RWorksheet_Octaviano#4b

Jirraine S. Octaviano

2023-11-08

Using Loop Function for() loop

1. Using the for loop, create an R script that will display a 5x5 matrix as shown in Figure 1. It must contain vectorA = [1,2,3,4,5] and a 5 x 5 zero matrix.

Hint Use abs() function to get the absolute value

```
matZero <- matrix(c(0,0,0,0,0), 5, 5)
vecTorA <- c(1,2,3,4,5)

for (i in 1:length(vecTorA)) {
  matZero[i, ] <- abs(vecTorA - vecTorA[i])
}
print(matZero)
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    0    1    2    3    4
## [2,]    1    0    1    2    3
## [3,]    2    1    0    1    2
## [4,]    3    2    1    0    1
## [5,]    4    3    2    1    0
```

2. Print the string "*" using for() function. The output should be the same as shown in Figure

```
  for(i in 1:5) {
  stars <- rep("*", i)
  print(stars)
}
```

```
## [1] "*"
## [1] "*" "*"
## [1] "*" "*" "*"
## [1] "*" "*" "*" "*"
## [1] "*" "*" "*" "*" "*"
```

3. Get an input from the user to print the Fibonacci sequence starting from the 1st input up to 500. Use repeat and break statements. Write the R Scripts and its output.

```
userInput<- as.integer(readline("Enter starting number for Fibonacci sequence: "))
```

```
## Enter starting number for Fibonacci sequence:
```

```
if(is.na(userInput || userInput < 0)) {
  cat("Please Enter Something")
} else {

x <- userInput
```

```
y <- 0


cat("Fibonacci sequence starting from", userInput, ":\n")

repeat {

  next_num <- x + y

  if (next_num > 500){
    break
  }
  cat(next_num, " ")
  x <- y
  y <- next_num

}
}
```

```
## Please Enter Something
```

Using Basic Graphics (plot(),barplot(),pie(),hist())

4. Import the dataset as shown in Figure 1 you have created previously.

A. What is the R script for importing an excel or a csv file? Display the first 6 rows of the dataset? Show your codes and its result.

```
prevData <- read.csv("HouseHold DATA.csv")
head(prevData)
```

```
##   Shoe.Size Height Gender
## 1       6.5   66.0      F
## 2       9.0   68.0      F
## 3       8.5   64.5      F
## 4       8.5   65.0      F
## 5      10.5   70.0      M
## 6       7.0   64.0      F
```

B. Create a subset for gender(female and male). How many observations are there in Male? How about in Female? Write the R scripts and its output.

```
males <- prevData[prevData$Gender == "M",]
males
```

```
##    Shoe.Size Height Gender
## 5       10.5   70.0      M
## 9       13.0   72.0      M
## 11      10.5   74.5      M
## 13      12.0   71.0      M
## 14      10.5   71.0      M
## 15      13.0   77.0      M
## 16      11.5   72.0      M
## 19      10.0   72.0      M
## 22       8.5   67.0      M
## 23      10.5   73.0      M
## 25      10.5   72.0      M
```

```
## 26      11.0   70.0      M
## 27       9.0   69.0      M
## 28      13.0   70.0      M
```

```
females <- prevData[prevData$Gender == "F",]
females
```

```
##    Shoe.Size Height Gender
## 1        6.5   66.0      F
## 2        9.0   68.0      F
## 3        8.5   64.5      F
## 4        8.5   65.0      F
## 6        7.0   64.0      F
## 7        9.5   70.0      F
## 8        9.0   71.0      F
## 10       7.5   64.0      F
## 12       8.5   67.0      F
## 17       8.5   59.0      F
## 18       5.0   62.0      F
## 20       6.5   66.0      F
## 21       7.5   64.0      F
## 24       8.5   69.0      F
```

```
numofMale <- nrow(males)
numofMale
```

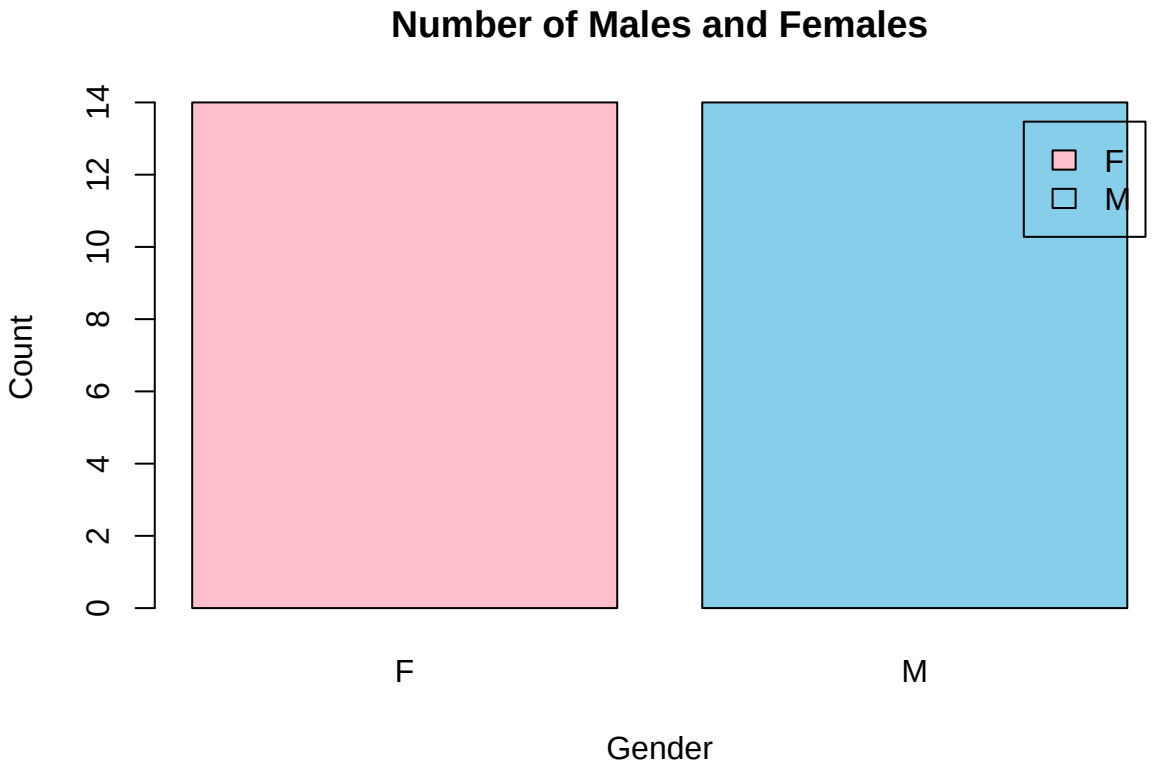
```
## [1] 14
```

```
numofFem <- nrow(females)
numofFem
```

```
## [1] 14
```

C. Create a graph for the number of males and females for Household Data. Use plot(),chart type = barplot. Make sure to place title, legends, and colors. Write the R scripts and its result.

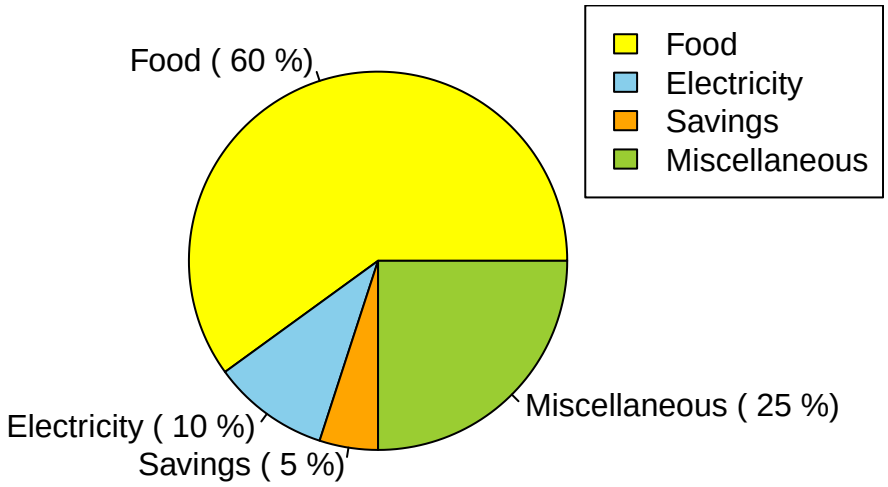
```
totalMaleFemale <- table(prevData$Gender)
barplot(totalMaleFemale,
        main = "Number of Males and Females",
        xlab = "Gender",
        ylab = "Count",
        col = c("pink", "skyblue"),
        legend.text = rownames(totalMaleFemale),
        beside = TRUE)
```



5. The monthly income of Dela Cruz family was spent on the following: A. Create a piechart that will include labels in percentage.Add some colors and title of the chart. Write the R scripts and show its output.

```r
spending_data <- data.frame(
  Category = c("Food", "Electricity", "Savings", "Miscellaneous"),
  Value = c(60, 10, 5, 25)
)

spending_data$Percentage <- spending_data$Value / sum(spending_data$Value) * 100

colors <- c("yellow", "skyblue", "orange", "yellowgreen")

pie(spending_data$Value,
    labels = paste(spending_data$Category, "(",spending_data$Percentage,"%)"),
    col = colors,
    main = "Monthly Income Spending of Dela Cruz Family")

legend("topright", spending_data$Category, fill = colors)
```

Monthly Income Spending of Dela Cruz Family



6. Use the iris dataset

```
data(iris)
```

A. Check for the structure of the dataset using the str() function. Describe what you have seen in the output.

```
str(iris)
```

```
## 'data.frame':    150 obs. of  5 variables:
##  $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

```
## -The dataset provides information on 150 different iris flowers, including their sepal and petal leng
```

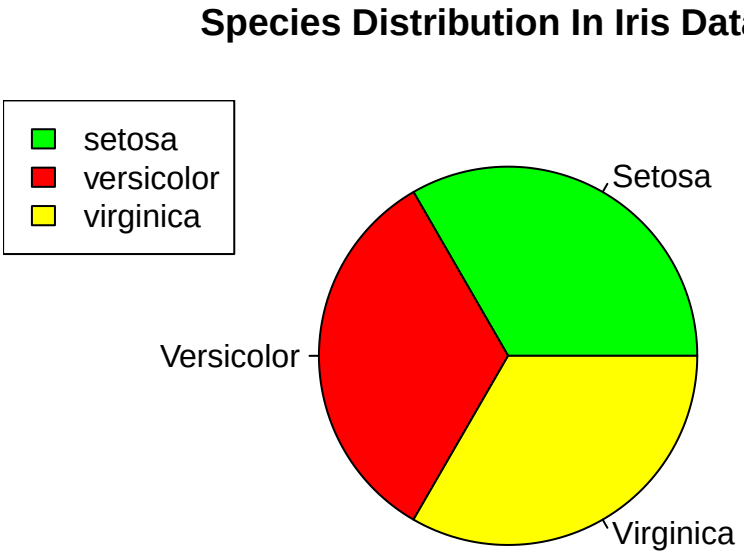
B. Create an R object that will contain the mean of the sepal.length, sepal.width,petal.length,and petal.width. What is the R script and its result?

```
meanOfFlowerS <- colMeans(iris[,1:4])
meanOfFlowerS
```

```
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width
##     5.843333     3.057333     3.758000     1.199333
```

C. Create a pie chart for the Species distribution. Add title, legends, and colors. Write the R script and its result.

```
specieS <- table(iris$Species)
nameOfspecieS <- c("Setosa", "Versicolor", "Virginica")

pie(specieS,
    labels = nameOfspecieS,
    col = c("green", "red", "yellow"),
    main = "Species Distribution In Iris Dataset")

legend("topleft", legend = levels(iris$Species), fill = c("green", "red", "yellow"),)
```



D. Subset the species into setosa, versicolor, and virginica. Write the R scripts and show the last six (6) rows of each species.

```
iris
```

```
##     Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
## 1            5.1         3.5          1.4         0.2     setosa
## 2            4.9         3.0          1.4         0.2     setosa
## 3            4.7         3.2          1.3         0.2     setosa
## 4            4.6         3.1          1.5         0.2     setosa
## 5            5.0         3.6          1.4         0.2     setosa
## 6            5.4         3.9          1.7         0.4     setosa
## 7            4.6         3.4          1.4         0.3     setosa
## 8            5.0         3.4          1.5         0.2     setosa
## 9            4.4         2.9          1.4         0.2     setosa
## 10           4.9         3.1          1.5         0.1     setosa
## 11           5.4         3.7          1.5         0.2     setosa
## 12           4.8         3.4          1.6         0.2     setosa
## 13           4.8         3.0          1.4         0.1     setosa
## 14           4.3         3.0          1.1         0.1     setosa
## 15           5.8         4.0          1.2         0.2     setosa
## 16           5.7         4.4          1.5         0.4     setosa
## 17           5.4         3.9          1.3         0.4     setosa
## 18           5.1         3.5          1.4         0.3     setosa
## 19           5.7         3.8          1.7         0.3     setosa
## 20           5.1         3.8          1.5         0.3     setosa
## 21           5.4         3.4          1.7         0.2     setosa
## 22           5.1         3.7          1.5         0.4     setosa
## 23           4.6         3.6          1.0         0.2     setosa
## 24           5.1         3.3          1.7         0.5     setosa
## 25           4.8         3.4          1.9         0.2     setosa
## 26           5.0         3.0          1.6         0.2     setosa
## 27           5.0         3.4          1.6         0.4     setosa
## 28           5.2         3.5          1.5         0.2     setosa
## 29           5.2         3.4          1.4         0.2     setosa
## 30           4.7         3.2          1.6         0.2     setosa
## 31           4.8         3.1          1.6         0.2     setosa
```

```
## 32          5.4         3.4          1.5         0.4     setosa
## 33          5.2         4.1          1.5         0.1     setosa
## 34          5.5         4.2          1.4         0.2     setosa
## 35          4.9         3.1          1.5         0.2     setosa
## 36          5.0         3.2          1.2         0.2     setosa
## 37          5.5         3.5          1.3         0.2     setosa
## 38          4.9         3.6          1.4         0.1     setosa
## 39          4.4         3.0          1.3         0.2     setosa
## 40          5.1         3.4          1.5         0.2     setosa
## 41          5.0         3.5          1.3         0.3     setosa
## 42          4.5         2.3          1.3         0.3     setosa
## 43          4.4         3.2          1.3         0.2     setosa
## 44          5.0         3.5          1.6         0.6     setosa
## 45          5.1         3.8          1.9         0.4     setosa
## 46          4.8         3.0          1.4         0.3     setosa
## 47          5.1         3.8          1.6         0.2     setosa
## 48          4.6         3.2          1.4         0.2     setosa
## 49          5.3         3.7          1.5         0.2     setosa
## 50          5.0         3.3          1.4         0.2     setosa
## 51          7.0         3.2          4.7         1.4 versicolor
## 52          6.4         3.2          4.5         1.5 versicolor
## 53          6.9         3.1          4.9         1.5 versicolor
## 54          5.5         2.3          4.0         1.3 versicolor
## 55          6.5         2.8          4.6         1.5 versicolor
## 56          5.7         2.8          4.5         1.3 versicolor
## 57          6.3         3.3          4.7         1.6 versicolor
## 58          4.9         2.4          3.3         1.0 versicolor
## 59          6.6         2.9          4.6         1.3 versicolor
## 60          5.2         2.7          3.9         1.4 versicolor
## 61          5.0         2.0          3.5         1.0 versicolor
## 62          5.9         3.0          4.2         1.5 versicolor
## 63          6.0         2.2          4.0         1.0 versicolor
## 64          6.1         2.9          4.7         1.4 versicolor
## 65          5.6         2.9          3.6         1.3 versicolor
## 66          6.7         3.1          4.4         1.4 versicolor
## 67          5.6         3.0          4.5         1.5 versicolor
## 68          5.8         2.7          4.1         1.0 versicolor
## 69          6.2         2.2          4.5         1.5 versicolor
## 70          5.6         2.5          3.9         1.1 versicolor
## 71          5.9         3.2          4.8         1.8 versicolor
## 72          6.1         2.8          4.0         1.3 versicolor
## 73          6.3         2.5          4.9         1.5 versicolor
## 74          6.1         2.8          4.7         1.2 versicolor
## 75          6.4         2.9          4.3         1.3 versicolor
## 76          6.6         3.0          4.4         1.4 versicolor
## 77          6.8         2.8          4.8         1.4 versicolor
## 78          6.7         3.0          5.0         1.7 versicolor
## 79          6.0         2.9          4.5         1.5 versicolor
## 80          5.7         2.6          3.5         1.0 versicolor
## 81          5.5         2.4          3.8         1.1 versicolor
## 82          5.5         2.4          3.7         1.0 versicolor
## 83          5.8         2.7          3.9         1.2 versicolor
## 84          6.0         2.7          5.1         1.6 versicolor
## 85          5.4         3.0          4.5         1.5 versicolor
```

```
## 86          6.0         3.4          4.5         1.6 versicolor
## 87          6.7         3.1          4.7         1.5 versicolor
## 88          6.3         2.3          4.4         1.3 versicolor
## 89          5.6         3.0          4.1         1.3 versicolor
## 90          5.5         2.5          4.0         1.3 versicolor
## 91          5.5         2.6          4.4         1.2 versicolor
## 92          6.1         3.0          4.6         1.4 versicolor
## 93          5.8         2.6          4.0         1.2 versicolor
## 94          5.0         2.3          3.3         1.0 versicolor
## 95          5.6         2.7          4.2         1.3 versicolor
## 96          5.7         3.0          4.2         1.2 versicolor
## 97          5.7         2.9          4.2         1.3 versicolor
## 98          6.2         2.9          4.3         1.3 versicolor
## 99          5.1         2.5          3.0         1.1 versicolor
## 100         5.7         2.8          4.1         1.3 versicolor
## 101         6.3         3.3          6.0         2.5  virginica
## 102         5.8         2.7          5.1         1.9  virginica
## 103         7.1         3.0          5.9         2.1  virginica
## 104         6.3         2.9          5.6         1.8  virginica
## 105         6.5         3.0          5.8         2.2  virginica
## 106         7.6         3.0          6.6         2.1  virginica
## 107         4.9         2.5          4.5         1.7  virginica
## 108         7.3         2.9          6.3         1.8  virginica
## 109         6.7         2.5          5.8         1.8  virginica
## 110         7.2         3.6          6.1         2.5  virginica
## 111         6.5         3.2          5.1         2.0  virginica
## 112         6.4         2.7          5.3         1.9  virginica
## 113         6.8         3.0          5.5         2.1  virginica
## 114         5.7         2.5          5.0         2.0  virginica
## 115         5.8         2.8          5.1         2.4  virginica
## 116         6.4         3.2          5.3         2.3  virginica
## 117         6.5         3.0          5.5         1.8  virginica
## 118         7.7         3.8          6.7         2.2  virginica
## 119         7.7         2.6          6.9         2.3  virginica
## 120         6.0         2.2          5.0         1.5  virginica
## 121         6.9         3.2          5.7         2.3  virginica
## 122         5.6         2.8          4.9         2.0  virginica
## 123         7.7         2.8          6.7         2.0  virginica
## 124         6.3         2.7          4.9         1.8  virginica
## 125         6.7         3.3          5.7         2.1  virginica
## 126         7.2         3.2          6.0         1.8  virginica
## 127         6.2         2.8          4.8         1.8  virginica
## 128         6.1         3.0          4.9         1.8  virginica
## 129         6.4         2.8          5.6         2.1  virginica
## 130         7.2         3.0          5.8         1.6  virginica
## 131         7.4         2.8          6.1         1.9  virginica
## 132         7.9         3.8          6.4         2.0  virginica
## 133         6.4         2.8          5.6         2.2  virginica
## 134         6.3         2.8          5.1         1.5  virginica
## 135         6.1         2.6          5.6         1.4  virginica
## 136         7.7         3.0          6.1         2.3  virginica
## 137         6.3         3.4          5.6         2.4  virginica
## 138         6.4         3.1          5.5         1.8  virginica
## 139         6.0         3.0          4.8         1.8  virginica
```

```
## 140          6.9         3.1          5.4         2.1  virginica
## 141          6.7         3.1          5.6         2.4  virginica
## 142          6.9         3.1          5.1         2.3  virginica
## 143          5.8         2.7          5.1         1.9  virginica
## 144          6.8         3.2          5.9         2.3  virginica
## 145          6.7         3.3          5.7         2.5  virginica
## 146          6.7         3.0          5.2         2.3  virginica
## 147          6.3         2.5          5.0         1.9  virginica
## 148          6.5         3.0          5.2         2.0  virginica
## 149          6.2         3.4          5.4         2.3  virginica
## 150          5.9         3.0          5.1         1.8  virginica
```

```
SubseTSetosa <- iris[iris$Species == "Setosa",]
SubseTSetosa
```

```
## [1] Sepal.Length Sepal.Width  Petal.Length Petal.Width  Species
## <0 rows> (or 0-length row.names)
```

```
SubseTVersicolor <- iris[iris$Species == "Versicolor",]
SubseTVersicolor
```

```
## [1] Sepal.Length Sepal.Width  Petal.Length Petal.Width  Species
## <0 rows> (or 0-length row.names)
```

```
SubseTVirginica <- iris[iris$Species == "Virginica",]
SubseTVirginica
```

```
## [1] Sepal.Length Sepal.Width  Petal.Length Petal.Width  Species
## <0 rows> (or 0-length row.names)
```

```
tail(SubseTSetosa)
```

```
## [1] Sepal.Length Sepal.Width  Petal.Length Petal.Width  Species
## <0 rows> (or 0-length row.names)
```

```
tail(SubseTVersicolor)
```

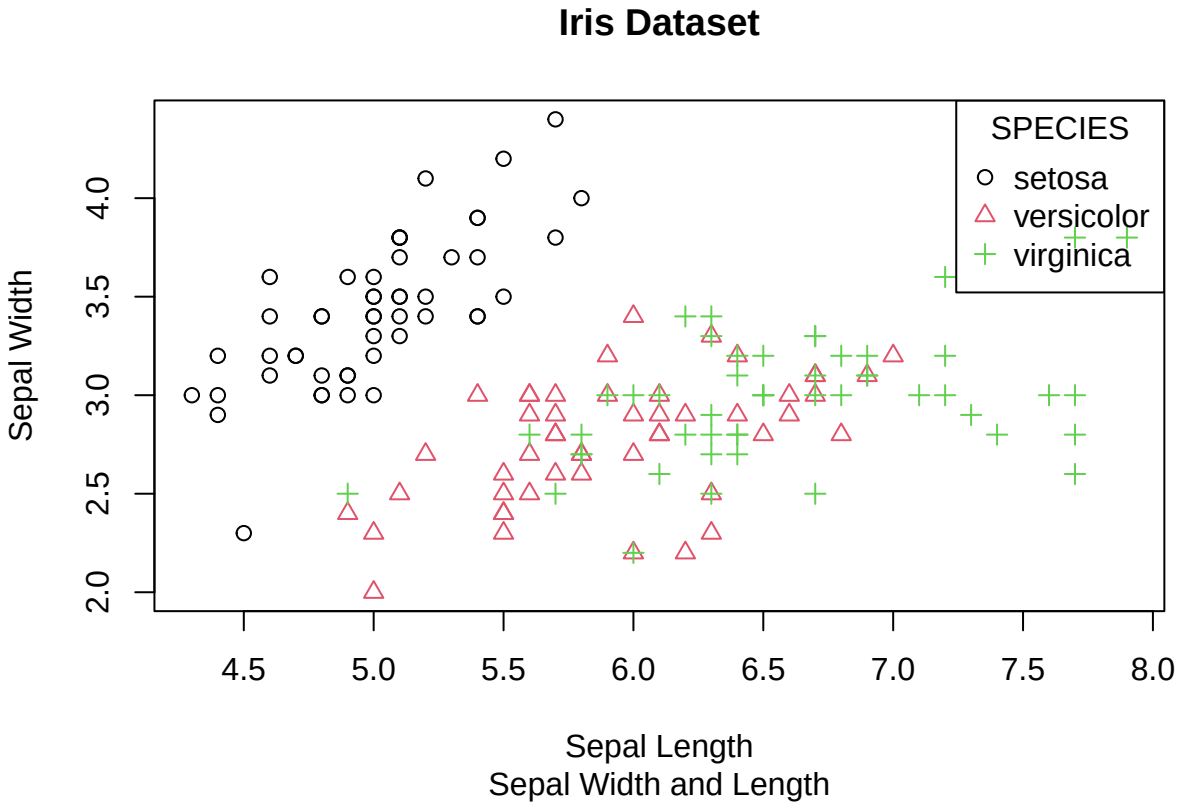
```
## [1] Sepal.Length Sepal.Width  Petal.Length Petal.Width  Species
## <0 rows> (or 0-length row.names)
```

```
tail(SubseTVirginica)
```

```
## [1] Sepal.Length Sepal.Width  Petal.Length Petal.Width  Species
## <0 rows> (or 0-length row.names)
```

E. Create a scatterplot of the sepal.length and sepal.width using the different species(setosa,versicolor,virginica). Add a title = "Iris Dataset", subtitle = "Sepal width and length, labels for the x and y axis, the pch symbol and colors should be based on the species.

```
plot(iris$Sepal.Length, iris$Sepal.Width,
     xlab = "Sepal Length", ylab = "Sepal Width",
     main = "Iris Dataset", sub = "Sepal Width and Length",
     pch = as.numeric(iris$Species), col = as.numeric(iris$Species))

legend("topright", legend = levels(iris$Species),
       pch = 1:3, col = 1:3,
       title = "SPECIES")
```



Hint:

Need to convert to factors the species to store categorical variables.

F. Interpret the result. The Scatterplot enables us to observe the variations in sepal length and width among various species of iris flowers.

The Setosa flowers, characterized by their short sepal length and wide width, are typically found in the upper left part of the plot. The Versicolor flowers have an average sepal length and width, located in the middle part. The Virginica flowers are typically long in sepal length and narrower in width, forming a group in the right part. Based on the plot, the differences between the three Iris species can be easily observed based on the differences in sepal length and width.

7. Import the alexa-file.xlsx. Check on the variations. Notice that there are extra whitespaces among black variants (Black Dot, Black Plus, Black Show, Black Spot). Also on the white variants (White Dot, White Plus, White Show, White Spot).

```
library(readxl)
Alexa_File <- read_excel("Alexa_File.xlsx")
Alexa_File
```

```
## # A tibble: 3,150 x 5
##    rating date                variation           verified_reviews      feedback
##     <dbl> <dttm>              <chr>               <chr>                    <dbl>
##  1      5 2018-07-31 00:00:00 Charcoal Fabric     Love my Echo!                1
##  2      5 2018-07-31 00:00:00 Charcoal Fabric     Loved it!                    1
##  3      4 2018-07-31 00:00:00 Walnut Finish       Sometimes while play~        1
##  4      5 2018-07-31 00:00:00 Charcoal Fabric     I have had a lot of ~        1
##  5      5 2018-07-31 00:00:00 Charcoal Fabric     Music                        1
##  6      5 2018-07-31 00:00:00 Heather Gray Fabric I received the echo ~        1
##  7      3 2018-07-31 00:00:00 Sandstone Fabric    Without having a cel~        1
##  8      5 2018-07-31 00:00:00 Charcoal Fabric     I think this is the ~        1
##  9      5 2018-07-30 00:00:00 Heather Gray Fabric looks great                  1
```

```
## 10      5 2018-07-30 00:00:00 Heather Gray Fabric Love it! I've listen~        1
## # i 3,140 more rows
```

A. Rename the white and black variants by using gsub() function.

```
Alexa_File$variation <- gsub("White  Dot", "WhiteDot", Alexa_File$variation)
Alexa_File$variation <- gsub("White  Plus", "WhitePlus", Alexa_File$variation)
Alexa_File$variation <- gsub("White  Show", "WhiteShow", Alexa_File$variation)
Alexa_File$variation <- gsub("White  Spot", "WhiteSpot", Alexa_File$variation)

Alexa_File$variation <- gsub("Black  Dot", "BlackDot", Alexa_File$variation)
Alexa_File$variation <- gsub("Black  Plus", "BlackPlus", Alexa_File$variation)
Alexa_File$variation <- gsub("Black  Show", "BlackShow", Alexa_File$variation)
Alexa_File$variation <- gsub("Black  Spot", "BlackSpot", Alexa_File$variation)

Alexa_File
```

```
## # A tibble: 3,150 x 5
##    rating date                variation           verified_reviews      feedback
##     <dbl> <dttm>              <chr>               <chr>                    <dbl>
##  1      5 2018-07-31 00:00:00 Charcoal Fabric     Love my Echo!                1
##  2      5 2018-07-31 00:00:00 Charcoal Fabric     Loved it!                    1
##  3      4 2018-07-31 00:00:00 Walnut Finish       Sometimes while play~        1
##  4      5 2018-07-31 00:00:00 Charcoal Fabric     I have had a lot of ~        1
##  5      5 2018-07-31 00:00:00 Charcoal Fabric     Music                        1
##  6      5 2018-07-31 00:00:00 Heather Gray Fabric I received the echo ~        1
##  7      3 2018-07-31 00:00:00 Sandstone Fabric    Without having a cel~        1
##  8      5 2018-07-31 00:00:00 Charcoal Fabric     I think this is the ~        1
##  9      5 2018-07-30 00:00:00 Heather Gray Fabric looks great                  1
## 10      5 2018-07-30 00:00:00 Heather Gray Fabric Love it! I've listen~        1
## # i 3,140 more rows
```

B. Get the total number of each variations and save it into another object. Save the object as variations.RData. Write the R scripts. What is its result?

```
install.packages("dplyr")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'
## (as 'lib' is unspecified)
```

```
library("dplyr")
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
variations_Total <- Alexa_File %>%
  count(Alexa_File$variation)

variations_Total
```

```
## # A tibble: 16 x 2
```

```
##    `Alexa_File$variation`           n
##    <chr>                        <int>
##  1 Black                          261
##  2 BlackDot                       516
##  3 BlackPlus                      270
##  4 BlackShow                      265
##  5 BlackSpot                      241
##  6 Charcoal Fabric                430
##  7 Configuration: Fire TV Stick   350
##  8 Heather Gray Fabric            157
##  9 Oak Finish                      14
## 10 Sandstone Fabric                90
## 11 Walnut Finish                    9
## 12 White                           91
## 13 WhiteDot                       184
## 14 WhitePlus                       78
## 15 WhiteShow                       85
## 16 WhiteSpot                      109
```

```
save(variations_Total, file = "VariaTionS.RData")
```

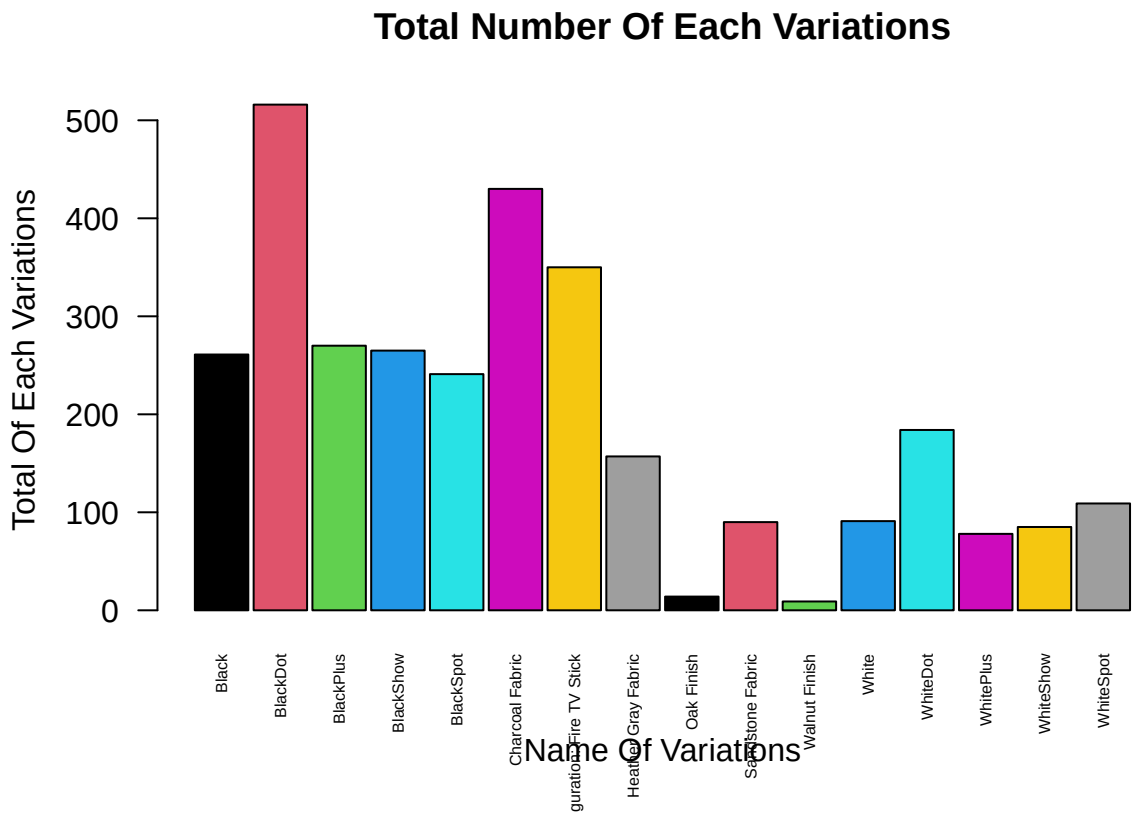
C. From the variations.RData, create a barplot(). Complete the details of the chart which include the title, color, labels of each bar.

```
load ("VariaTionS.RData")
variations_Total
```

```
## # A tibble: 16 x 2
##    `Alexa_File$variation`           n
##    <chr>                        <int>
##  1 Black                          261
##  2 BlackDot                       516
##  3 BlackPlus                      270
##  4 BlackShow                      265
##  5 BlackSpot                      241
##  6 Charcoal Fabric                430
##  7 Configuration: Fire TV Stick   350
##  8 Heather Gray Fabric            157
##  9 Oak Finish                      14
## 10 Sandstone Fabric                90
## 11 Walnut Finish                    9
## 12 White                           91
## 13 WhiteDot                       184
## 14 WhitePlus                       78
## 15 WhiteShow                       85
## 16 WhiteSpot                      109
```

```
varNames <- variations_Total$'Alexa_File$variation'

totalPlot <- barplot(variations_Total$n,
                     names.arg = varNames,
                     main = "Total Number Of Each Variations",
                     xlab = "Name Of Variations",
                     ylab = "Total Of Each Variations",
                     col = 1:16,
                     space = 0.1,
```

```
                    cex.names = 0.5,
                    las = 2 )
```



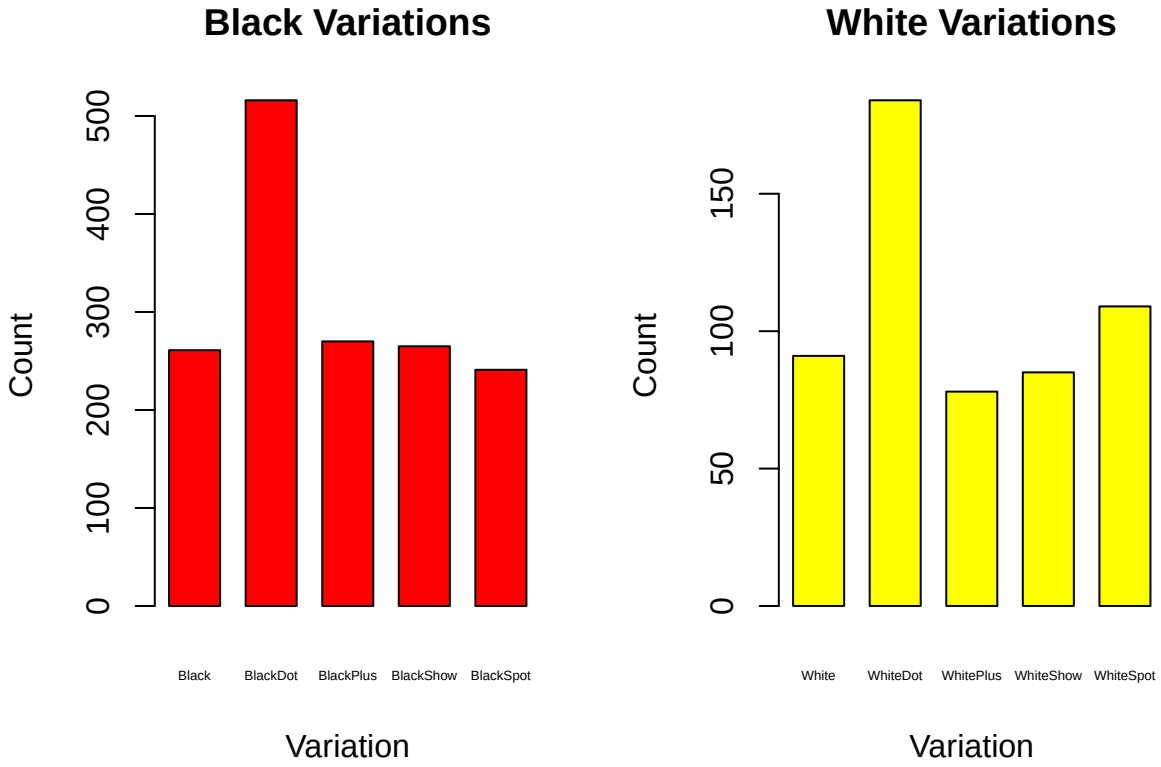
D. Create a barplot() for the black and white variations. Plot it in 1 frame, side by side. Complete the details of the chart.

```
blackVars <- variations_Total[variations_Total$`Alexa_File$variation` %in% c("Black", "BlackPlus" , "Bla

whiteVars <- variations_Total[variations_Total$`Alexa_File$variation` %in% c("White", "WhiteDot", "White

par(mfrow = c(1,2))
blackVars
```

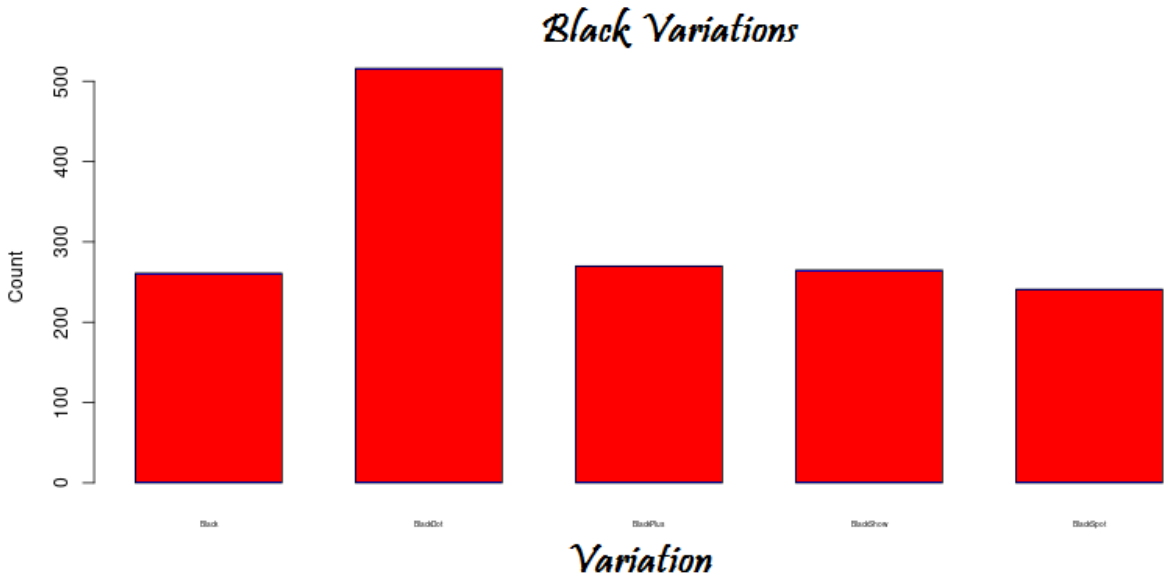
```
## # A tibble: 5 x 2
##   `Alexa_File$variation`     n
##   <chr>                  <int>
## 1 Black                    261
## 2 BlackDot                 516
## 3 BlackPlus                270
## 4 BlackShow                265
## 5 BlackSpot                241
```

```
blackPlot <- barplot(height = blackVars$n,
        names.arg = blackVars$`Alexa_File$variation`,
        col = c("red"),
        main = "Black Variations",
        xlab = "Variation",
        ylab = "Count",
        border = "black",
```

```
        space = 0.5,
        cex.names = 0.4)


whitePlot <- barplot(height = whiteVars$n,
        names.arg = whiteVars$`Alexa_File$variation`,
        col = c("yellow"),
        main = "White Variations",
        xlab = "Variation",
        ylab = "Count",
        border = "black",
        space = 0.5,
        cex.names = 0.4)
```




```
knitr::include_graphics("/cloud/project/RWorksheet#4/RWorksheet#4b/blackVars.png")
```




```
knitr::include_graphics("/cloud/project/RWorksheet#4/RWorksheet#4b/whiteVars.png")
```