```r
setwd("D:/RISE - WPU/R Studio")
getwd()
spotify = read.csv("Spotify.csv")
View(spotify)
dim(spotify)
summary(spotify)
str(spotify)
names(spotify)

# Plotting bar graph for beats.per.minute
barplot(spotify$Beats.Per.Minute)

# Hsitogram
hist(spotify$Beats.Per.Minute)

hist(spotify$Beats.Per.Minute,breaks = 6,
     main = "Distribution of Beats per Minute",
     xlab = "Beats per minute",
     ylab = "Frequency of songs",
     col = "#6495ED")

# Histogram for Energy col
hist(spotify$Energy)

hist(spotify$Energy,
     main = "Distribution of Energy",
     xlab = "Energy",
     ylab = "Frequency of songs",
     col = "#6495ED")

# Histogram for Liveness col
hist(spotify$Liveness,breaks = 4,
     main = "Distribution of Liveness",
     xlab = "Liveness",
     ylab = "Frequency of songs",
     col = "#9FE2BF")

# Histogram for Valence col
hist(spotify$Valence,breaks = 4,
     main = "Distribution of valence",
     xlab = "valence",
     ylab = "Frequency of songs",
     col = "#9FE2BF")

# Box Plot for Length

boxplot(spotify$Length.,
        horizontal = T,
        notch = T,
        main = "Box Plot for length of the song",
        xlab = "Length of the song",
        sub = "Length is in seconds",
        col = "Blue")

# Box Plot for Beats per minute
```

```r
boxplot(spotify$Beats.Per.Minute,
        horizontal = T,
        notch = F,
        main = "Box Plot for Beats per minute",
        xlab = "Beats per minute",
        col = "Blue")

# Scatter plot for speechiness n popularity

plot(spotify$Speechiness., spotify$Popularity,col = "Green")

plot(spotify$Loudness..dB.., spotify$Danceability,col = "Green")

# Pie Chart
library(dplyr)
top3 = spotify %>% group_by(Genre) %>% summarise(AvgLive=mean(Liveness)) %>%
arrange(desc(AvgLive)) %>% head(3)
View(top3)

pie(top3$AvgLive, top3$Genre,
    radius = 1,
    clockwise = T,
    col = rainbow(4))

# HeatMap

x_cor = spotify[,5:14]
View(x_cor)

cor_mat = cor(x_cor)
cor_mat

install.packages("pheatmap")
library(pheatmap)

pheatmap(cor_mat, display_numbers = T)


# Data Visualization using ggplot2
# Components of ggplot:
# 1. Data : dataframe created or imported
# 2. Aesthetics : color, size or shape
# 3. Geometry : geom() - used along with the + operator
# 4. Statistics : aggregating & cleaning the data
# 5. Facets : Splitting of charts
# 6. Co-ordinates : x & y axis
# 7. Theme : add attractiveness to the chart

# various plots available
# histogram - geom_histogram()
# density - geom_density()
# bar or column - geom_bar(), to flip the co-ordinates: coord_flip()
# Scatter - geom_point(), geom_smooth()
# line - geom_line()

install.packages("ggplot2")
```

```r
library(ggplot2)
library(dplyr)

?economics

df = economics
View(economics)

names(df)

# Create a line chart for date and unemployment

line_graph = ggplot(df,aes(x=date,y=unemploy)) +geom_line(col = "Blue",
                                                    size = 1)
line_graph

?geom_line

#ggtitle

line2 = line_graph +ggtitle("Unemployment Rate of USA")
line2

# Change background

line3 = line2 + theme_minimal()
line3

library(readxl)
cement = read_excel("Cement Consumption.xlsx")
View(cement)
names(cement)

line1 <- ggplot(cement,aes(x=Month, y=`Cement Consumption`)) +
geom_line(colour = 'blue')
line1


# Scatter plot
?faithful

df5 <- faithful
View(df5)

scatter1 <- ggplot(df5,aes(x = eruptions, y = waiting,color = eruptions>3)) +
geom_point(pch=7)
scatter1

# Box Plot
# DC Store sales

store = read_excel("DC Store Sales.xlsx")
View(store)

# Plot city by total sales
```

```r
data1 = store %>% group_by(City) %>% summarise(Total_sales = sum(Total))
View(data1)

bar_gp = ggplot(data1, aes(x=City, y=Total_sales))
bar_gp2 = bar_gp + geom_bar(stat = "identity", width = 0.5, aes(fill = City))
+theme_classic() +labs(title = "City wise Sales") +theme(legend.position =
"bottom")
bar_gp2

bar_gp3 = bar_gp2 +coord_flip()
bar_gp3
```