RWork-sheet_Sabio#4b.Rmd

2023-11-08

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)
matrixA<- matrix( nrow = 5, ncol = 5)

  for (i in 1:5){
    for (j in 1:5){
      matrixA[i, j] <- abs(vectorA[i] - vectorA[j])
    }
  }
print(matrixA)
```

```
##      [,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.

```
n <- 5
  for ( i in 1:n){
    for (j in 1:i){
    cat("*")
  }
cat("\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.

```
fibSequence <- as.integer(readline(prompt = "Enter the number of terms: "))
```

```
## Enter the number of terms:
```

```
a <- 0
b <- 1
cat("Fibonacci Sequence:", a, b)
```

```
## Fibonacci Sequence: 0 1
```

```
repeat {
c <- a + b
if (c > 500) {
break
}
cat(", ",c)
a <- b
b <- c
}
```

```
## ,  1,  2,  3,  5,  8,  13,  21,  34,  55,  89,  144,  233,  377
```

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

```
install.packages("readxl")
```

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

```
library(readxl)
data <- read_excel("Shoe Sizes.xlsx")
```

```
## New names:
## * ‘Height‘ -> ‘Height...2‘
## * ‘Gender‘ -> ‘Gender...3‘
## * ‘Height‘ -> ‘Height...5‘
## * ‘Gender‘ -> ‘Gender...6‘
```

```
data
```

```
## # A tibble: 14 x 6
##    ‘Shoe Size‘ Height...2 Gender...3 ‘Shoe size‘ Height...5 Gender...6
##          <dbl>      <dbl> <chr>                <dbl>      <dbl> <chr>
##  1         6.5       66   F                     13           77 M
##  2         9         68   F                     11.5         72 M
##  3         8.5       64.5 F                      8.5         59 F
##  4         8.5       65   F                      5           62 F
##  5        10.5       70   M                     10           72 M
##  6         7         64   F                      6.5         66 F
##  7         9.5       70   F                      7.5         64 F
##  8         9         71   F                      8.5         67 M
##  9        13         72   M                     10.5         73 M
## 10         7.5       64   F                      8.5         69 F
## 11        10.5       74   M                 10.5         72 M
## 12         8.5       67   F                 11           70 M
## 13        12         71   M                  9           69 M
## 14        10.5       71   M                 13           70 M
```

```r
head(data, n=6)
```

```
## # A tibble: 6 x 6
##   ‘Shoe Size‘ Height...2 Gender...3 ‘Shoe size‘ Height...5 Gender...6
##         <dbl>      <dbl> <chr>            <dbl>      <dbl> <chr>
## 1         6.5       66   F                 13           77 M
## 2         9         68   F                 11.5         72 M
## 3         8.5       64.5 F                  8.5         59 F
## 4         8.5       65   F                  5           62 F
## 5        10.5       70   M                 10           72 M
## 6         7         64   F                  6.5         66 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.

```r
maleSubset <- subset(data, Gender...3  == "M")
femaleSubset <- subset(data, Gender...3  == "F")

Male <- nrow(maleSubset)
Female <- nrow(femaleSubset)

cat("Number of observations of Male in Gender...3 is: ", Male, "\n")
```

```
## Number of observations of Male in Gender...3 is:  5
```

```r
cat("Number of observations of Female in Gender...3 is: ", Female, "\n")
```

```
## Number of observations of Female in Gender...3 is:  9
```

```r
maleSubset1 <- subset(data, Gender...6  == "M")
femaleSubset1 <- subset(data, Gender...6  == "F")

Male1 <- nrow(maleSubset1)
Female1 <- nrow(femaleSubset1)

cat("Number of observations of Male in Gender...6 is: ", Male1, "\n")
```

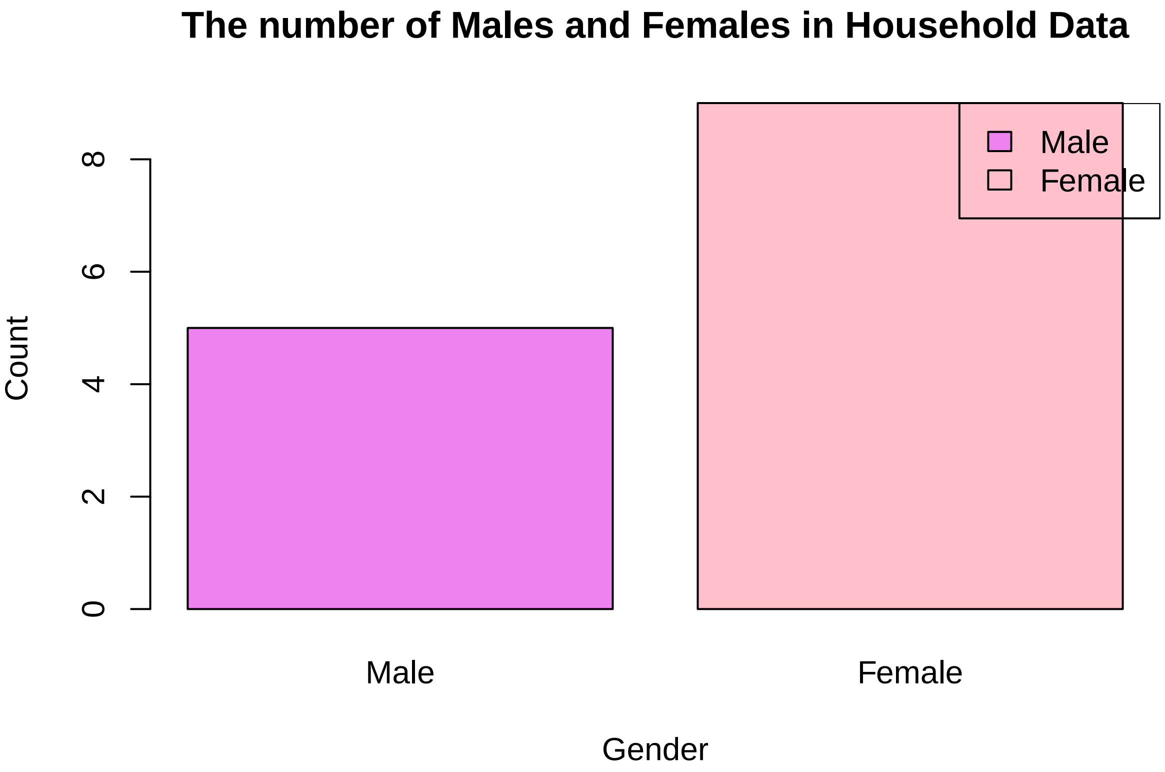
```
## Number of observations of Male in Gender...6 is:  9
```

```r
cat("Number of observations of Female in Gender...6 is: ", Female1, "\n")
```

```
## Number of observations of Female in Gender...6 is:  5
```

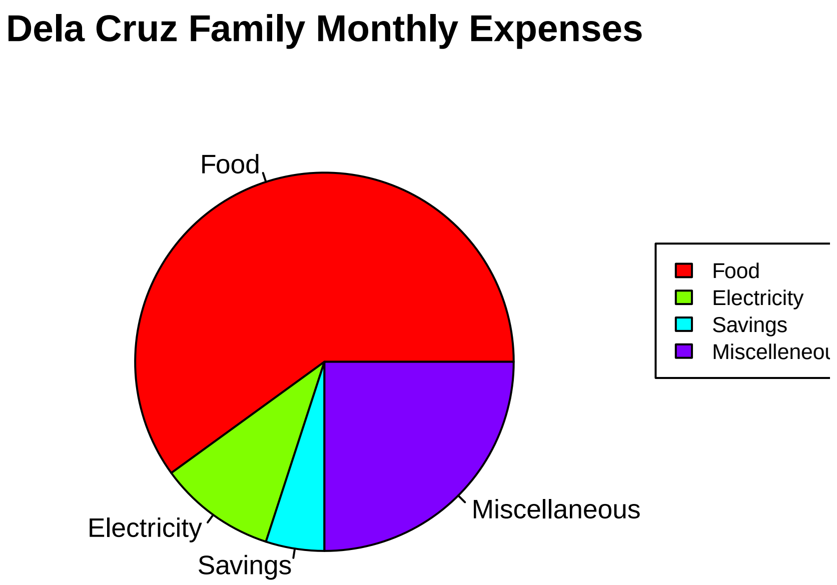
c. Create a graph for the number of males and females for Household Data. Use plot(), chart type = barplot. Make sure to place title, legends, and colors. Write the R scripts and its result.

```r
count <- c(Male, Female)
gender <- c("Male", "Female")
barplot(count,
names.arg = gender,
main = "The number of Males and Females in Household Data",
xlab = "Gender",
ylab = "Count",
col = c("violet", "pink"))
legend("topright",
legend = gender,
fill = c("violet", "pink"))
```



5a. 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
expenses <- c(60, 10, 5, 25)
categories <- c("Food", "Electricity", "Savings", "Miscellaneous")

percentages <- round((expenses / sum(expenses)) * 100, 1)

pie(expenses, main="Dela Cruz Family Monthly Expenses", col = rainbow(length(expenses)),
    labels = categories, cex = 0.8)
legend (1.4,0.5,
        c("Food","Electricity","Savings","Miscelleneous"), cex = 0.7, fill =rainbow(length(expenses)))
```



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

```r
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 ...
```

```r
#It is a data frame that has 150 observation of 5 variables which is sepal.length, sepal.width, petal.l
```

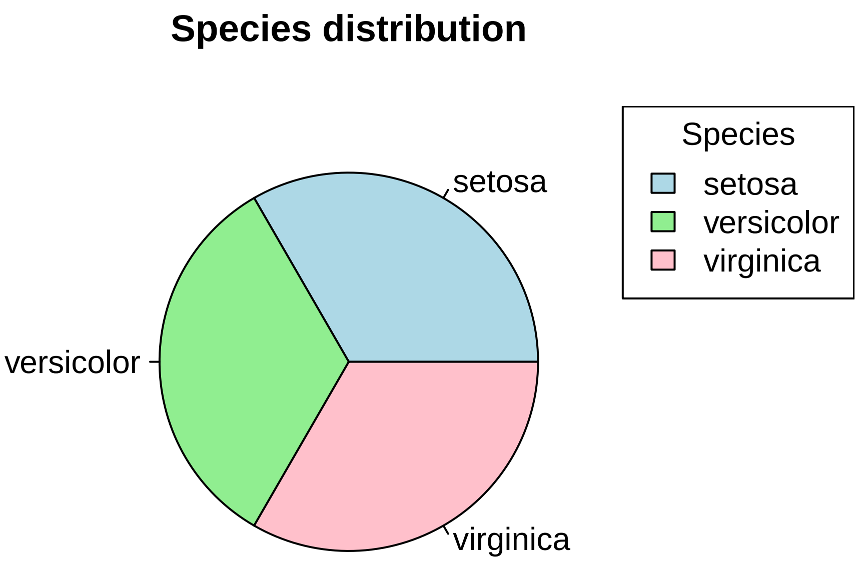
b. Create an R object that will contain the mean of the sepal.length, sepal.width,petal.length,and petal.width. What is the R script and its result?

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

```
## [1] 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.

```r
pie(table(iris$Species),
main = "Species distribution",
labels = levels(iris$Species),
col = c("lightblue","lightgreen","pink"))
legend("topright", legend = levels(iris$Species), fill = c("lightblue","lightgreen","pink") , title = "
```



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

```
subsetSetosa <- tail(subset(iris, Species == "setosa"), 6)
subsetSetosa
```

```
##    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
```

```
subsetVersicolor <- tail(subset(iris, Species == "versicolor"), 6)
subsetVersicolor
```

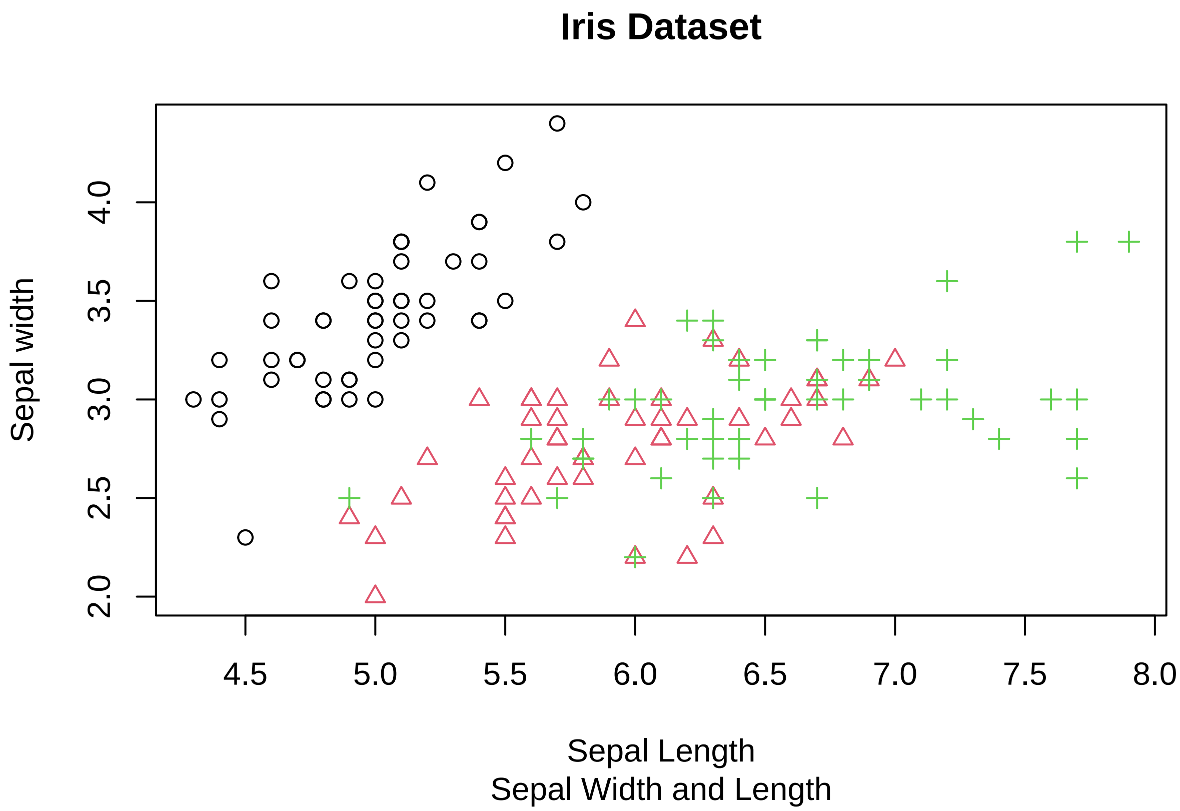
```
##     Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
## 95           5.6         2.7          4.2         1.3 versicolor
## 96           5.7         3.0          4.2         1.2 versicolor
## 97           5.7         2.9          4.2         1.3 versicolor
## 98           6.2     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

```
subsetVirginica <- tail(subset(iris, Species == "virginica"), 6)
subsetVirginica
```

```
##     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
```

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.

```
iris$Species<- as.factor(iris$Species)
plot(iris$Sepal.Length,iris$Sepal.Width,pch = as.integer(iris$Species),col = as.integer(iris$Species), m
```



```
as.factor(iris$Species)
```

```
##   [1] setosa     setosa     setosa     setosa     setosa     setosa
##   [7] setosa     setosa     setosa     setosa     setosa     setosa
##  [13] setosa     setosa     setosa     setosa     setosa     setosa
##  [19] setosa     setosa     setosa     setosa     setosa     setosa
##  [25] setosa     setosa     setosa     setosa     setosa     setosa
##  [31] setosa     setosa     setosa     setosa     setosa     setosa
##  [37] setosa     setosa     setosa     setosa     setosa     setosa
##  [43] setosa     setosa     setosa     setosa     setosa     setosa
##  [49] setosa     setosa     versicolor versicolor versicolor versicolor
##  [55] versicolor versicolor versicolor versicolor versicolor versicolor
##  [61] versicolor versicolor versicolor versicolor versicolor versicolor
##  [67] versicolor versicolor versicolor versicolor versicolor versicolor
##  [73] versicolor versicolor versicolor versicolor versicolor versicolor
##  [79] versicolor versicolor versicolor versicolor versicolor versicolor
##  [85] versicolor versicolor versicolor versicolor versicolor versicolor
##  [91] versicolor versicolor versicolor versicolor versicolor versicolor
##  [97] versicolor versicolor versicolor versicolor virginica  virginica
## [103] virginica  virginica  virginica  virginica  virginica  virginica
## [109] virginica  virginica  virginica  virginica  virginica  virginica
## [115] virginica  virginica  virginica  virginica  virginica  virginica
## [121] virginica  virginica  virginica  virginica  virginica  virginica
## [127] virginica  virginica  virginica  virginica  virginica  virginica
## [133] virginica  virginica  virginica  virginica  virginica  virginica
## [139] virginica  virginica  virginica  virginica  virginica  virginica
## [145] virginica  virginica  virginica  virginica  virginica  virginica
## Levels: setosa versicolor virginica
```

```
## [1] setosa setosa setosa setosa setosa setosa
## [7] setosa setosa setosa setosa setosa setosa
## [13] setosa setosa setosa setosa setosa setosa
## [19] setosa setosa setosa setosa setosa setosa
## [25] setosa setosa setosa setosa setosa setosa
## [31] setosa setosa setosa setosa setosa setosa
## [37] setosa setosa setosa setosa setosa setosa
## [43] setosa setosa setosa setosa setosa setosa
## [49] setosa setosa versicolor versicolor versicolor versicolor
## [55] versicolor versicolor versicolor versicolor versicolor versicolor
## [61] versicolor versicolor versicolor versicolor versicolor versicolor
## [67] versicolor versicolor versicolor versicolor versicolor versicolor
## [73] versicolor versicolor versicolor versicolor versicolor versicolor
## [79] versicolor versicolor versicolor versicolor versicolor versicolor
## [85] versicolor versicolor versicolor versicolor versicolor versicolor
## [91] versicolor versicolor versicolor versicolor versicolor versicolor
## [97] versicolor versicolor versicolor versicolor virginica virginica
## [103] virginica virginica virginica virginica virginica virginica
## [109] virginica virginica virginica virginica virginica virginica
## [115] virginica virginica virginica virginica virginica virginica
## [121] virginica virginica virginica virginica virginica virginica
## [127] virginica virginica virginica virginica virginica virginica
## [133] virginica virginica virginica virginica virginica virginica
## [139] virginica virginica virginica virginica virginica virginica
## [145] virginica virginica virginica virginica virginica virginica
## Levels: setosa versicolor virginica
```

f. Interpret the result.

```
#the scatterplot result represents the relationship between sepal length and width for different specie
```

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

```
install.packages("readxl")
```

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

```
library(readxl)
alexaFile <- read_excel("alexa_file.xlsx")

alexaVar <- gsub("Black Plus", "Black Plus", alexaFile$variation)
alexaFile$variation <- gsub("Black Show", "Black Show", alexaFile$variation)
```

```
alexaFile$variation <- gsub("Black Spot", "Black Spot", alexaFile$variation)
alexaFile$variation <- gsub("Black Dot", "Black Dot", alexaFile$variation)
alexaFile$variation <- gsub("White Dot", "White Dot", alexaFile$variation)
alexaFile$variation <- gsub("White Plus", "White Plus", alexaFile$variation)
alexaFile$variation <- gsub("White Show", "White Show", alexaFile$variation)
alexaFile$variation <- gsub("White Spot", "White Spot", alexaFile$variation)
```

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?

```
library(dplyr)
```

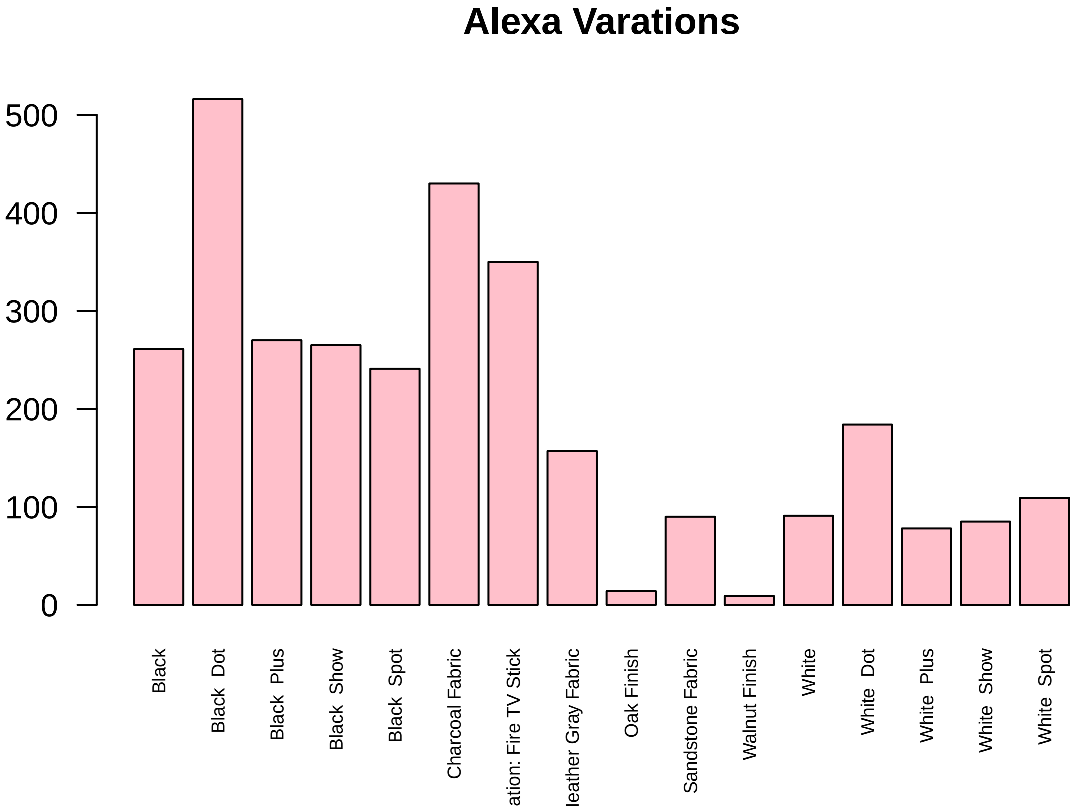
```
##
## Attaching package: ’dplyr’

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

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

```
save(alexaFile, file = "variations.RData")
load("variations.RData")
alexaVar <- alexaFile%>%count(alexaFile$variation)
alexaVar
```

```
## # A tibble: 16 x 2
##    ‘alexaFile$variation‘            n
##    <chr>                        <int>
##  1 Black                          261
##  2 Black  Dot                     516
##  3 Black  Plus                    270
##  4 Black  Show                    265
##  5 Black  Spot                    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 White  Dot                     184
## 14 White  Plus                     78
## 15 White  Show                     85
## 16 White  Spot                    109
```

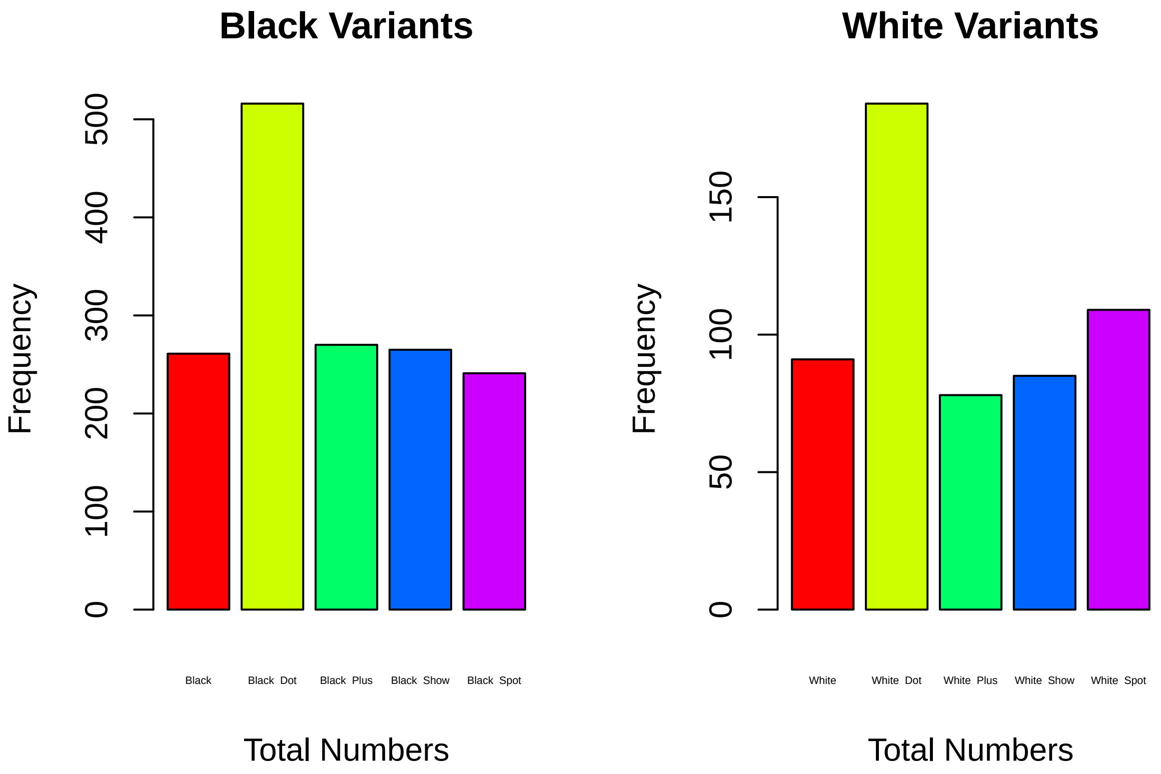
7c.From the variations.RData, create a barplot(). Complete the details of the chartwhich include the title, color, labels of each bar.

```
barplot(
height = alexaVar$n,
names.arg = alexaVar$`alexaFile$variation`,
col = "pink",
main = "Alexa Varations",
las = 2,
cex.names = 0.58
)
```



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

```
par(mfrow = c(1, 2))
blackVariants <- alexaVar[1:5,]
whiteVariants <- alexaVar[12:16,]
barplot(
height = blackVariants$n,
names.arg = blackVariants$`alexaFile$variation`,
main = "Black Variants",
col = rainbow(5),
xlab = 'Total Numbers',
ylab = 'Frequency',
cex.names = 0.35,
)
barplot(
height = whiteVariants$n,
names.arg = whiteVariants$`alexaFile$variation`,
main = "White Variants",
col = rainbow(5),
xlab = 'Total Numbers',
ylab = 'Frequency',
cex.names = 0.35,
)
```


Black Variants
Frequency
Black
Black Dot
Black Plus
Black Show
Black Spot
Total Numbers
White Variants
Frequency
White
White Dot
White Plus
White Show
White Spot
Total Numbers