RWorksheet_Suero#4b

Jiruel Suero

2023-11-02

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.

```
vectorA <- c(1,2,3,4,5)

matrix_A <- matrix(0, nrow = 5, ncol = 5)

for (j in 1:5)
  for (k in 1:5)
  {
    matrix_A[j, k] <- abs(vectorA[j] - vectorA[k])
  }

matrix_A
```

```
##      [,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 2.

```
for (j in 1:5) {
  cat(paste0("\"", rep("*", j), "\""), "\n")
}
```

```
## "*"
## "*" "*"
## "*" "*" "*"
## "*" "*" "*" "*"
## "*" "*" "*" "*" "*"
```

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
```

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

4a. 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.

```
imported<- read.csv("householdData.csv")
head(imported)
```

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

4b. 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 <- imported[imported$Gender == "M",]
males
```

```
##     X ShoeSize Height Gender
## 5   5     10.5   70.0      M
## 9   9     13.0   72.0      M
## 11 11     10.5   74.5      M
## 13 13     12.0   71.0      M
## 14 14     10.5   71.0      M
## 15 15     13.0   77.0      M
## 16 16     11.5   72.0      M
## 19 19     10.0   72.0      M
## 22 22      8.5   67.0      M
## 23 23     10.5   73.0      M
## 25 25     10.5   72.0      M
## 26 26     11.0   70.0      M
## 27 27      9.0   69.0      M
## 28 28     13.0   70.0      M
```

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

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

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

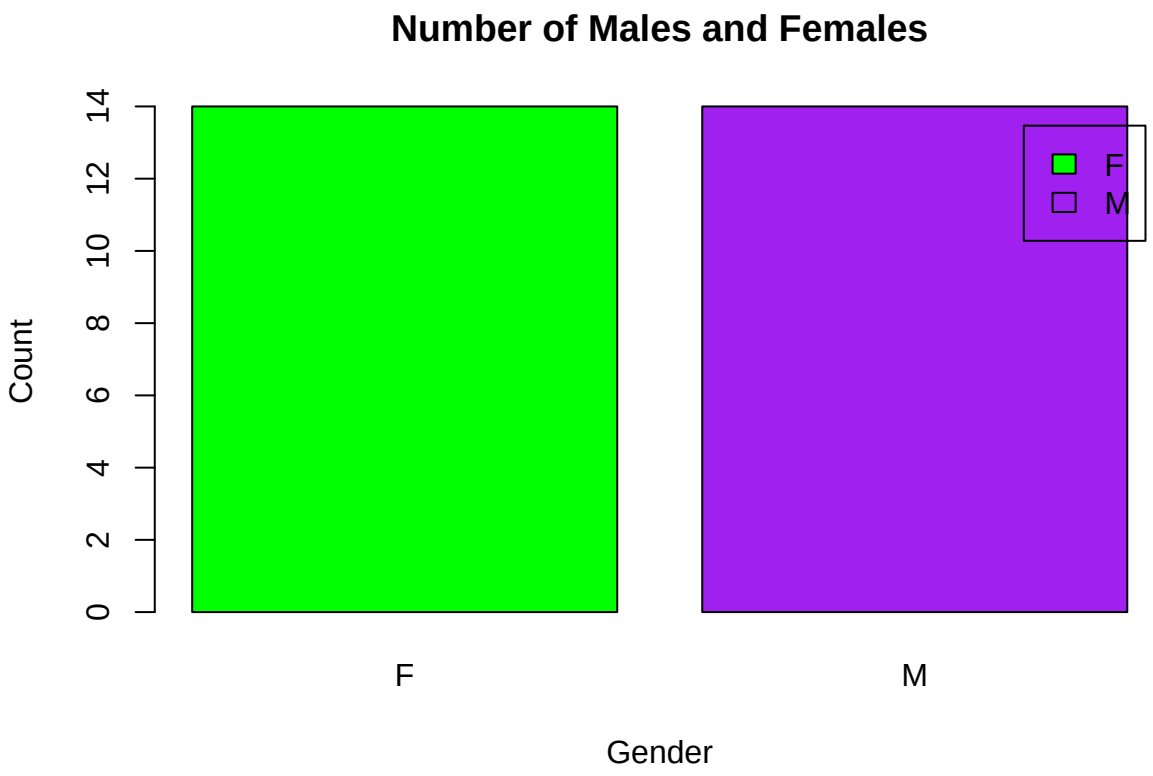
```
## [1] 14
```

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

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

4c. 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

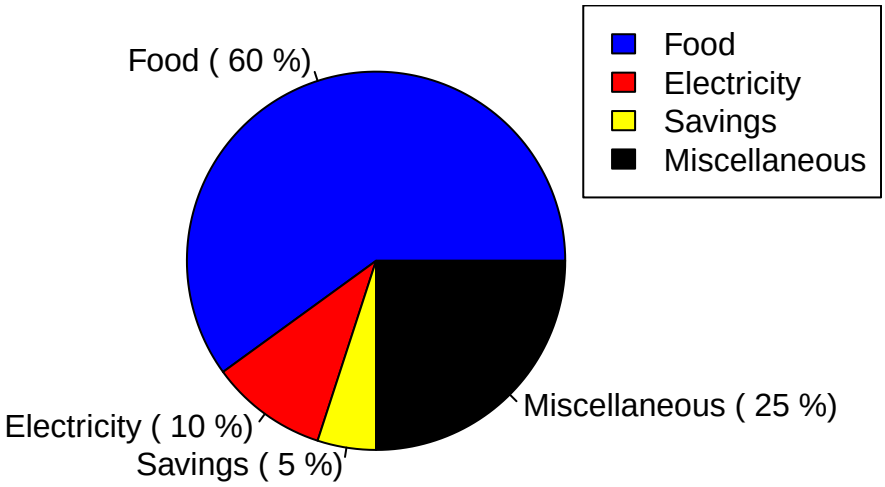
```r
totalMaleFemale <- table(imported$Gender)
barplot(totalMaleFemale,
        main = "Number of Males and Females",
        xlab = "Gender",
        ylab = "Count",
        col = c("green", "purple"),
        legend.text = rownames(totalMaleFemale),
        beside = TRUE)
```



5.The monthly income of Dela Cruz family was spent on the following:

```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("blue", "red", "yellow", "black")

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



6a. Check for the structure of the data set using the str() function. Describe what you have seen in the output.

```
data(iris)
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 comprises details on iris flowers, encompassing measurements of sepal length and width, as
```

6b.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?

```
meanIris <- c(mean(iris$Sepal.Length), mean(iris$Sepal.Width), mean(iris$Petal.Length), mean(iris$Petal
meanIris
```

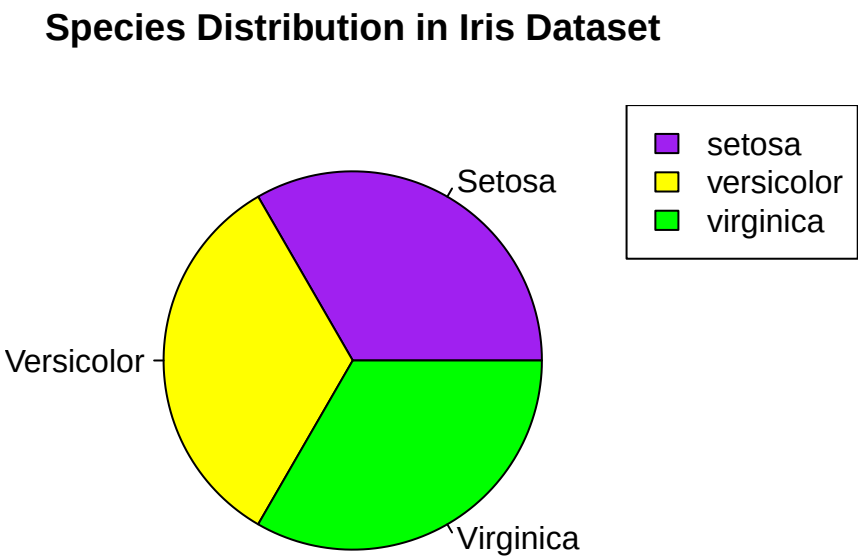
```
## [1] 5.843333 3.057333 3.758000 1.199333
```

```
meanSepalLength <- mean(iris$Sepal.Length)
meanSepalWidth <- mean(iris$Sepal.Width)
meanPetalLength <- mean(iris$Petal.Length)
meanPetalWidth <- mean(iris$Petal.Width)

meanIris <- data.frame(Sepal_Length = meanSepalLength,
                       Sepal_Width = meanSepalWidth,
                       Petal_Length = meanPetalLength,
                       Petal_Width = meanPetalWidth)
meanIris
```

```
##   Sepal_Length Sepal_Width Petal_Length Petal_Width
## 1     5.843333    3.057333        3.758    1.199333
```

6c. Create a pie chart for the Species distribution. Add title, legends, and colors. Write the R script and its result.

```r
species_count <- table(iris$Species)
name_of_Species <- c("Setosa", "Versicolor", "Virginica")
pie(species_count,
    labels = name_of_Species,
    col = c("purple", "yellow", "green"),
    main = "Species Distribution in Iris Dataset")

legend("topright", legend = levels(iris$Species), fill = c("purple", "yellow", "green"),)
```



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

```r
setosaSubset <- subset(iris, Species == "setosa")
versicolorSubset <- subset(iris, Species == "versicolor")
virginicaSubset <- subset(iris, Species == "virginica")


tail(setosaSubset, 6)
```

```
##    Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 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
```

```r
tail(versicolorSubset, 6)
```

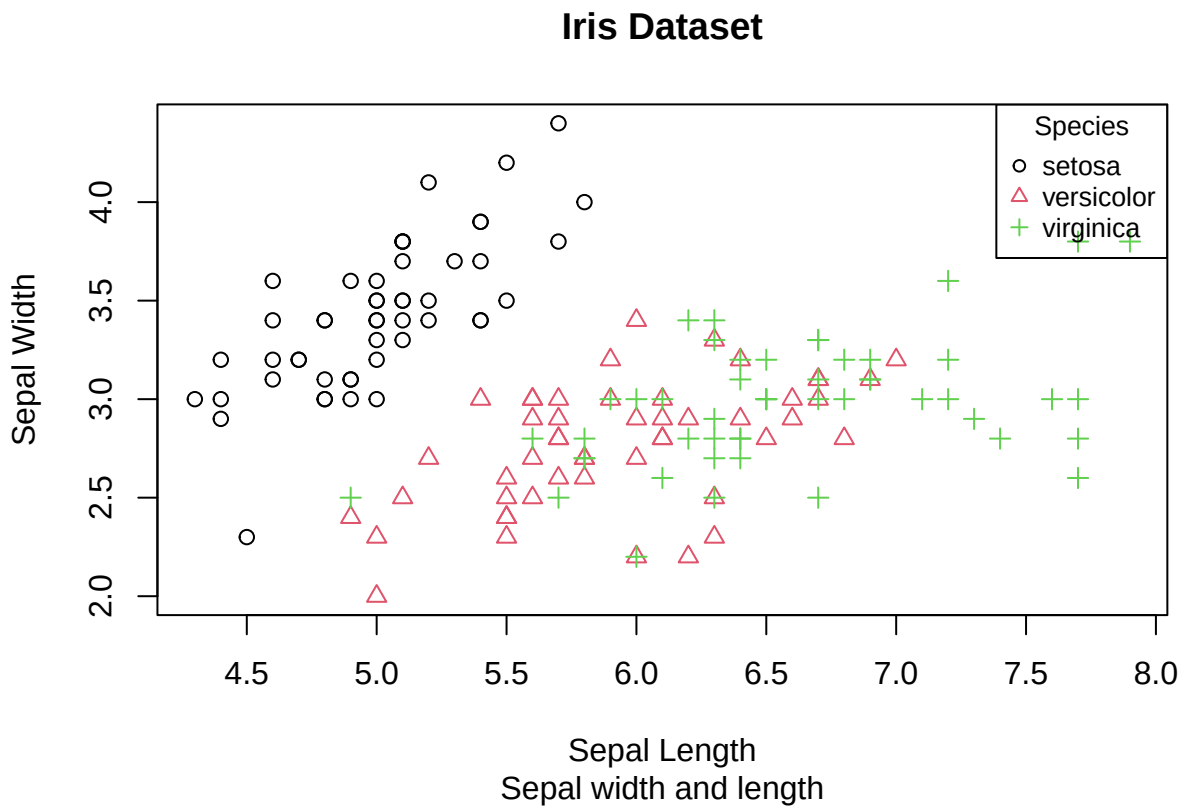
```
##     Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
## 95           5.6         2.7          4.2         1.3 versicolor
## 96           5.7         3.0          4.2         1.2 versicolor
## 97           5.7         2.9          4.2         1.3 versicolor
## 98           6.2         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
```

```
tail(virginicaSubset, 6)
```

```
##     Sepal.Length Sepal.Width Petal.Length Petal.Width   Species
## 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
```

6e. 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), col = 1:3, pch = 1:3, cex = 0.8, title = "Species")
```



6f. Interpret the result.

The dataset is structured as a data frame with 150 rows and five columns. The four numerical variables, Petal.Length, Petal.Width, Sepal.Length, and Sepal.Width, represent measurements of petal and sepal dimensions. Additionally, there's a factor variable called Species, indicating the species of iris flowers with three categories: "setosa," "versicolor," and "virginica." These categories distinguish different types of iris flowers in the dataset.

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<- read_excel("alexa_file.xlsx")
file_path <- "/cloud/project/RWorksheet_Suero#4.xlsx"
alexa
```

```
## # 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
```

7a. Rename the white and black variants by using gsub() function.

```
alexa$variation <- gsub("Black  Dot",
                        "BlackDot",
                        alexa$variation)

alexa$variation <- gsub("Black  Plus",
                        "BlackPlus",
                        alexa$variation)

alexa$variation <- gsub("Black  Show",
                        "BlackShow",
                        alexa$variation)

alexa$variation <- gsub("Black  Spot",
                        "BlackSpot",
                        alexa$variation)

alexa$variation <- gsub("White  Dot",
                        "WhiteDot",
                        alexa$variation)

alexa$variation <- gsub("White  Plus",
                        "WhitePlus",
                        alexa$variation)

alexa$variation <- gsub("White  Show",
                        "WhiteShow",
                        alexa$variation)

alexa$variation <- gsub("White  Spot",
                        "WhiteSpot",
                        alexa$variation)
```

```
alexa
```

```
## # 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
```

7b. 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?

Hint: Use the dplyr package. Make sure to install it before loading the package.

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

```
variationsTotal <- alexa %>%
  count(alexa$variation)

variationsTotal
```

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

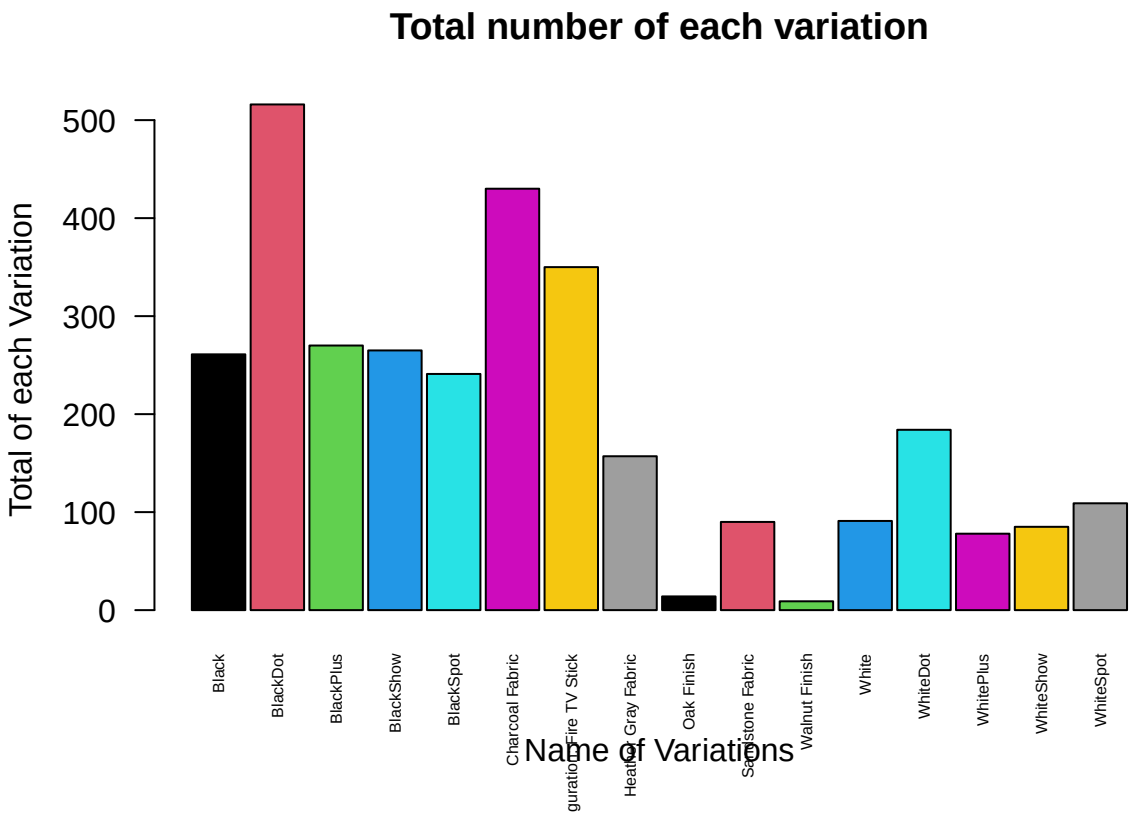
```
save(variationsTotal, file = "variations.RData")
```

7c. From the variations.RData, create a barplot(). Complete the details of the chart which include the title, color, labels of each bar.

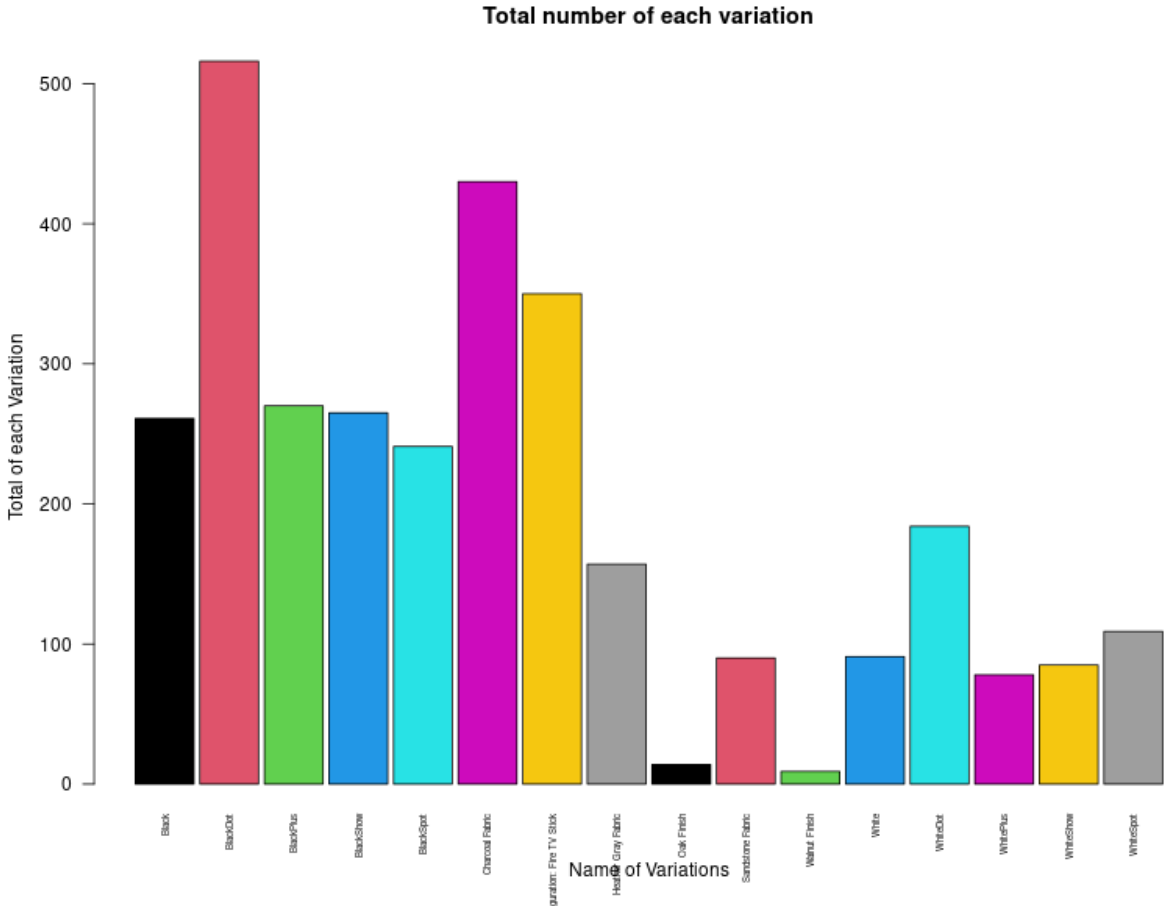
```
load("variations.RData")
variationsTotal
```

```
## # A tibble: 16 x 2
##    `alexa$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 <- variationsTotal$`alexa$variation`

totalPlot <- barplot(variationsTotal$n,
        names.arg = varNames,
        main = "Total number of each variation",
        xlab = "Name of Variations",
        ylab = "Total of each Variation",
        col = 1:16,
        space = 0.1,
        cex.names = 0.5,
        las = 2)
```

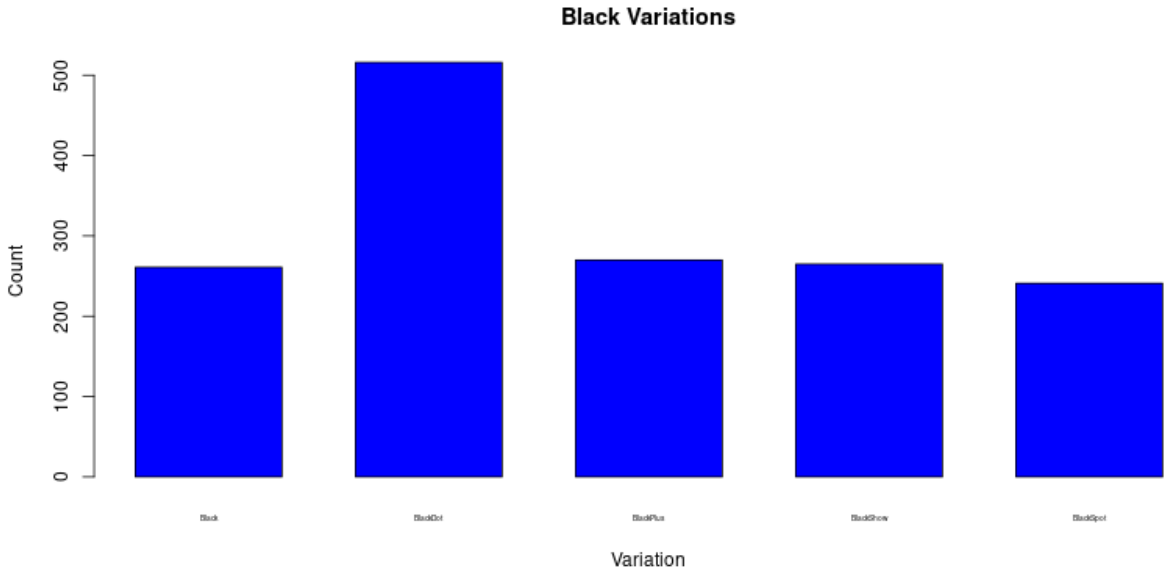



```
png("/cloud/project/RWorksheet_Suero#4/variationsTotal.png", width = 800, height = 600, units = "px", p
knitr::include_graphics("/cloud/project/RWorksheet_Suero#4/variationsTotal.png")
```



7d. Create a barplot() for the black and white variations. Plot it in 1 frame, side by side. Complete the details of the chart.

```
blackVars <- variationsTotal[variationsTotal$`alexa$variation` %in% c("Black", "BlackPlus", "BlackShow"
whiteVars <- variationsTotal[variationsTotal$`alexa$variation` %in% c("White", "WhiteDot", "WhitePlus",

par(mfrow = c(1, 2))

blackPlot <- barplot(height = blackVars$n,
                     names.arg = blackVars$`alexa$variation`,
                     col = c("blue"),
                     main = "Black Variations",
                     xlab = "Variation",
                     ylab = "Count",
                     border = "black",
                     space = 0.5,
                     cex.names = 0.4)
png("/cloud/project/RWorksheet_Suero#4/blackVars.png", width = 800, height = 400, units = "px", pointsi
png("/cloud/project/RWorksheet_Suero#4/whiteVars.png", width = 800, height = 400, units = "px", pointsi


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




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