RWorkSheet_Matias#4b

2023-11-09

1.Using the for loop

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

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

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

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.

```
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.

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

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

```
if (is.na(Input) || Input < 0) {
  cat("Please enter a valid non-negative number.")
} else {
  x <- Input
  y <- 0

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

  repeat {
```

```
    next_num <- x + y

    if (next_num > 500) {
      break
    }

    cat(next_num, " ")
    x <- y
    y <- next_num
  }
}
```

```
## Please enter a valid non-negative number.
```

4. What is the R script for importing an excel or a csv file?

```
importData <- read.csv("/cloud/project/HOUSEHOLD DATA.csv")
head(importData)
```

```
##   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
```

4b. Create a subset for gender(female and male). How many observations are there in Male? How about in Female?

```
male <- importData[importData$Gender == "M",]
male
```

```
##    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
```

```
female <- importData[importData$Gender == "F",]
female
```

```
##    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
```

```r
numMale <- nrow(male)
numMale
```

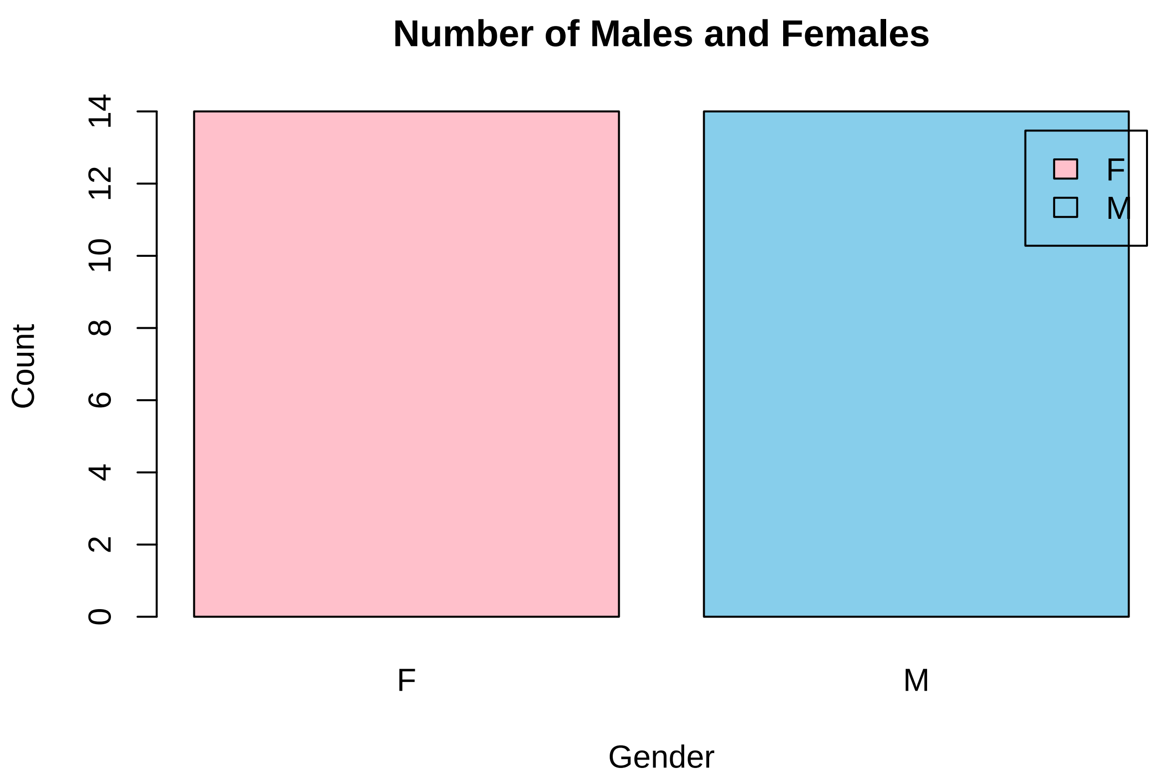
```
## [1] 14
```

```r
numFem <- nrow(female)
numFem
```

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

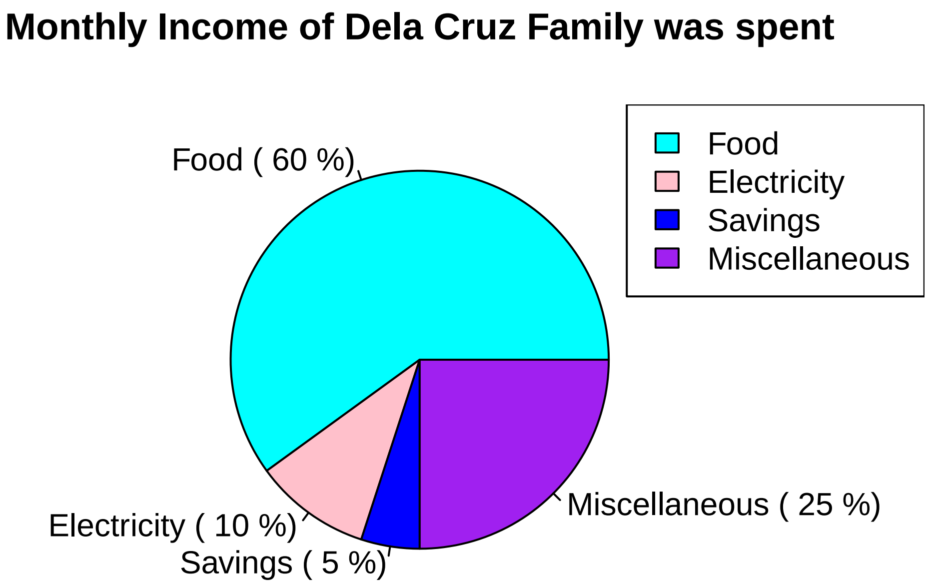
4c. Create a graph for the number of males and females for Household Data. Use plot(), chart type = barplot. Make sure to place title, legends, and colors.

```r
totalMaleFemale <- table(importData$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:

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

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

colors <- c("cyan", "pink", "blue", "purple")

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

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



6a. Check for the structure of the data set using the str() function

```
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 ...
```

6b.Create an R object that will contain the mean of the sepal.length, sepal.width,petal.length,and petal.width.

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

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

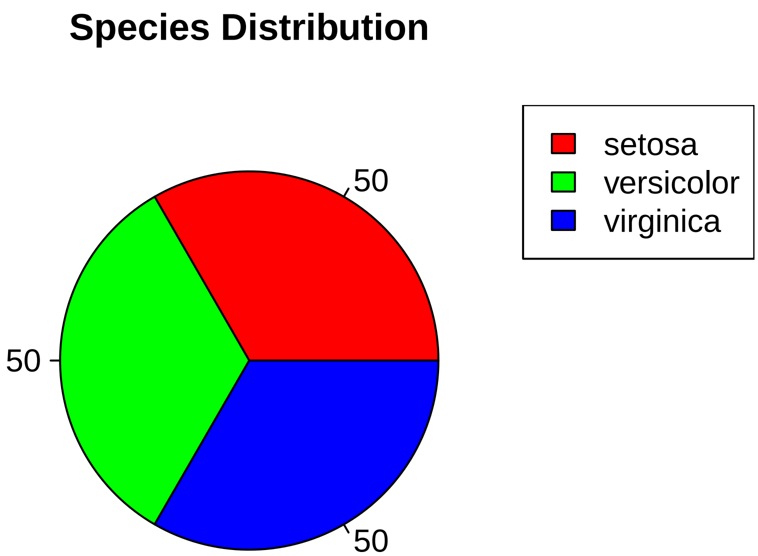
```
meanSepal_Length <- mean(iris$Sepal.Length)
meanSepal_Width <- mean(iris$Sepal.Width)
```

```
meanPetal_Length <- mean(iris$Petal.Length)
meanPetal_Width <- mean(iris$Petal.Width)

mean_Iris <- data.frame(Sepal_Length = meanSepal_Length,
                        Sepal_Width = meanSepal_Width,
                        Petal_Length = meanPetal_Length,
                        Petal_Width = meanPetal_Width)
mean_Iris
```

```
##   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.

```
Species <- table(iris$Species)
pie(Species, labels = Species, col = rainbow(length(Species)), main = "Species Distribution")
legend("topright", names(Species), cex = 1.0 , fill = rainbow(length(Species)))
```



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

```
setosa<- subset(iris, Species == "setosa")
versicolor <- subset(iris, Species == "versicolor")
virginica <- subset(iris, Species == "virginica")

tail(setosa, 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
```

```
tail(versicolor, 6)
```

```
##     Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
## 95           5.6         2.7          4.2         1.3 versicolor
```

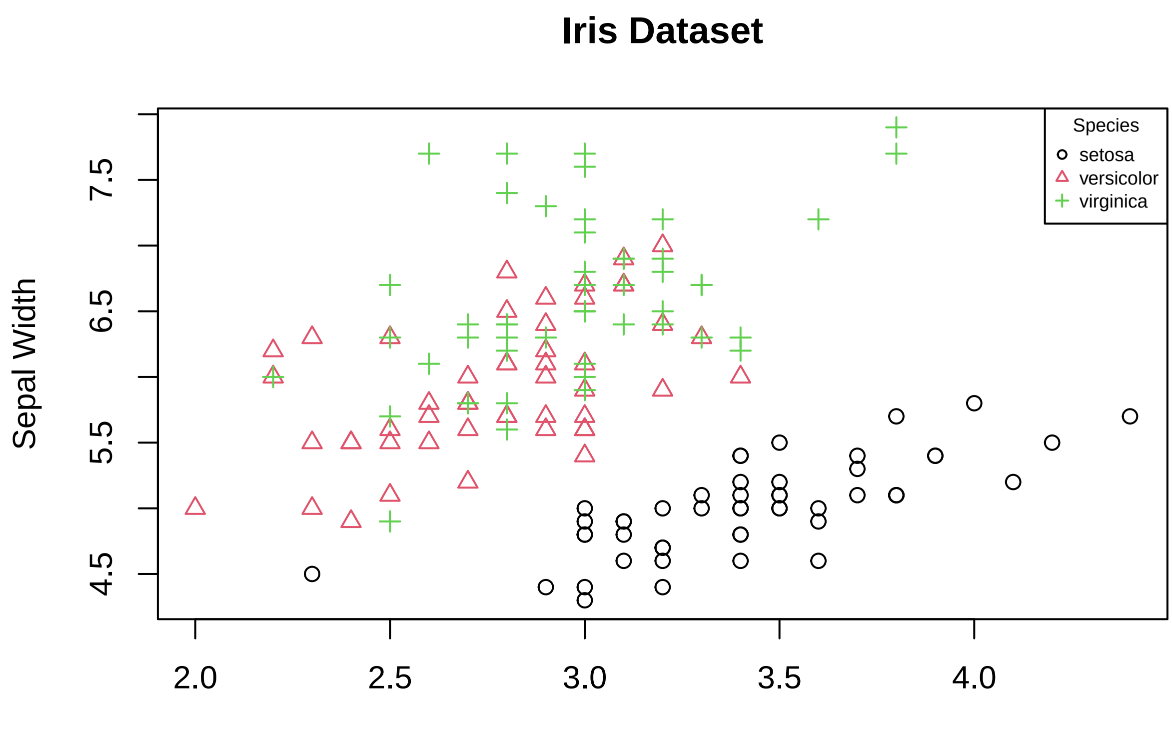
```
## 96           5.7         3.0          4.2         1.2 versicolor
## 97           5.7         2.9          4.2         1.3 versicolor
## 98           6.2         2.9          4.3         1.3 versicolor
## 99           5.1         2.5          3.0         1.1 versicolor
## 100          5.7         2.8          4.1         1.3 versicolor
```

```r
tail(virginica, 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.

```r
iris$Species <- as.factor(iris$Species)

# Convert the "Species" column to a factor
iris$Species <- as.factor(iris$Species)

# Create a scatterplot
plot(
  Sepal.Length ~ Sepal.Width,
  data = iris,
  pch = as.integer(iris$Species),
  # Use different pch symbols for each species
  col = as.integer(iris$Species),
  # Use different colors for each species
  xlab = "Sepal Length",
  ylab = "Sepal Width",
  main = "Iris Dataset",
  sub = "Sepal width and length"
)

# Add a legend
legend("topright", legend = levels(iris$Species), col = 1:3, pch = 1:3, cex = 0.6, title = "Species")
```



Sepal width and length

6f. Interpret the result.

The relation between sepal length and sepal width for each species in the Iris dataset is visually shown by the scatter plot. Every point represents a single observation, and species-specific colors and plotting traits are used to distinguish between the points. The tale offers an explanation of the hues and symbols connected to each species.

7.Import the alexa-file.xlsx. Check on the variations. Notice that there are ex-tra 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")
```

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

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

```r
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
```

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.

```r
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
```

```r
variation_total <- alexa_file %>%
  count(alexa_file$variation)

variation_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
```

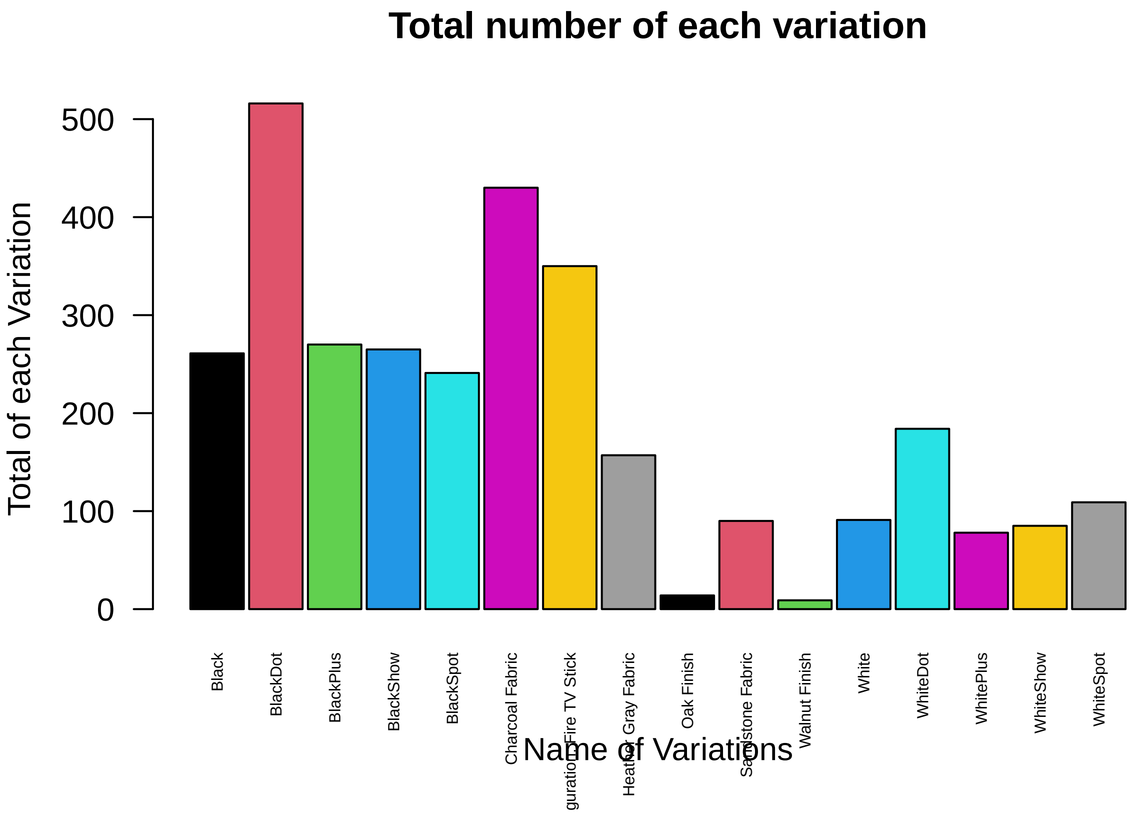
```r
save(variation_total, file = "variations.RData")
```

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

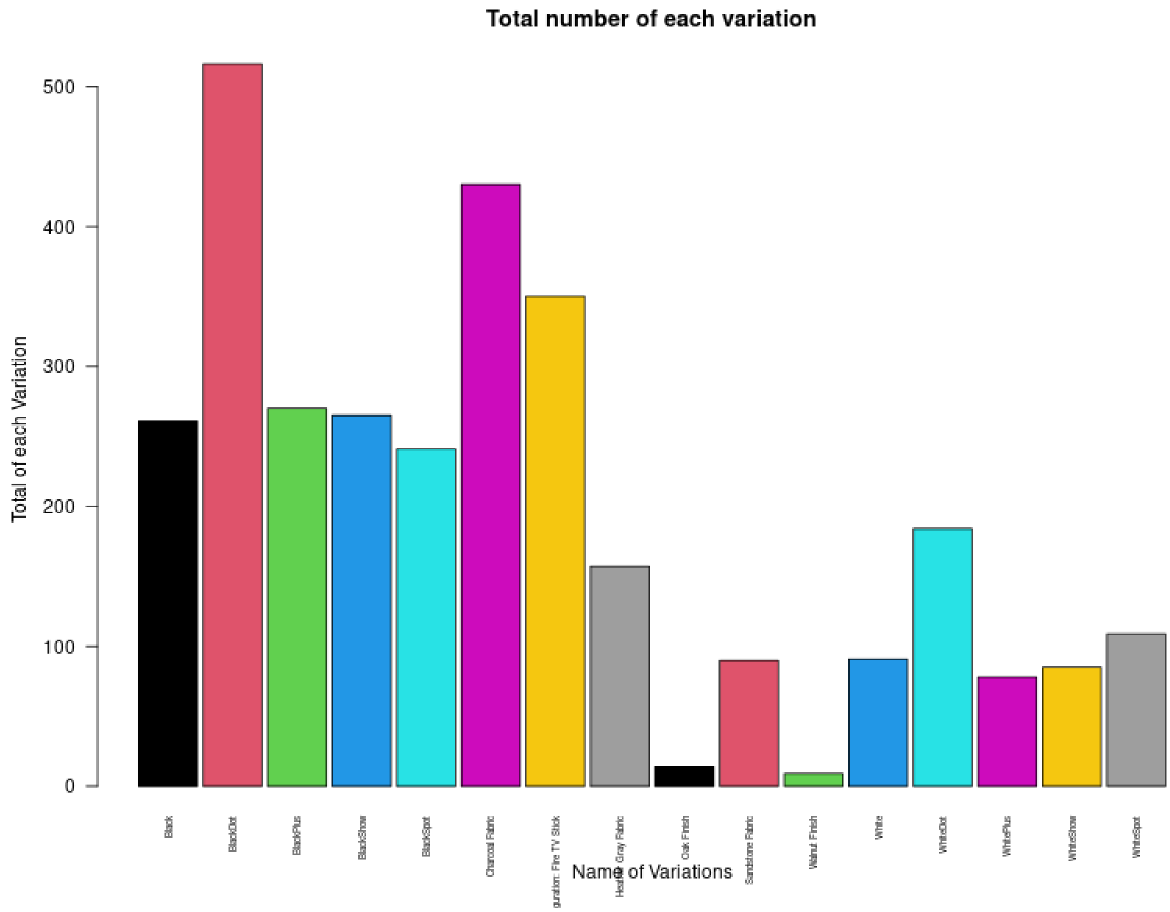
```r
load("variations.RData")

variation_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
```

```r
variation_names <- variation_total$`alexa_file$variation`

total_Plot <- barplot(variation_total$n,
        names.arg = variation_names,
        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)
```




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



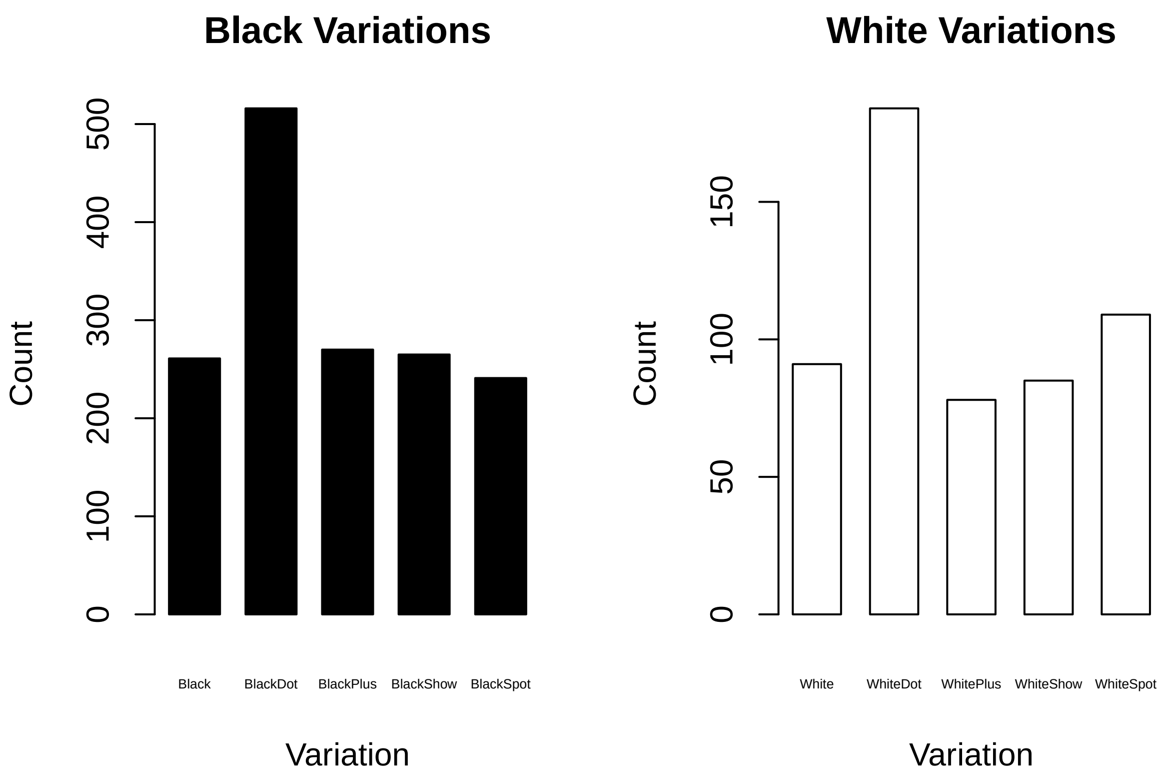
7d. Create a barplot() for the black and white variations. Plot it in 1 frame, side by side. Complete the details of the chart.

```
blackvariation <- variation_total[variation_total$`alexa_file$variation` %in% c("Black", "BlackPlus" ,

whitevariation <- variation_total[variation_total$`alexa_file$variation` %in% c("White", "WhiteDot", "Wh

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

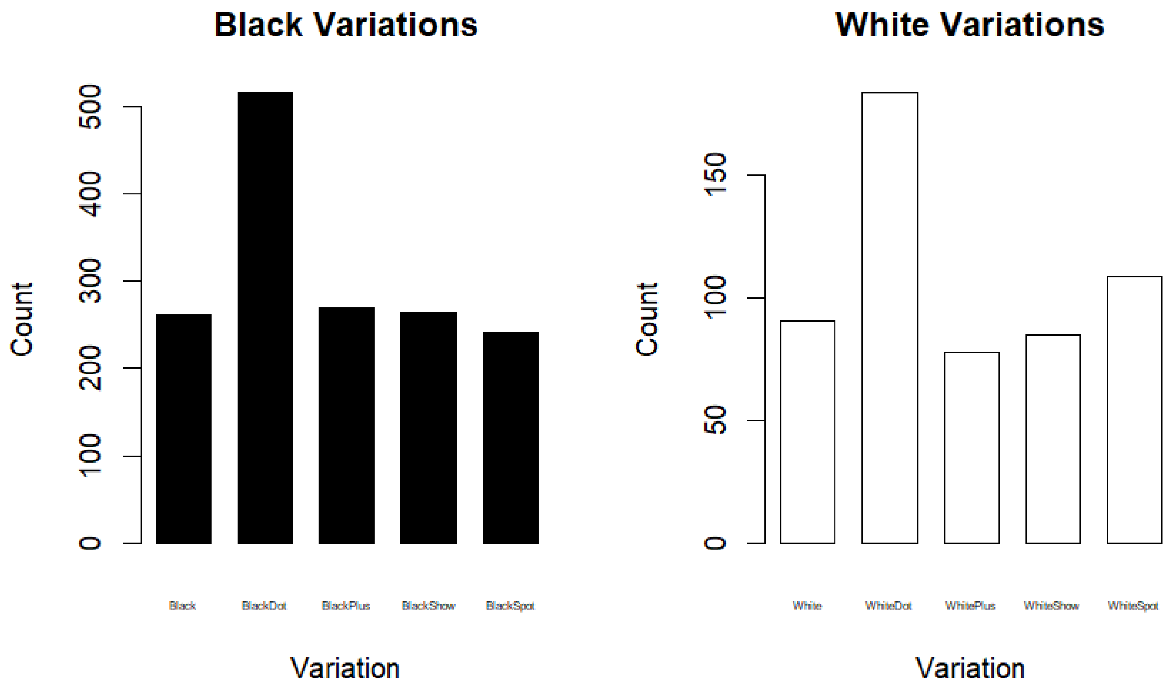
```
## # A tibble: 5 x 2
##   `alexa_file$variation`     n
##   <chr>                  <int>
## 1 Black                    261
## 2 BlackDot                 516
## 3 BlackPlus                270
## 4 BlackShow                265
## 5 BlackSpot                241
```

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

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


whitePlot <- barplot(height = whitevariation$n,
        names.arg = whitevariation$`alexa_file$variation`,
        col = c("white"),
        main = "White Variations",
        xlab = "Variation",
        ylab = "Count",
        border = "black",
        space = 0.6,
        cex.names = 0.4)
```




```
knitr::include_graphics("/cloud/project/RWorksheet_Matias#4/blackNwhiteVars.png")
```


Black Variations
Count
0
100
200
300
400
500
Black
BlackDot
BlackPlus
BlackShow
BlackSpot
Variation
White Variations
Count
0
50
100
150
White
WhiteDot
WhitePlus
WhiteShow
WhiteSpot
Variation