The COVID19.csv data set is raw data extracted from Worldometer (March 1st, 2021). Use COVID19.csv data to answer the questions considering the following:

- The data set requires data cleaning and manipulation before any analysis.
- Explain your results with a numerical summary and graphs, wherever it is applicable.
- The results should be based on skewness or normal distribution, therefore, do check your data if it is skewed or normally distributed.

Step 1: Keep rows containing country information and remove the rest of the rows. Apart from the country, you should not be having any other row. Move on to the next step only after finishing this.

```
clean_data= read.csv("D:/R coding/Covid19.csv")
# a. Remove 1st 8 rows
clean_data = slice(raw_data, 9:n() )
clean_data = subset(clean_data, clean_data$Country.Other != "Total:")
head(clean_data)

# b. remove last 8 rows
clean_data = slice(clean_data, n():9)
tail(clean_data)
str(clean_data)

# c. Remove 1st and 2nd cols
clean_data2 = clean_data[-c(1,2)]
str(clean_data2)

# d.Replace nul values with 'Na'
clean_data2 <- clean_data2 %>% replace(.==" ", NA)
clean_data2 <- clean_data2 %>% replace(.=="", NA)
clean_data2
```

Step 2: Calculate the missing percentage of each column using a function. If any column has missing data more than 5%, please remove it. (Do not try this for rows)

```
na_percent = function(x){
  (sum(is.na(x))/length(x))*100
}
na_perc = apply(clean_data2, 2, na_percent)
na_perc = which(na_perc>10)
na_perc

# If any column has missing data more than 5%, please remove it.
clean_data2 = clean_data2[-c(na_perc)]
str(clean_data2)
```

Step 3: Give a better column name after cleaning your data

```
colnames(clean_data2)

colnames(clean_data2) = c ("Country","TotalCases", "TotalDeaths","TotalRecovered",
                           "ActiveCases","TotCasesPerMilPop", "DeathsPerMilPop",
                           "TotalTests","TestsPerMilPop","Population","Continent",
                           "XCaseeveryXppl","XDeatheveryXppl", "XTesteveryXppl" )
str(clean_data2)
```

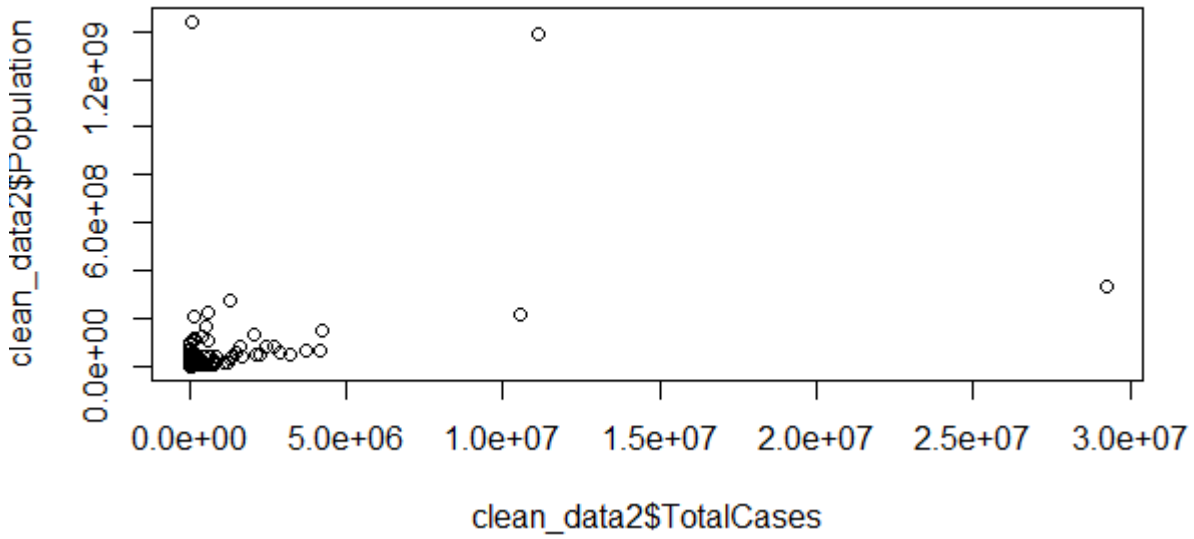
```
#converting variables to correct datatypes
clean_data2$Continent = as.factor(clean_data2$Continent)
# int_data = clean_data[c(2:10, 12:14)]
# str(int_data)



for (i in c(2:10, 12:14)){
  #int_data[,i] = as.numeric(gsub(",", "", int_data[,i]))
  clean_data2[,i] = as.numeric(gsub(",", "", clean_data2[,i]))

}
str(clean_data2)
#Data summary and skewness check
summary(clean_data2)

for (i in c(3:7, 9)){
  hist(clean_data2[,i], main = colnames(clean_data2[,i]))
}
#Data is skewed to the right
```

1. Create plots for total cases, total death, and total recovery. Explain with a figure for each

```
boxplot(clean_data2$TotalCases)
boxplot(cbind(clean_data2$TotalCases, clean_data2$TotalDeaths,
              clean_data2$TotalRecovered ), main = 'Cases plot',
        names = c('Total Cases', 'Total deaths', 'Total Recovery'),
        col = c("#999999", "#E69F00", "#56B4E9"),frame= FALSE, na.rm =TRUE)

library("ggpubr")
ggdensity(clean_data2$TotalCases,
          main = "Density plot for Total Cases",
          xlab = "Total Cases")

ggdensity(clean_data2$TotalDeaths,
          main = "Density plot for Total Cases",
          xlab = "Total Cases")

ggdensity(clean_data2$TotalRecovered,
          main = "Density plot for Total Cases",
          xlab = "Total Cases")
```

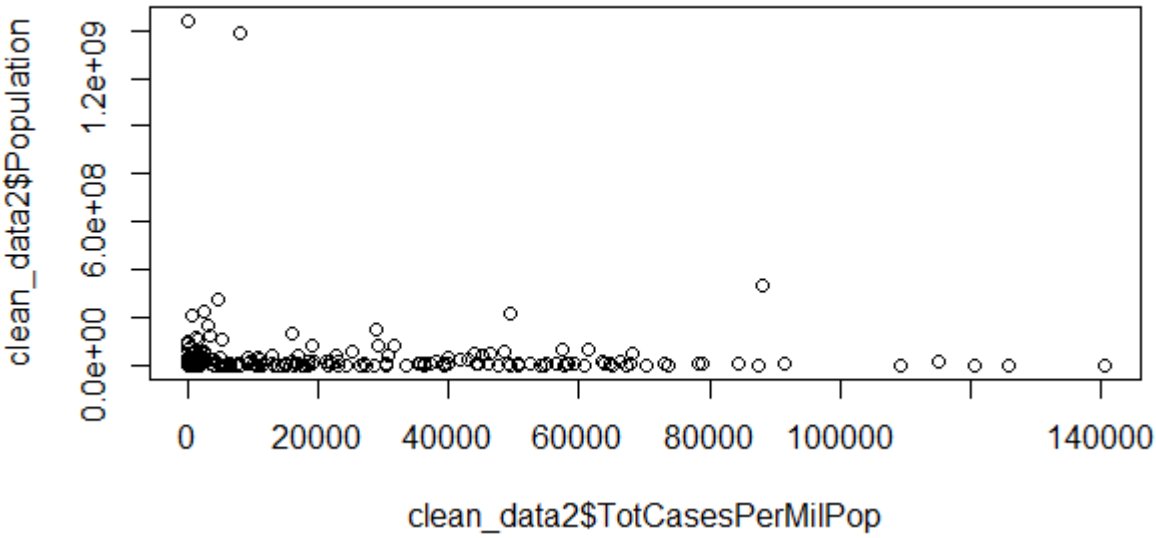
2. Create a plot to examine the correlation between total cases and total population. Explain if there is any correlation between total cases and total population.

```
plot(clean_data2$TotalCases, clean_data2$Population, type="p")
##There is no correlation between population and number of cases
```



3. Create a plot to examine the correlation between Tot Cases/1M pop and total population. Explain if there is any correlation between them?
   plot(clean_data2$TotCasesPerMilPop, clean_data2$Population, type="p")

   ## There is no correlation between these either



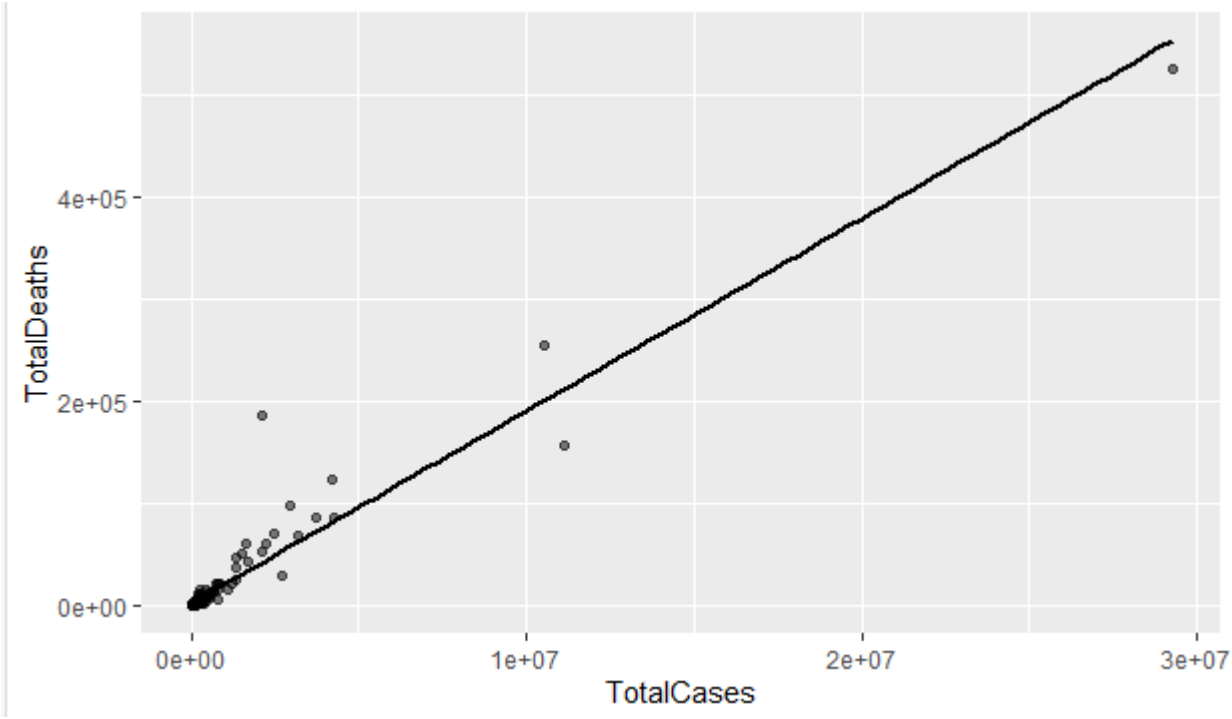
4. Which column do you feel is better for comparison purposes, total cases or TotCases/1M pop. Explain.

TotCases/1M pop has been scaled and such is better for comparison, as it is more resistant to differences in population. Also, a ratio value(TotCases/1M pop) works better when comparing.

5. Create a plot to examine the correlation between total cases and total death. Explain the figure.

The variables show weak correlation

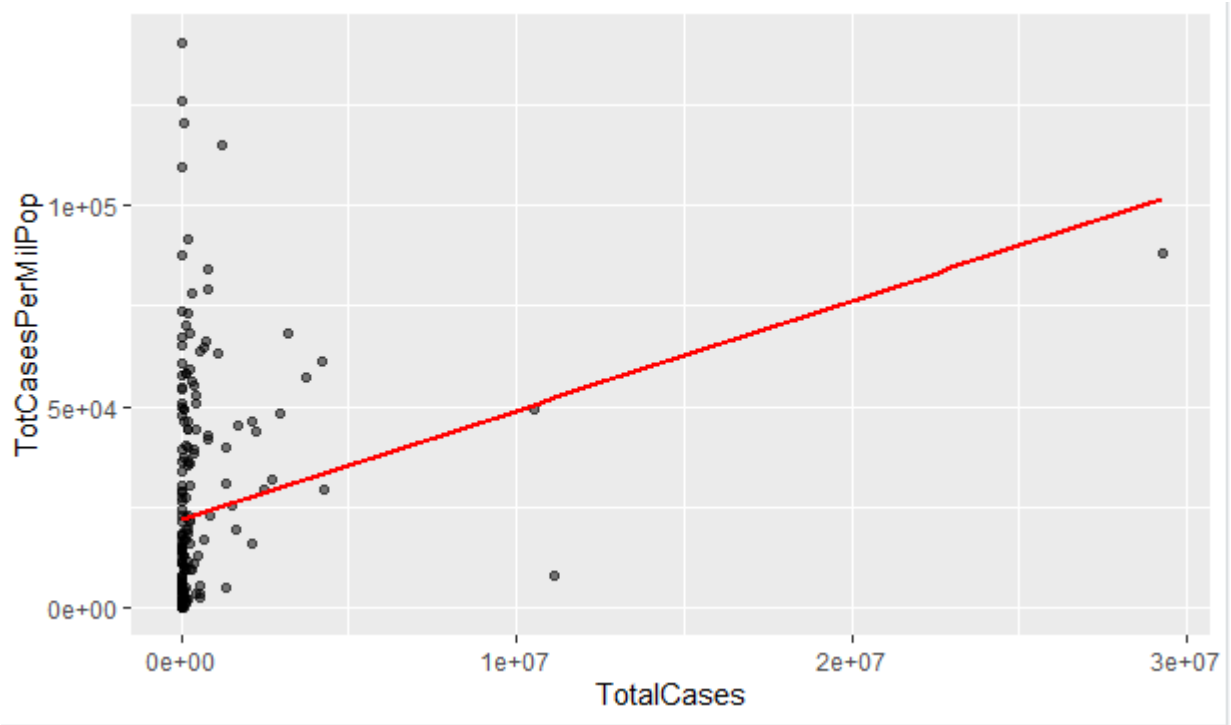
```
ggplot(clean_data2, aes(x = TotalCases, y = TotalDeaths ))+
  geom_point(alpha = 0.5)+
  stat_smooth(method = "lm", col = "black", se = FALSE)
```



6. Create a plot to examine the correlation between total cases and Deaths/1M pop. Explain the figure. Which column is more suitable to compare the result, total death or Death/1Mpop?

```
ggplot(clean_data2, aes(x = TotalCases, y = TotCasesPerMilPop ))+
  geom_point(alpha = 0.5)+
  stat_smooth(method = "lm", col = "red", se = FALSE)
```

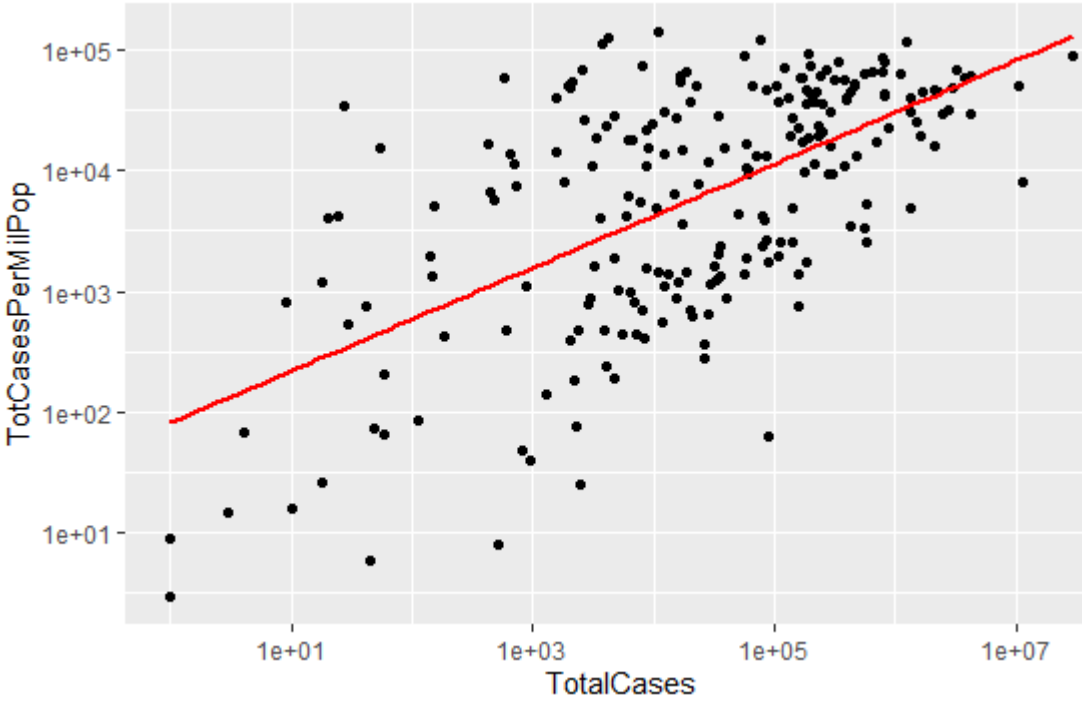
Finding: There is no correlation between these two variables, as such, it might be better to compare total death with total cases




```
#let's check again with log values
ggplot(clean_data2, aes(y=TotCasesPerMilPop, x=TotalCases)) +
  geom_point()  +
  coord_fixed()+
  scale_x_log10()+
```

```
  scale_y_log10()+
  stat_smooth(method = "lm", col = "red", se = FALSE)
```

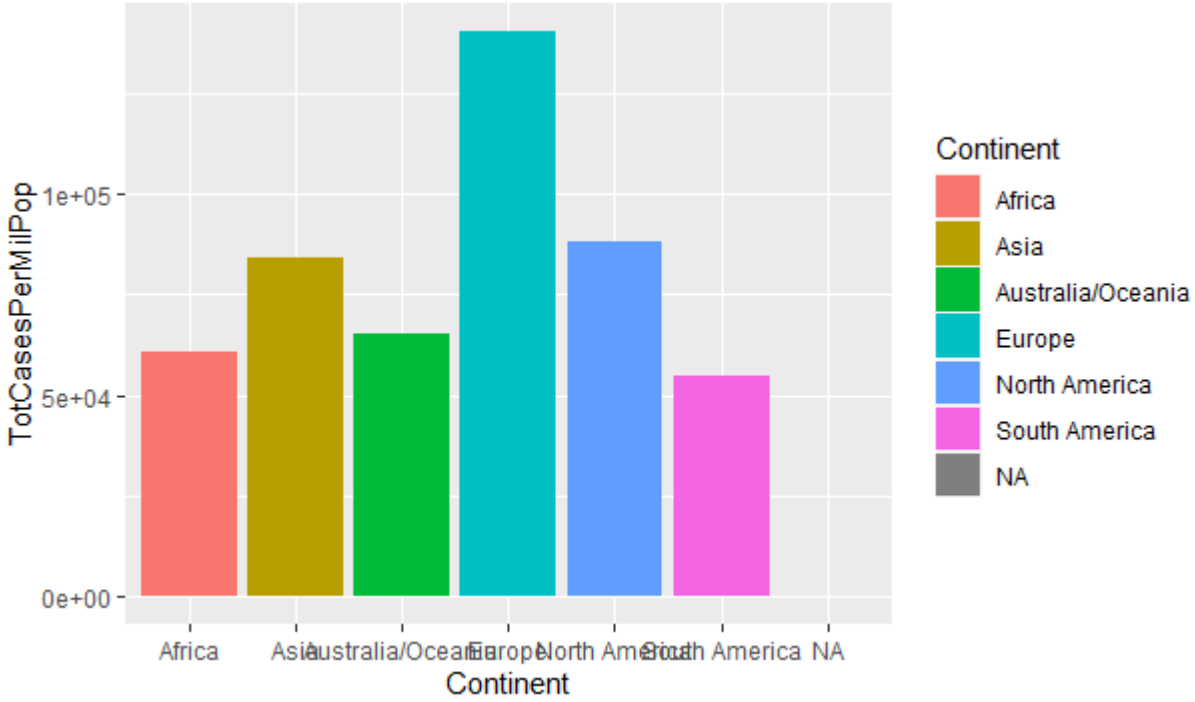
Finding: The correlation is still grossly weak



7. Compare Tot Cases/1M pop by continent, and explain your result.

```
ggplot(clean_data2, aes(x= Continent, y=TotCasesPerMilPop, fill=Continent)) +
  geom_bar(position = "dodge", stat = "identity")
```

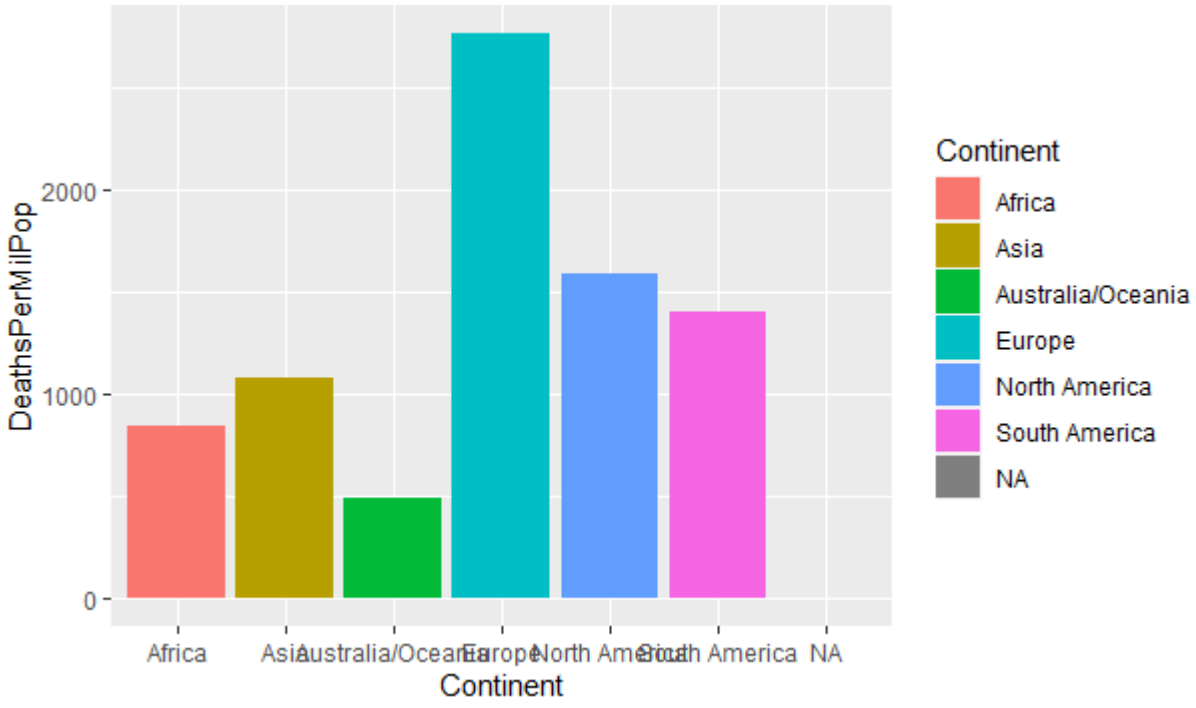
Finding: European countries have the highest cases per million population while Africa and South America have the least cases



8. Compare Deaths/1M pop by continent, and explain your result.

```
ggplot(clean_data2, aes(x= Continent, y=DeathsPerMilPop, fill=Continent)) +
  geom_bar(position = "dodge", stat = "identity")
```

Finding: Europe has the highest deaths per million population while Australia has the least

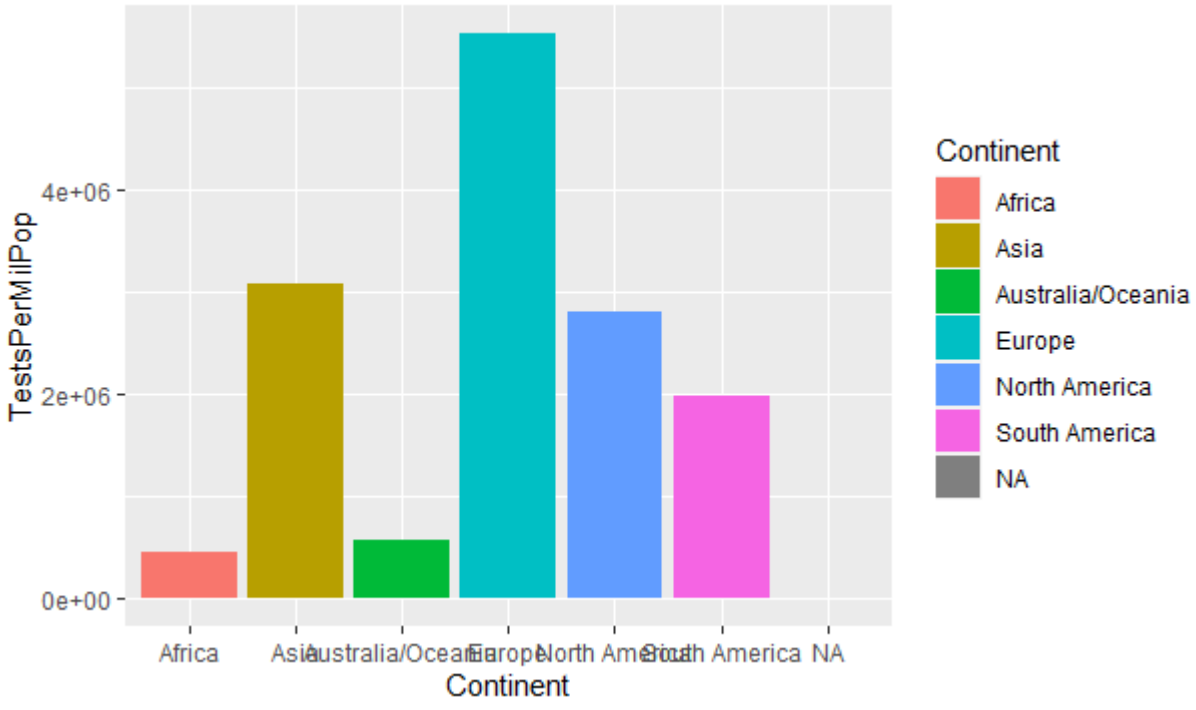


9. Which country is best among testing the COVID19 and which country is worst? There are two columns total test vs. test/M. Choose appropriate column

```
> clean_data2$Country[which.max(clean_data2$TestsPerMilPop)]
[1] "Gibraltar"
> clean_data2$Country[which.min(clean_data2$TestsPerMilPop)]
[1] "Yemen"
>
```

10. Compare your COVID19 test results by continent? There are two columns total test vs test/M. Choose appropriate column

```
Levels: Africa Asia Australia/Oceania Europe North America South America
> clean_data2$Continent[which.min(clean_data2$TestsPerMilPop)]
[1] Asia
Levels: Africa Asia Australia/Oceania Europe North America South America
>
> ggplot(clean_data2, aes(x= Continent, y=TestsPerMilPop, fill=Continent)) +
+   geom_bar(position = "dodge", stat = "identity")
```



11. Check if Tests/1M pop is skewed or normally distributed.

```
ggdensity(clean_data2$TestsPerMilPop,
        main = "Density plot for Test per million population")
```