10-import

I will look at 2018 mass shooting data that we used in previous class. I would like to see what states seem to have the biggest problems with this. I would also like to look at the deadliness/injurty rate of shootings relative to other shootings within each state and between the states.

To be able to better compare shooting data between states taking into account their population differences I will need to look at state population data along with the shooting data.

Hopefully this can help me identify states where shootings are a large problem and where they are less of a problem. Additionally I want to see what are the deadliest/injury fullest (?) shootings.

A great way to compare states in terms of how deadly/injury prone their shootings are would be if we looked at average death/injury rate. An even better way would be to look at deaths/injuries per 1 million people. This gives puts the shootings into perspectives because it takes into acount the amount of shootings in each state. For example, it would be better to live in a state thats had one shooting where 5 died, rather than one of similar size thats had fifty shootings where 4 have died. Altough the average would be lower - the deaths per 1 million would be much higher. A feature to find average deaths/injuries will be the first step into looking into the mass shooting rates as it is a commmon statistic to calculate. My features will build to calculate deaths/injuries per 1 million people by state!

import librarys

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------- tidyverse 1.2
```

```
## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

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

```
library(janitor)
```

```
##
## Attaching package: 'janitor'
```

```
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```
library(assertr)
library(dplyr)
library(ggplot2)
library(assertable)
```

import shooting and population data

The data that I will import below will be of two datasets - The first is the 2018 shooting data which has seven columns these columns include information on the date of the shooting, the name of the shooter + some victim info, number of killed, number of wounded, city of shooting, state of shooting, and the source. The second is the population data set which contains three columns; the state name, the state abreviation, and the state population in 2018.

```
# shooting data
s_data <- read_csv("./shooting_data.csv")
```

```
## Parsed with column specification:
## cols(
##   date = col_character(),
##   name_semicolon_delimited = col_character(),
##   killed = col_double(),
##   wounded = col_double(),
##   city = col_character(),
##   state = col_character(),
##   sources_semicolon_delimited = col_character()
## )
```

```
# population data
p_data <- read_csv("./pop_data.csv")
```

```
## Parsed with column specification:
## cols(
##   `STATE NAME` = col_character(),
##   `STATE ABRV` = col_character(),
##   POPULATION = col_double()
## )
```

logical checks of data upon intake (seperate pipelines to ensure all assertr statements run)

assertr statement to check killed # is reasonable: killed needs to be non negative

```
s_data <- s_data %>%
  # assert between 0 and infinity
  assert(within_bounds(0, Inf), 'killed')
```

assertr statement to check wounded # is reasonable: wounded needs to be non negative

```
s_data <- s_data %>%
  # assert between 0 and infinity
  assert(within_bounds(0, Inf), 'wounded')
```

assertr statement to check population # is reasonable: population needs to be non negative and less than us population at the time (~330 million)

```
p_data <- p_data %>%
  # assert between 0 and 330 million
  assert(within_bounds(0, 330000000), 'POPULATION')
```

cleaning shooting and population data

clean names to remove weird casing and variable spacing - also insure that there are no doubles only integers as number killed and wounded both have to be a whole number

```
s_data <- s_data %>% clean_names()
# change doubles to ints
s_data <- s_data %>% mutate_if(is.double,as.integer)
```

clean names to remove weird casing and variable spacing - insure doubles are integers as population must be a whole number

```
p_data <- p_data %>% clean_names()
# change doubles to ints
p_data <- p_data %>% mutate_if(is.double,as.integer)
```