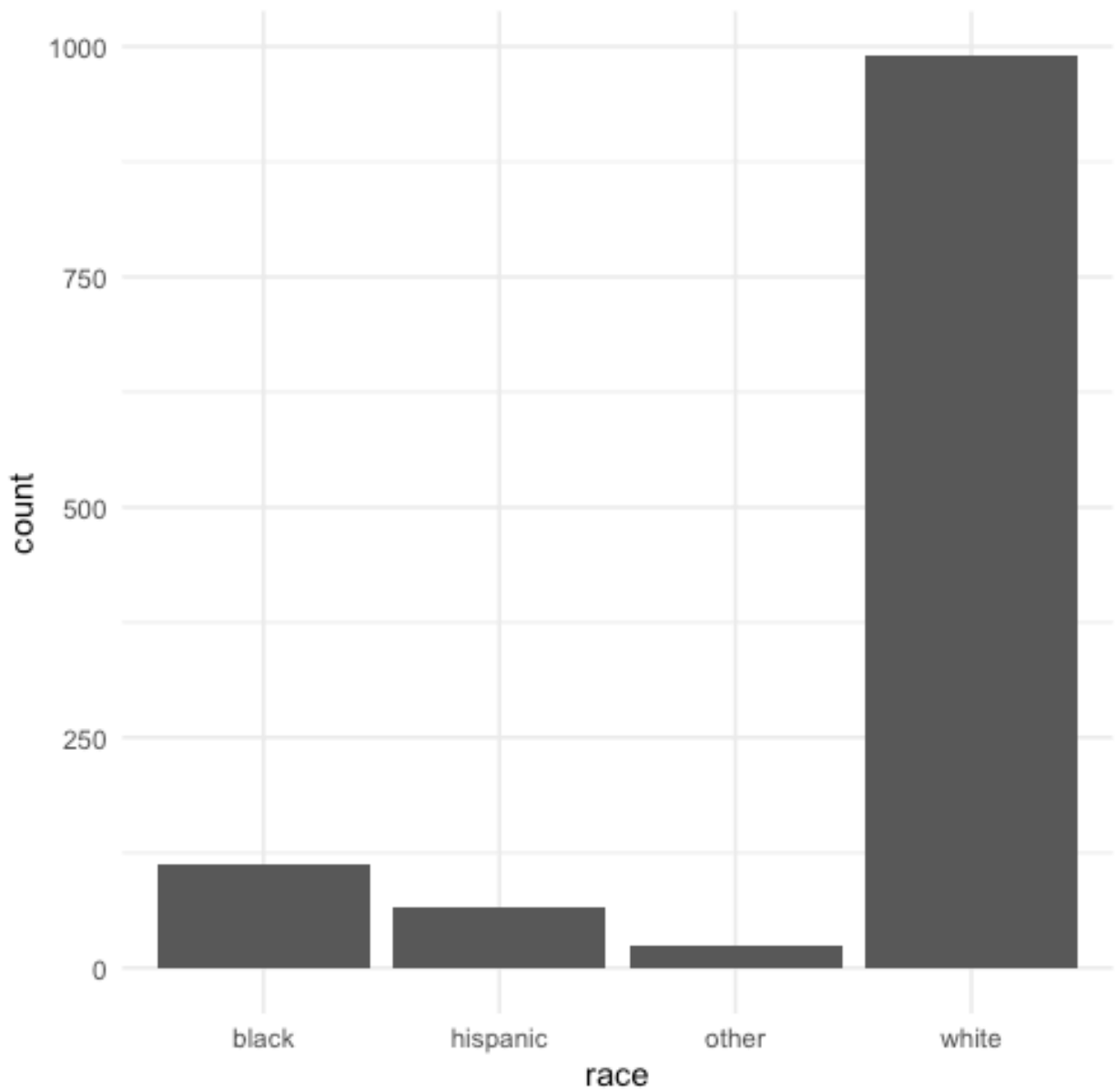
```
ggplot(heights_df, aes(x=race, y=earn)) + geom_point()+ geom_boxplot()
```



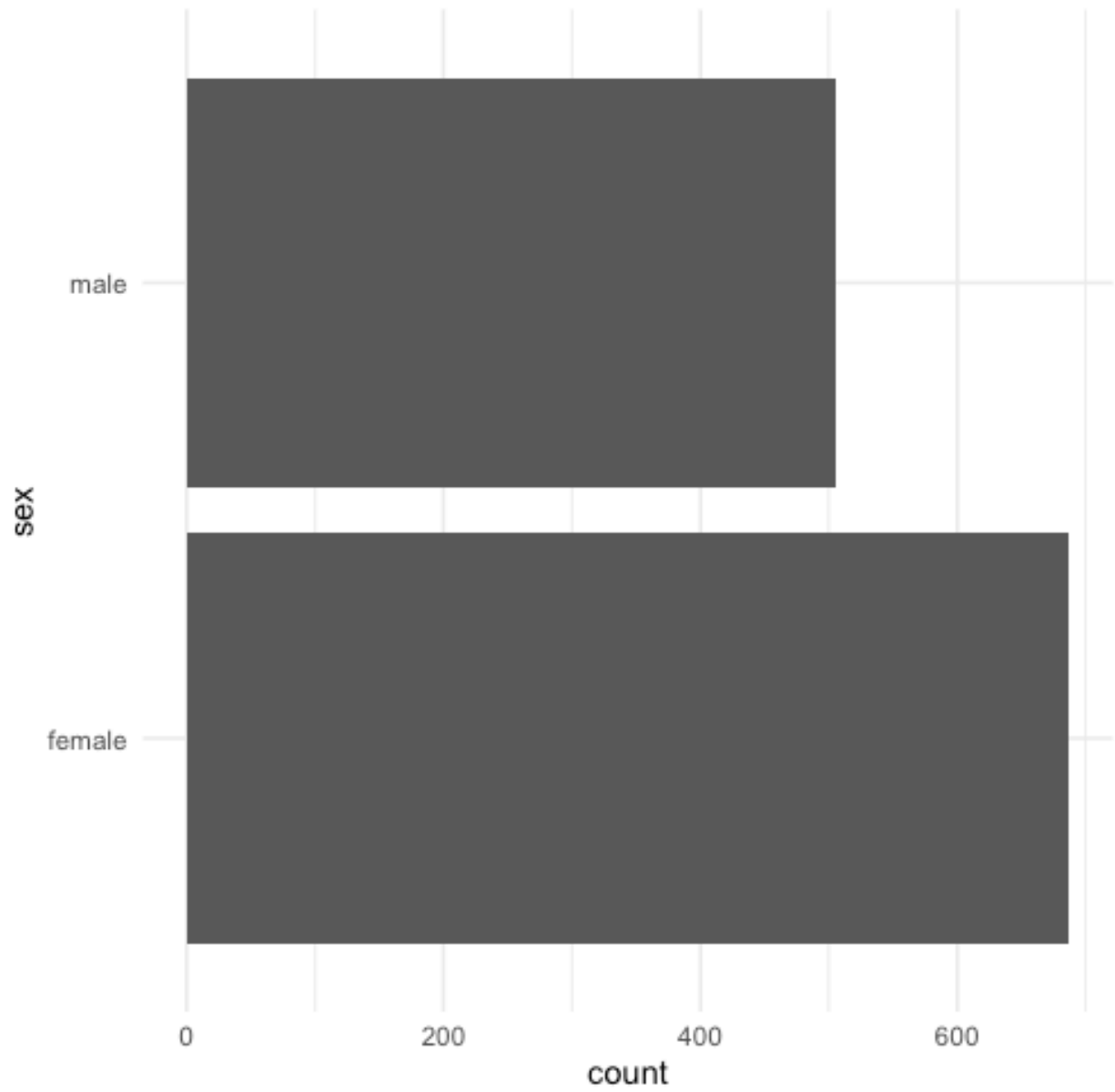
```
# https://ggplot2.tidyverse.org/reference/geom\_bar.html  
## Using `geom_bar()` plot a bar chart of the number of records for each `sex`  
ggplot(heights_df, aes(sex)) + geom_bar()
```



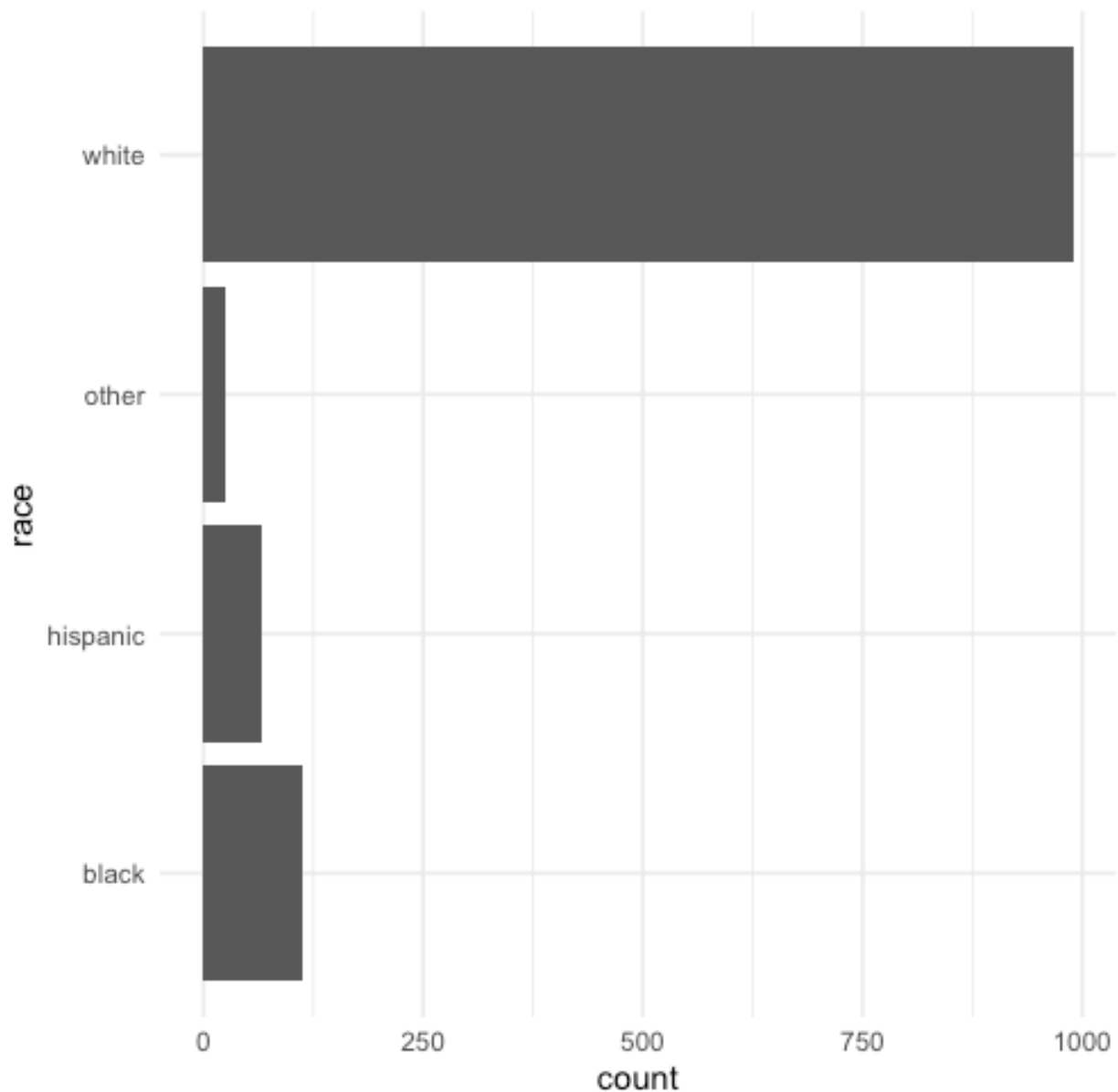
```
## Using `geom_bar()` plot a bar chart of the number of records for each race  
ggplot(heights_df, aes(race)) + geom_bar()
```



```
## Create a horizontal bar chart by adding `coord_flip()` to the previous plot  
ggplot(heights_df, aes(sex)) + geom_bar() + coord_flip()
```



```
ggplot(heights_df, aes(race)) + geom_bar() + coord_flip()
```

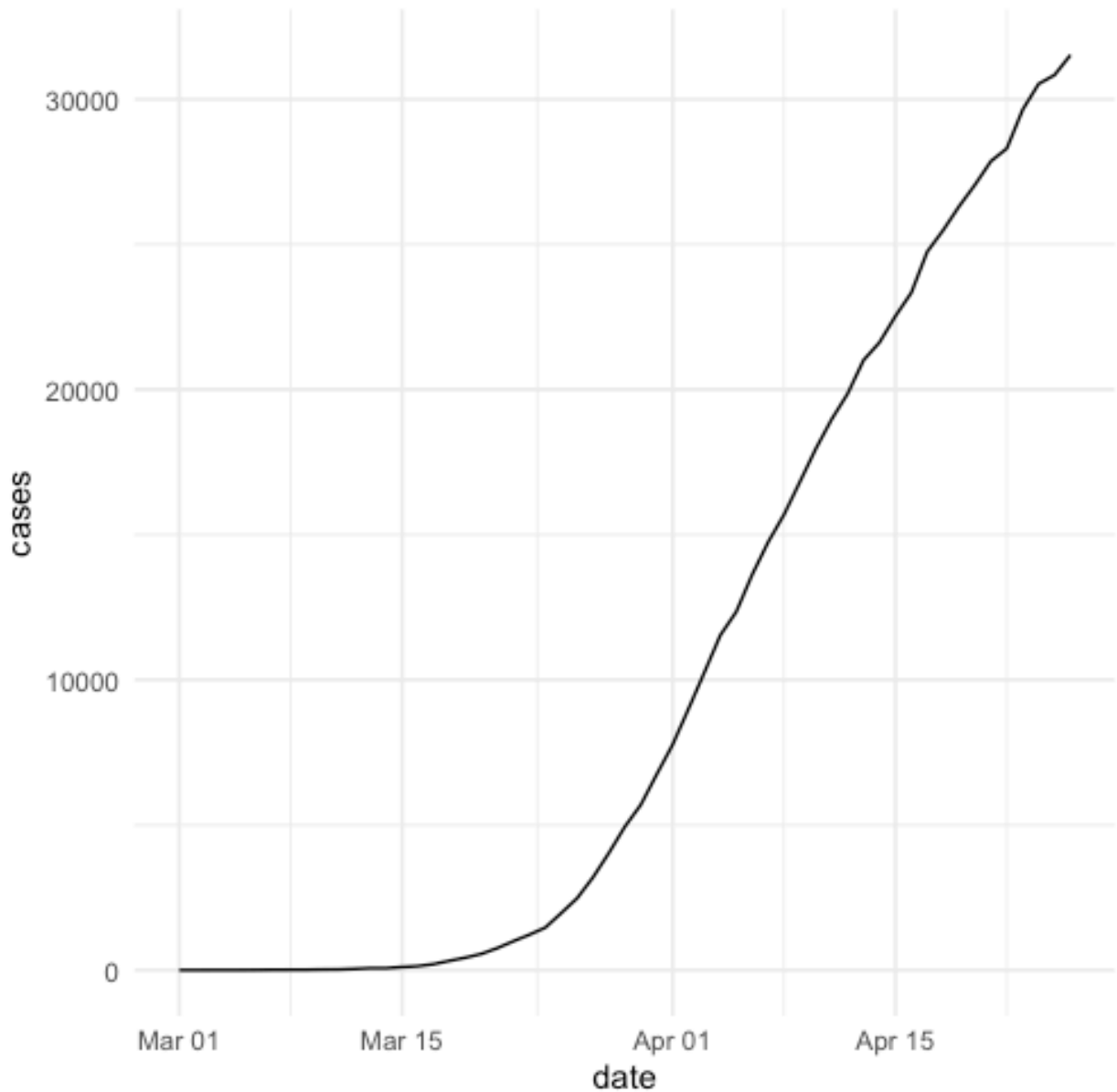


```
# https://www.rdocumentation.org/packages/ggplot2/versions/3.3.0/topics/geom\_path
## Load the file ``data/nytimes/covid-19-data/us-states.csv`` and
## assign it to the `covid_df` dataframe
covid_df <- read.csv("Users/dipikasharma/R_Projects/DSC520/data/nytimes/covid-19-data/us-
states.csv")
##covid_df
## Parse the date column using `as.Date()`
covid_df$date <- as.Date(covid_df$date)
covid_df
## Create three dataframes named `california_df`, `ny_df`, and `florida_df`
## containing the data from California, New York, and Florida
```

```

california_df <- covid_df[ which( covid_df$state == "California"), ]
ny_df <- covid_df[ which( covid_df$state == "New York"), ]
florida_df <- covid_df[ which( covid_df$state == "Florida"), ]
#california_df
#ny_df
#florida_df
## Plot the number of cases in Florida using `geom_line()`
ggplot(data=florida_df, aes(x=date, y=cases, group=1)) + geom_line()

```



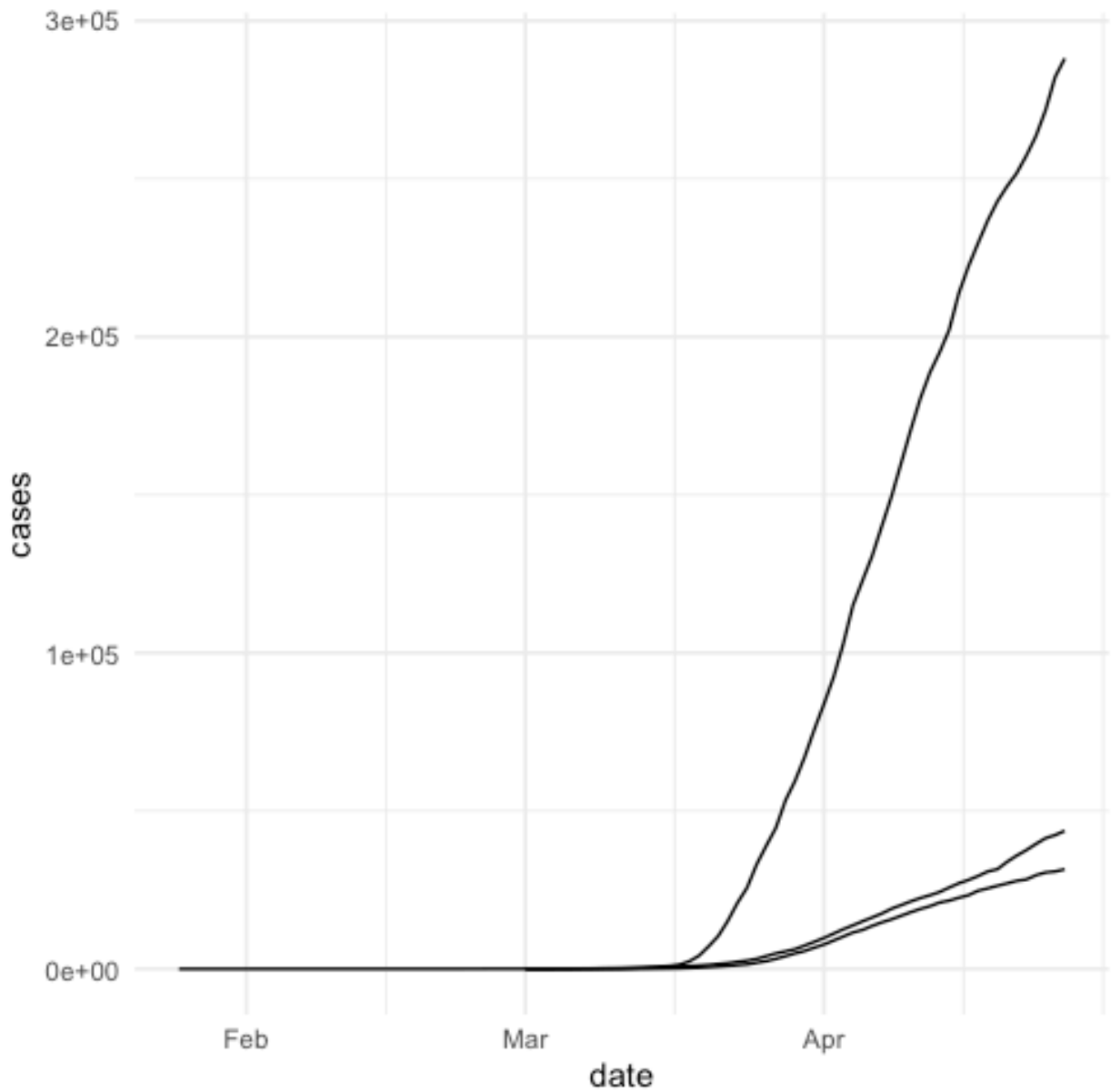
```

## Add lines for New York and California to the plot
ggplot(data=florida_df, aes(x=date, group=1)) +
  geom_line(aes(y = cases)) +

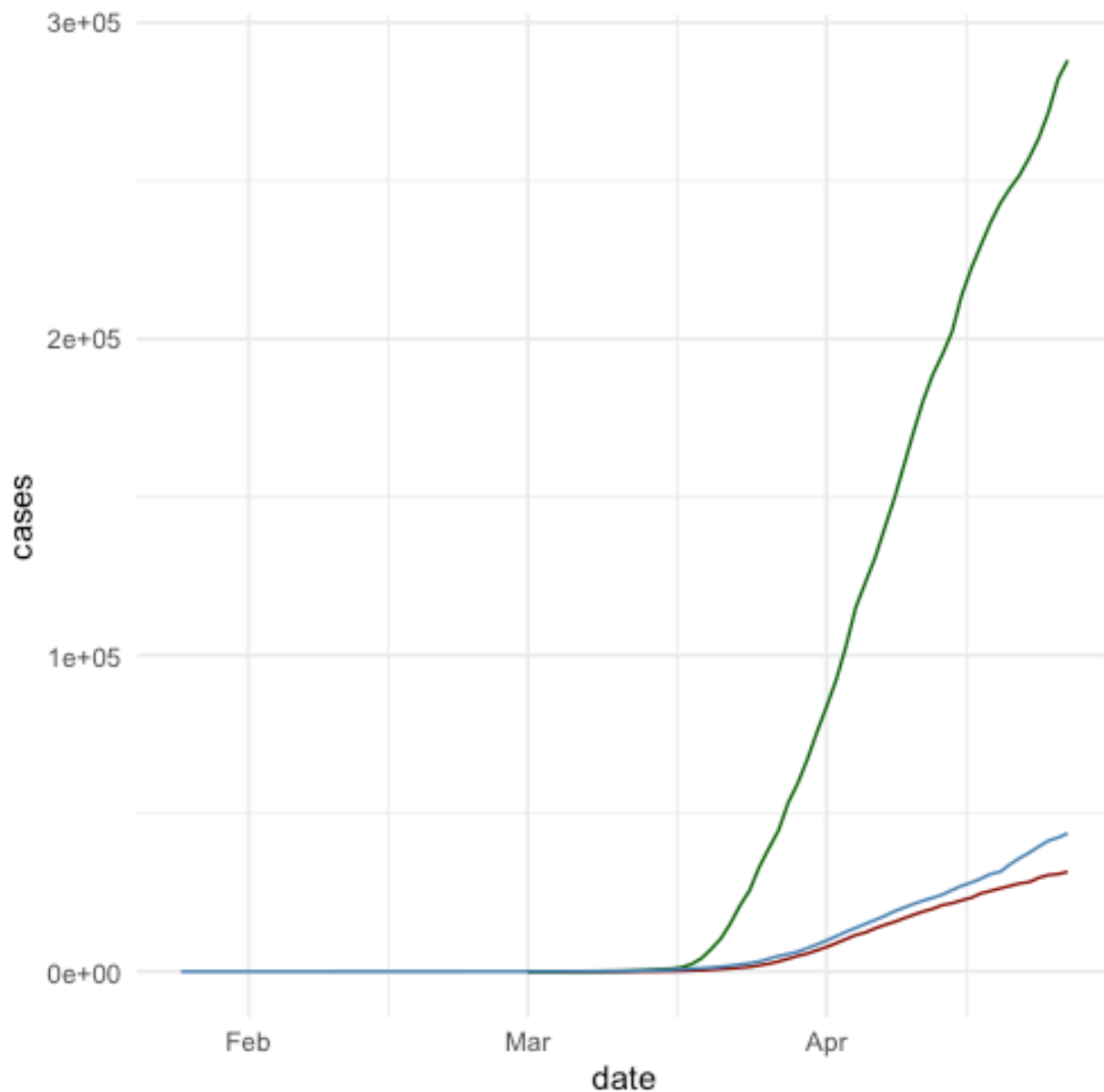
```



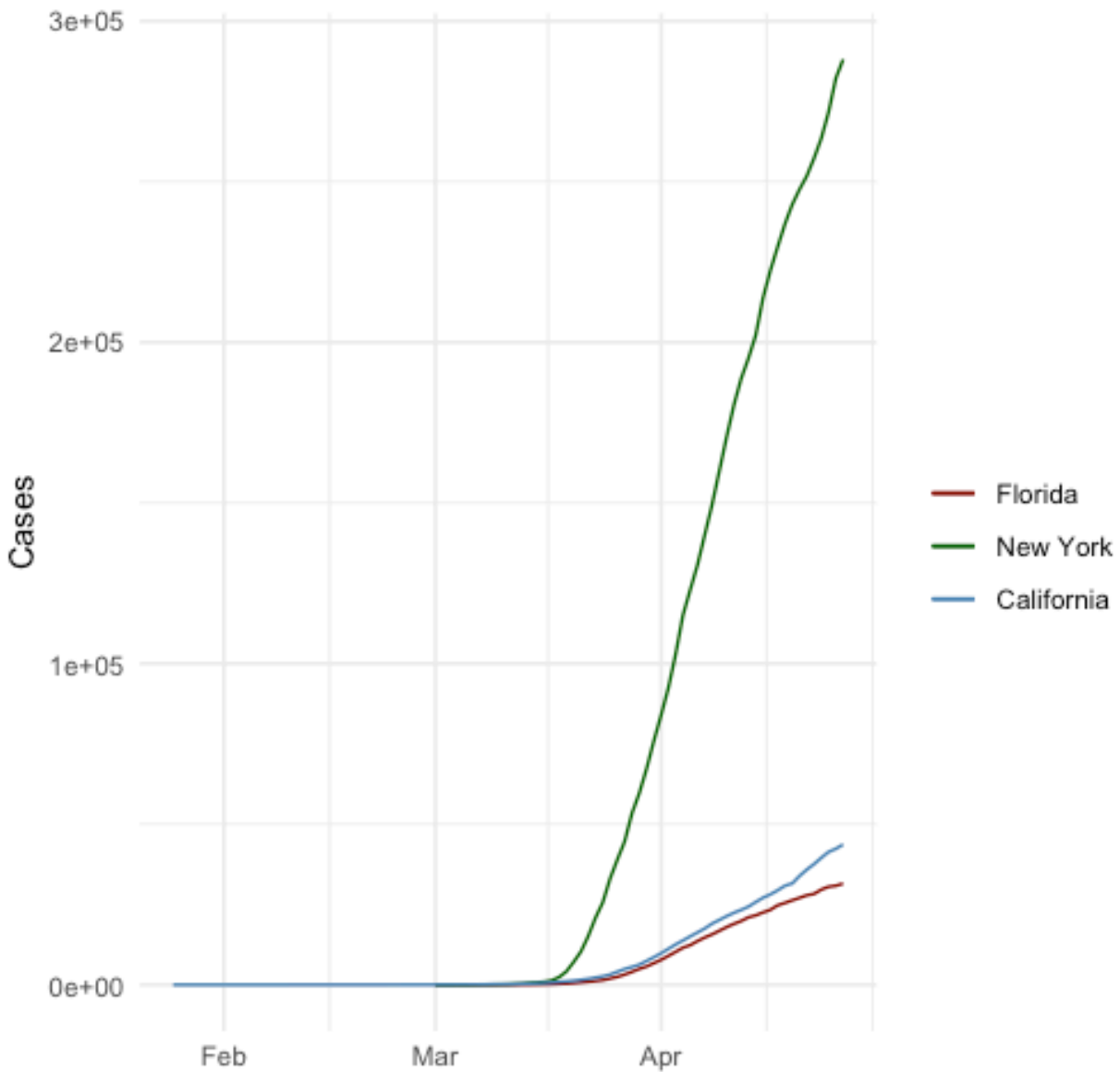
```
geom_line(data=ny_df, aes(y = cases)) +  
geom_line(data=california_df, aes(y = cases))
```



```
## Use the colors "darkred", "darkgreen", and "steelblue" for Florida, New York, and California  
ggplot(data=florida_df, aes(x=date, group=1)) +  
  geom_line(aes(y = cases), color = "darkred") +  
  geom_line(data=ny_df, aes(y = cases), color="darkgreen") +  
  geom_line(data=california_df, aes(y = cases), color="steelblue")
```



```
## Add a legend to the plot using `scale_colour_manual`
## Add a blank (" ") label to the x-axis and the label "Cases" to the y axis
ggplot(data=florida_df, aes(x=date, group=1)) +
  geom_line(aes(y = cases, colour = "Florida")) +
  geom_line(data=ny_df, aes(y = cases, colour="New York")) +
  geom_line(data=california_df, aes(y = cases, colour="California")) +
  scale_colour_manual("",
    breaks = c("Florida", "New York", "California"),
    values = c("darkred", "darkgreen", "steelblue")) +
  xlab(" ") + ylab("Cases")
```



```
## Scale the y axis using `scale_y_log10()`
ggplot(data=florida_df, aes(x=date, group=1)) +
  geom_line(aes(y = cases, colour = "Florida")) +
  geom_line(data=ny_df, aes(y = cases, colour="New York")) +
  geom_line(data=california_df, aes(y = cases, colour="California")) +
  scale_colour_manual("",
    breaks = c("Florida", "New York", "California"),
    values = c("darkred", "darkgreen", "steelblue")) +
  xlab(" ") + ylab("Cases") + scale_y_log10()
```

