7. Create a factor marital_status with levels Married, single, divorced. Perform the following operations on this factor.

a. Check the variable is a factor

b. Access the 2nd and 4th element in the factor

c. Remove third element from the factor

d. Modify the second element of the factor

e. Add a new level widowed to the factor and add the same level to the factor marital_status.

```
marital_statuses <- factor(c("Married", "Single", "Divorced", "Married", "Single", "Divorced"))
print(marital_statuses)

cat("\na. Check the variable is a factor\n")
is_factor <- is.factor(marital_statuses)
print(is_factor)

cat("\nb. Access the 2nd and 4th element in the factor\n")
second_element <- marital_statuses[c(2, 4)]
print(second_element)

cat("\nc. Remove third element from the factor\n")
removed_factor <- marital_statuses[-3]
print(removed_factor)

cat("\nd. Modify the second element of the factor\n")
marital_statuses[2] <- "Married"
print(marital_statuses)
```

```
cat("\ne. Add new level widowed to the factor and add the same level to the factor marital_status \n")
marital_statuses <- factor(marital_statuses, levels = c(levels(marital_statuses), "Widowed"))
print(marital_statuses)
marital_statuses <- factor(c(as.character(marital_statuses), "Widowed"))
print(marital_statuses)
```

Output:

```
[1] Married Single Divorced Married Single Divorced
Levels: Divorced Married Single
a. Check the variable is a factor
[1] TRUE

b. Access the 2nd and 4th element in the factor
[1] Single Married
Levels: Divorced Married Single

c. Remove third element from the factor
[1] Married Single Married Single Divorced
Levels: Divorced Married Single

d. Modify the second element of the factor
[1] Married Married Divorced Married Single Divorced
Levels: Divorced Married Single



e. Add new level widowed to the factor and add the same level to the factor marital_status
```

```
[1] Married Married Divorced Married Single Divorced
Levels: Divorced Married Single Widowed
[1] Married Married Divorced Married Single Divorced Widowed
Levels: Divorced Married Single Widowed
```

8. Write a R language Script for following operation on Iris Data Set

1. Load the Iris Dataset

2. View first six rows of iris dataset

3. Summarize iris dataset

4. Display number of rows and columns

5. Display column names of dataset.

6. Create histogram of values for sepal length

7. Create scatterplot of sepal width vs. sepal length

8. Create boxplot of sepal width vs. sepal length

9. Find Pearson correlation between Sepal.Length and Petal.Length

10. Create correlation matrix for dataset.

```
cat("Write a R language Script for the following operation on Iris Data Set.\n")
cat("\n1. Load the Iris Dataset")
print(data(iris))

cat("\n2. View the first six rows of the iris dataset.\n")
```

```
print(head(iris))

cat("\n3. Summarize iris dataset")
summary(iris)

cat("\n4. Display number of rows and columns.\n")
Dim_row_col <- dim(iris)
print(Dim_row_col)

cat("\n5. Display column names of dataset.\n")
Names_iris <- names(iris)
print(Names_iris)

cat("\n6. Create a histogram of values for sepal length.\n")
hist(iris$Sepal.Length,
col = "steelblue",
main = "Histogram",
xlab = "Length",
ylab = "Frequency"
)

cat("\n7. Create a scatterplot of sepal width vs. sepal length.\n")
plot(iris$Sepal.Width, iris$Sepal.Length,
main = "Sepal Width vs. Sepal Length",
xlab = "Sepal Width",
ylab = "Sepal Length",
col = "steelblue"
)

cat("\n8. Create boxplot of sepal width vs. sepal length.\n")
boxplot(Sepal.Width ~ Sepal.Length,
data = iris,
main = "Sepal Width by Sepal Length",
xlab = "Species",
ylab = "Sepal Length",
col = "steelblue",
border = "black"
)

cat("\n9. Find Pearson correlation between Sepal.Length and Petal.Length.\n")
correlation <- cor(iris$Sepal.Length, iris$Petal.Length)
print(correlation)
cat("\n10. Create correlation matrix for dataset.\n")
cor_matrix <- cor(iris[, 1:4])
print(cor_matrix)
```

Output:

```
Write a R language Script for the following operation on Iris Data Set.
1. Load the Iris Dataset
[1] "iris"

2. View the first six rows of the iris dataset.
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
```

3. Summarize iris dataset

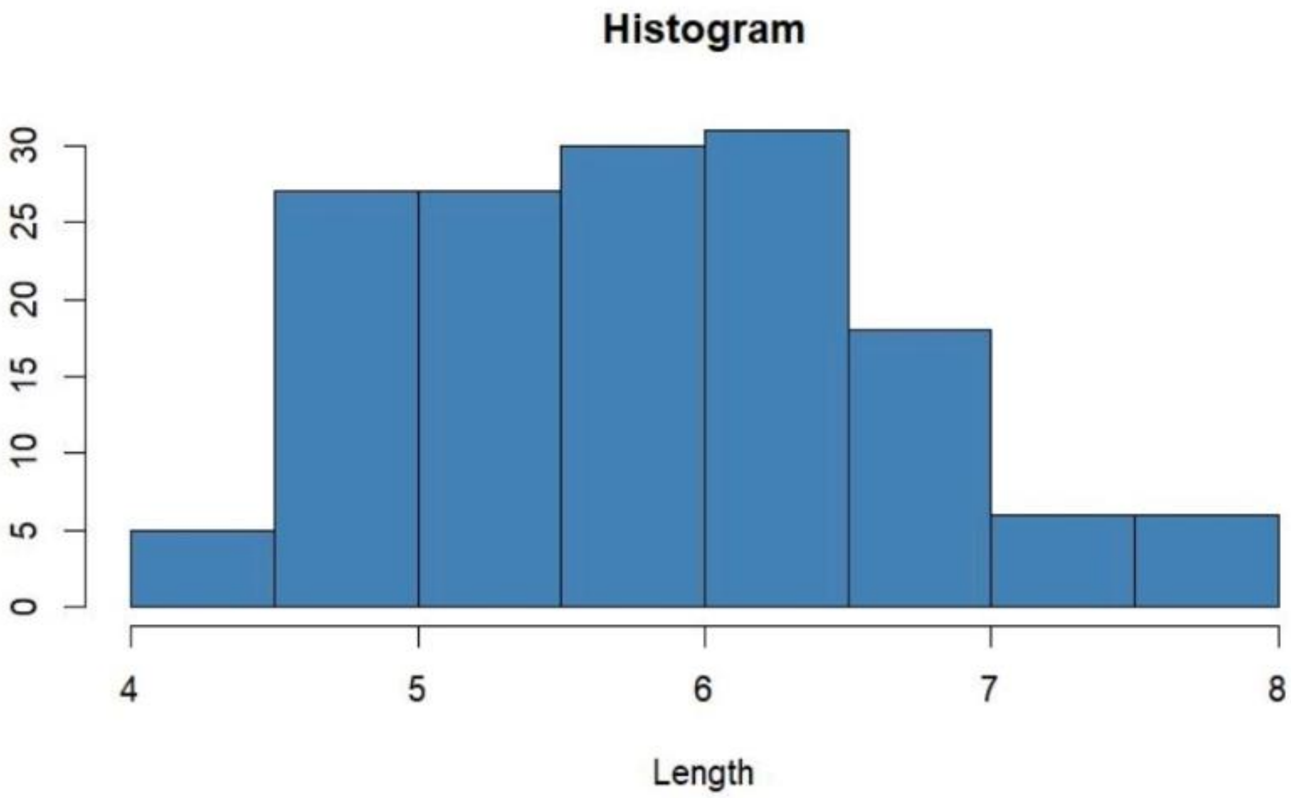
4. Display number of rows and columns.

```
[1] 150 5
```

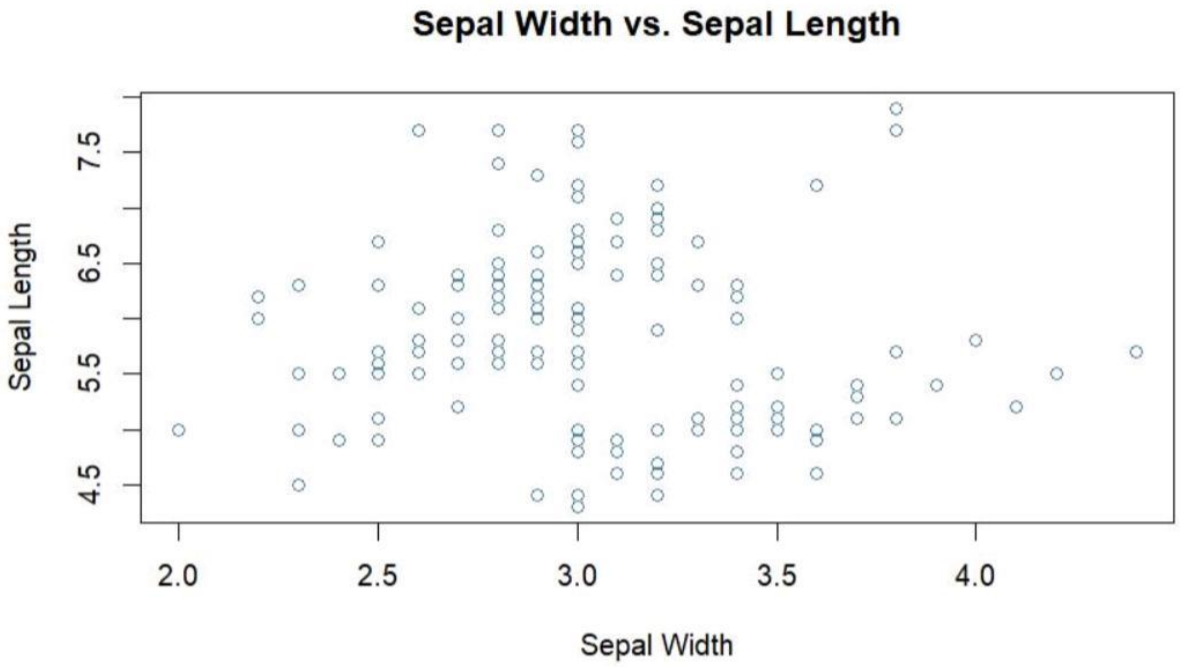
5. Display column names of dataset.

```
[1] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width" "Species"
```

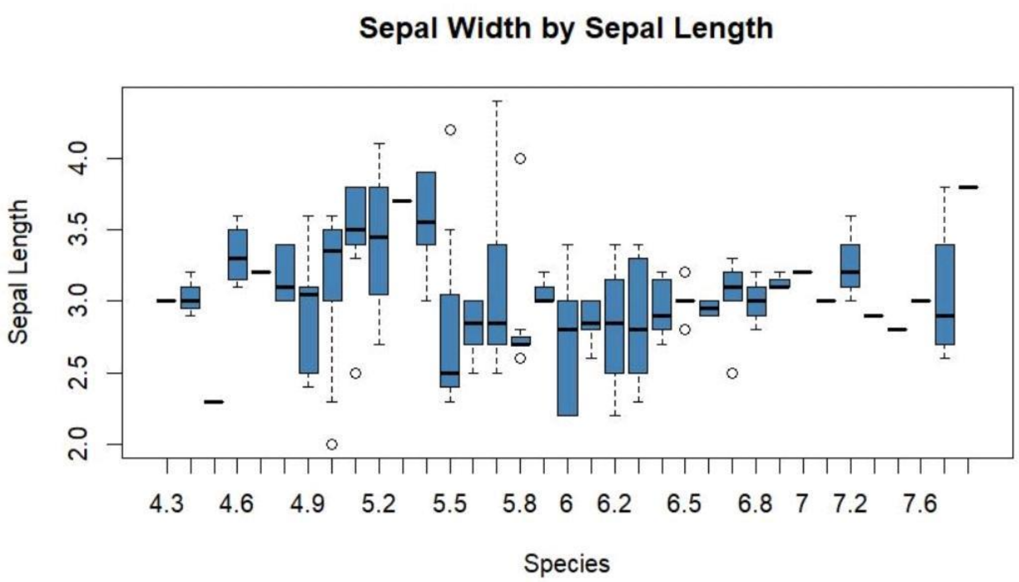
6. Create histogram of values for sepal length



7. Create a scatterplot of sepal width vs. sepal length



8. Create a boxplot of sepal width vs. sepal length



9. Find Pearson correlation between Sepal.Length and Petal.Length.

```
[1] 0.8717538
```

10. Create correlation matrix for dataset.

```
Sepal.Length Sepal.Width Petal.Length
Petal.Width
Sepal.Length
1.0000000 -0.1175698
0.8717538
0.8179411
Sepal.Width
-0.1175698
1.0000000
-0.4284401 -0.3661259
Petal.Length
0.8717538 -0.4284401
1.0000000
0.9628654
Petal.Width
0.8179411 -0.3661259
0.9628654
1.0000000
```