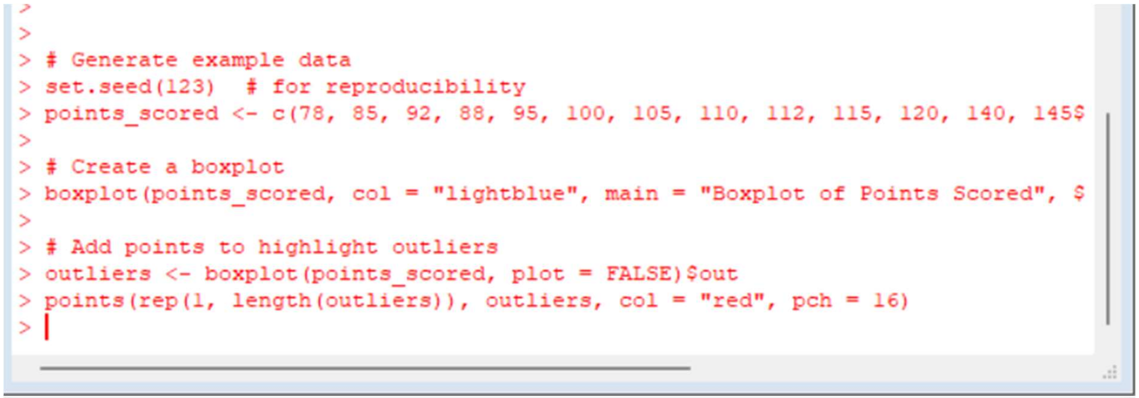
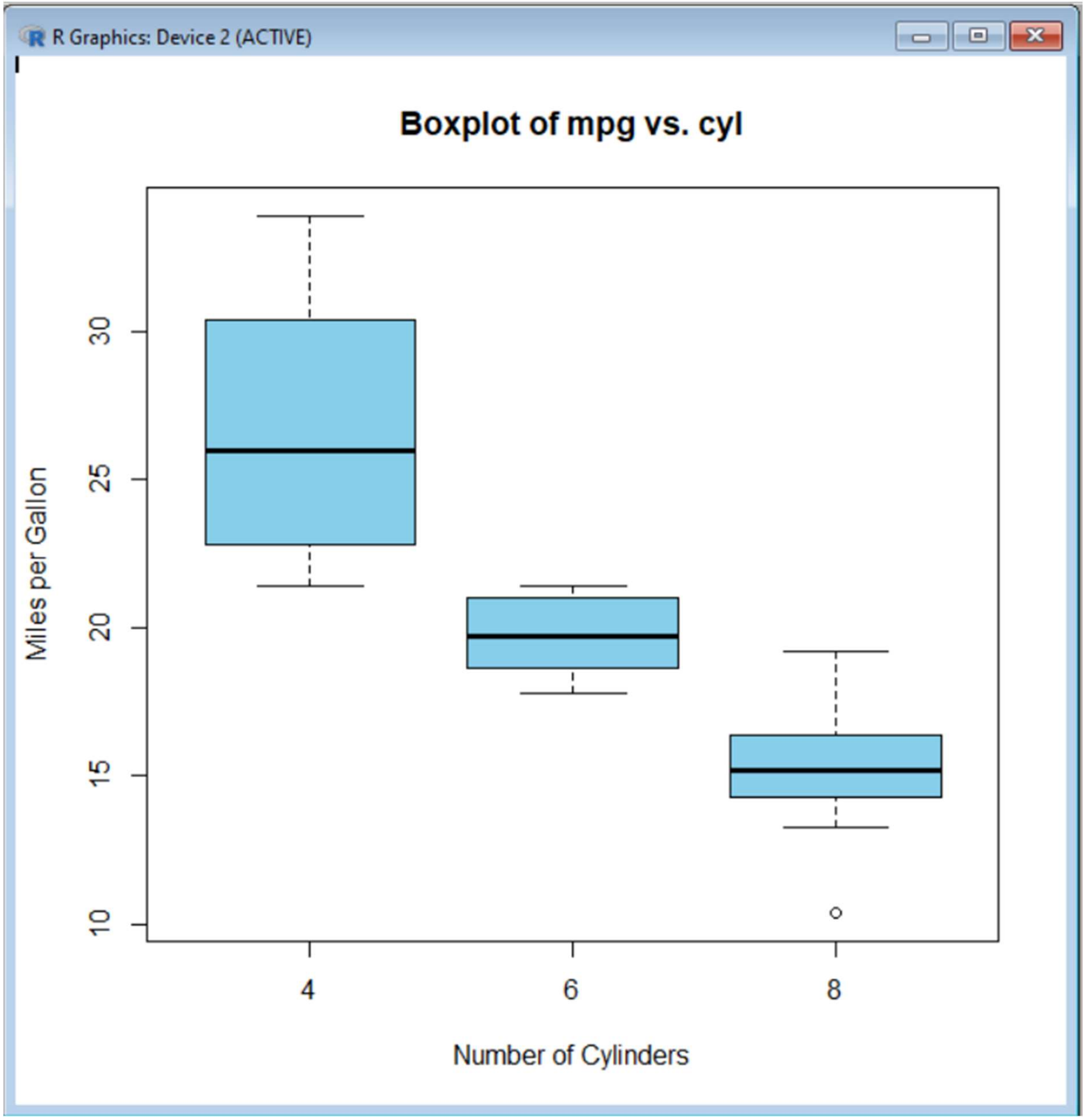
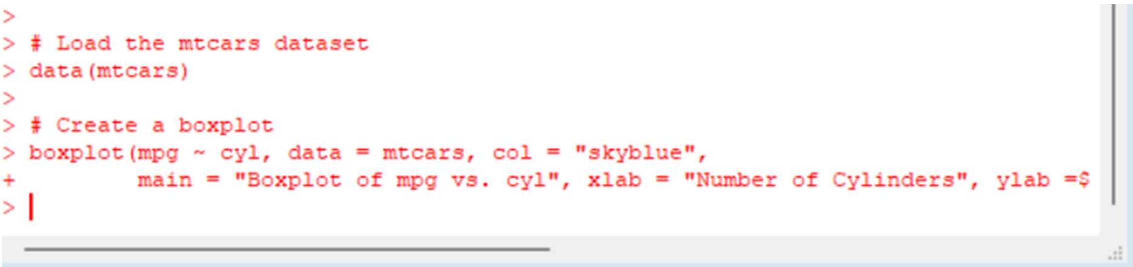
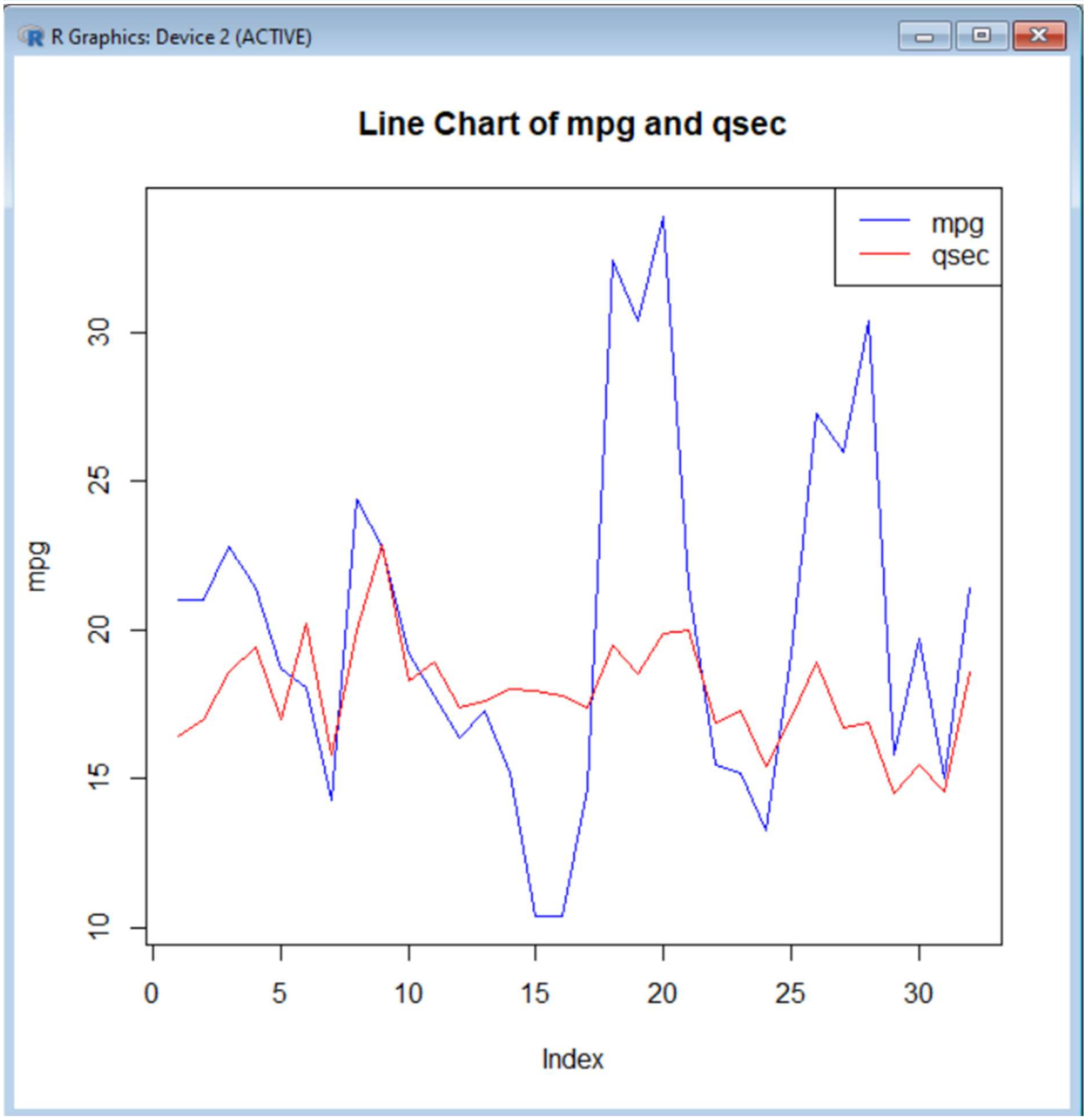


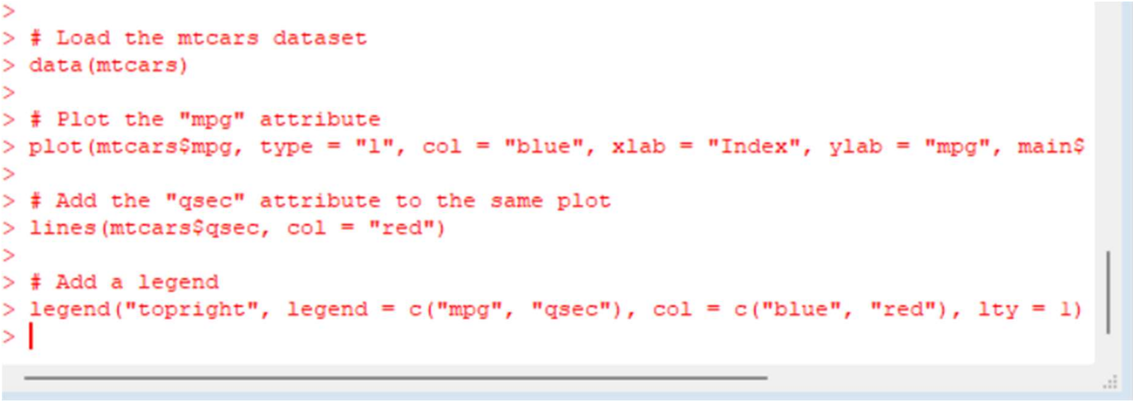
```
>
>
> # Generate example data
> set.seed(123)  # for reproducibility
> points_scored <- c(78, 85, 92, 88, 95, 100, 105, 110, 112, 115, 120, 140, 145$
>
> # Create a boxplot
> boxplot(points_scored, col = "lightblue", main = "Boxplot of Points Scored", $
>
> # Add points to highlight outliers
> outliers <- boxplot(points_scored, plot = FALSE)$out
> points(rep(1, length(outliers)), outliers, col = "red", pch = 16)
> 
```



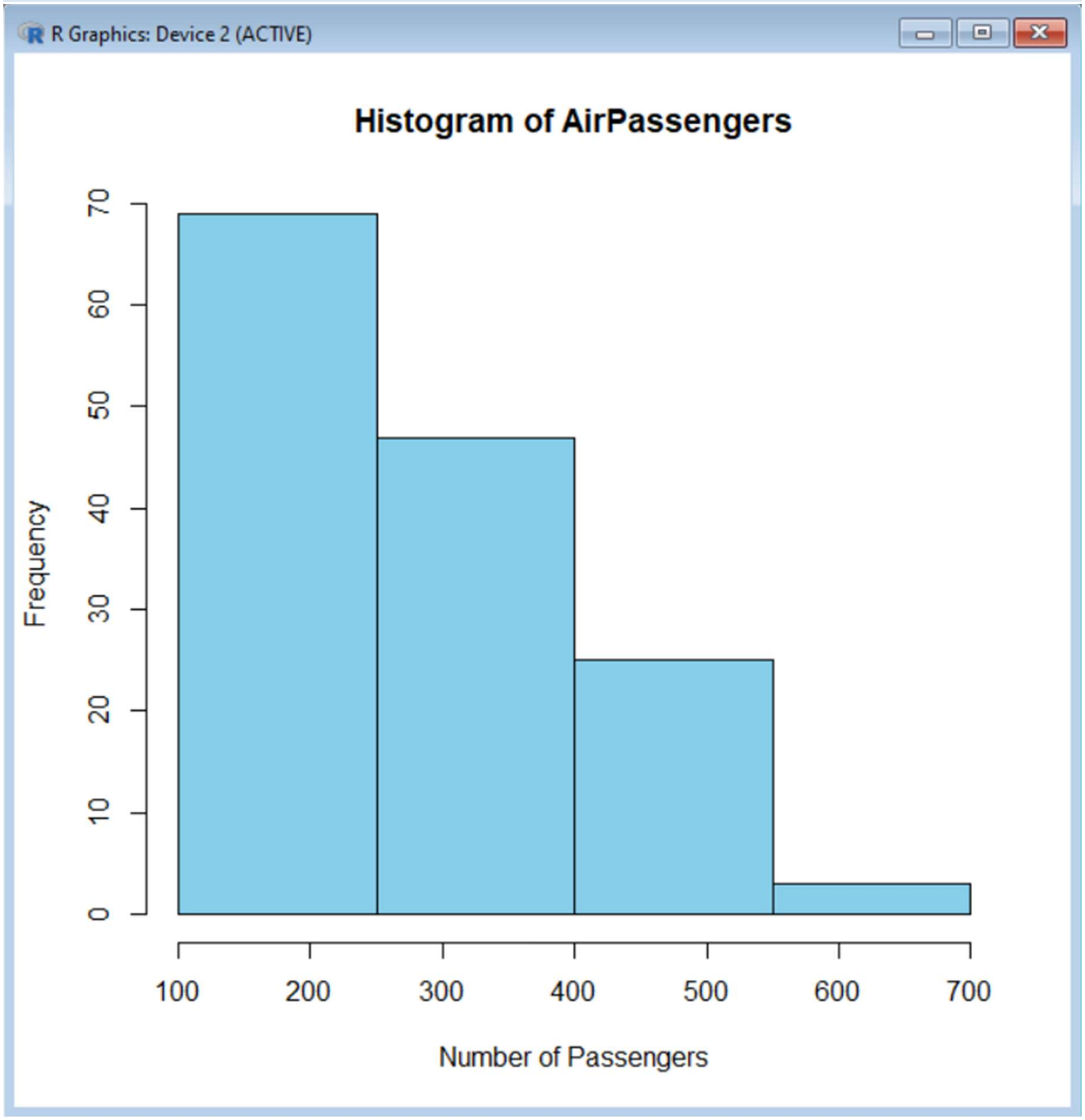


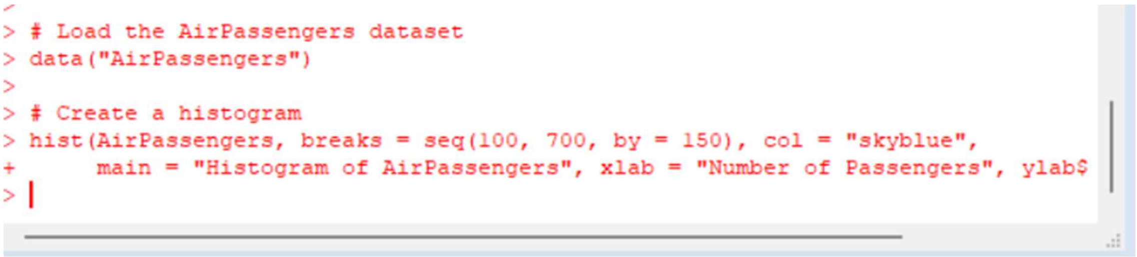
```
>
> # Load the mtcars dataset
> data(mtcars)
>
> # Create a boxplot
> boxplot(mpg ~ cyl, data = mtcars, col = "skyblue",
+         main = "Boxplot of mpg vs. cyl", xlab = "Number of Cylinders", ylab =$
> 
```



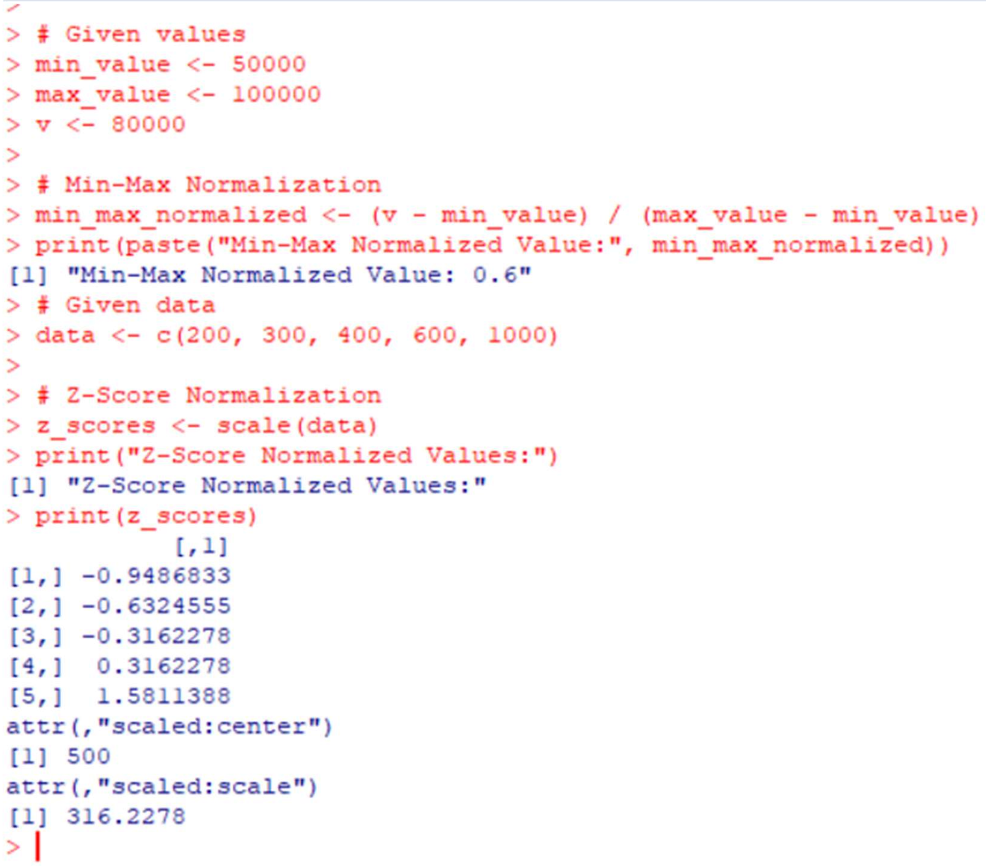


```
>
> # Load the mtcars dataset
> data(mtcars)
>
> # Plot the "mpg" attribute
> plot(mtcars$mpg, type = "l", col = "blue", xlab = "Index", ylab = "mpg", main$
>
> # Add the "qsec" attribute to the same plot
> lines(mtcars$qsec, col = "red")
>
> # Add a legend
> legend("topright", legend = c("mpg", "qsec"), col = c("blue", "red"), lty = 1)
>
```

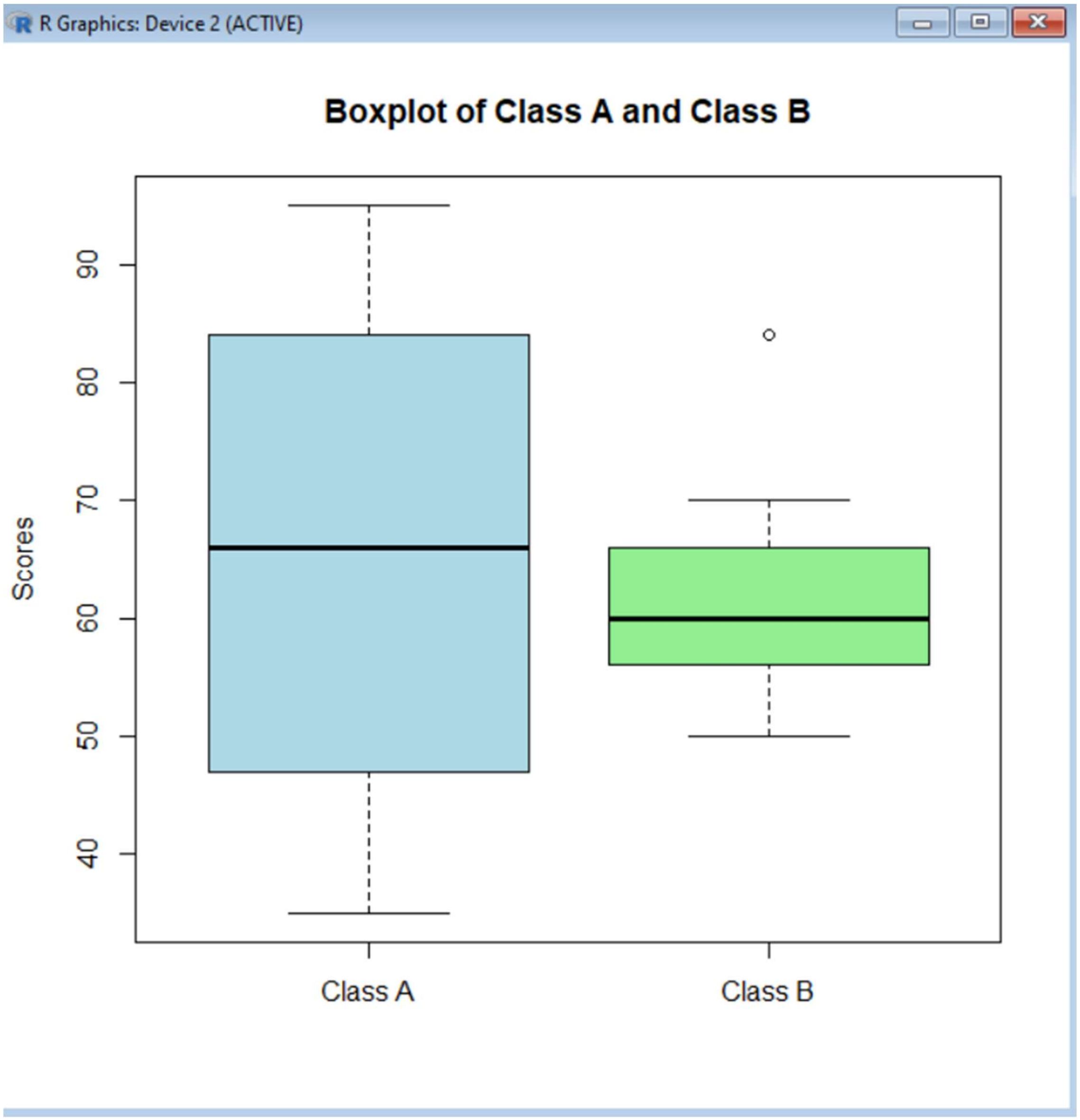


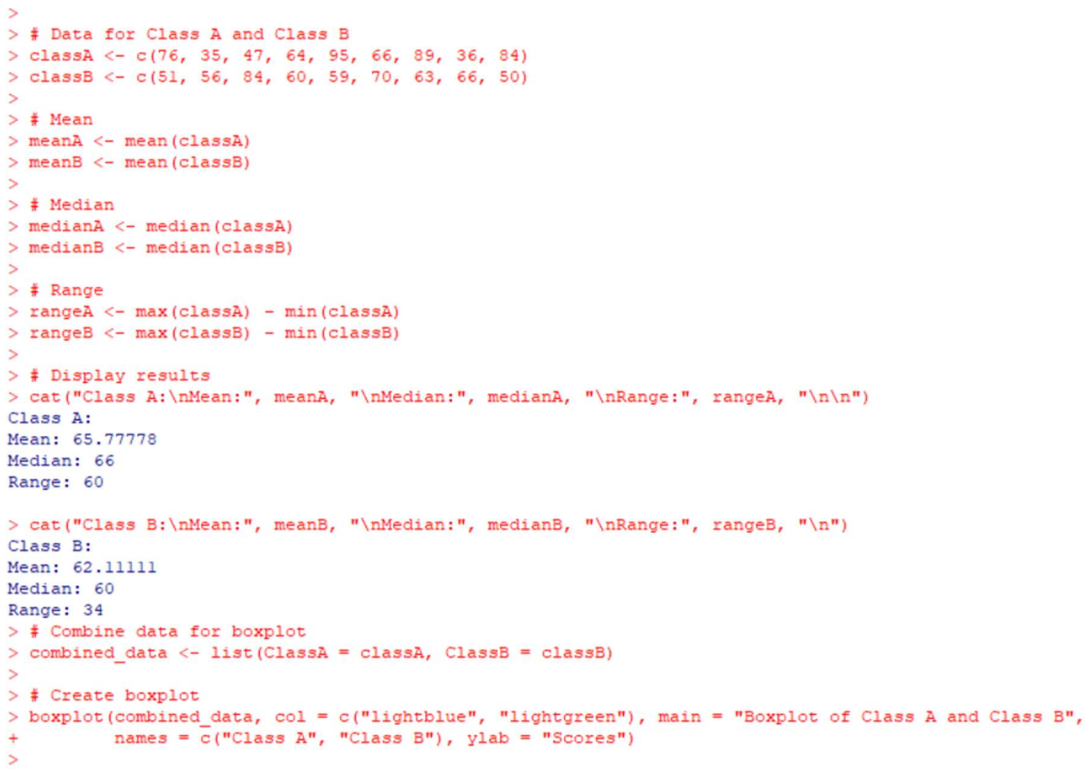


```
> # Load the AirPassengers dataset
> data("AirPassengers")
>
> # Create a histogram
> hist(AirPassengers, breaks = seq(100, 700, by = 150), col = "skyblue",
+      main = "Histogram of AirPassengers", xlab = "Number of Passengers", ylab$
>
```

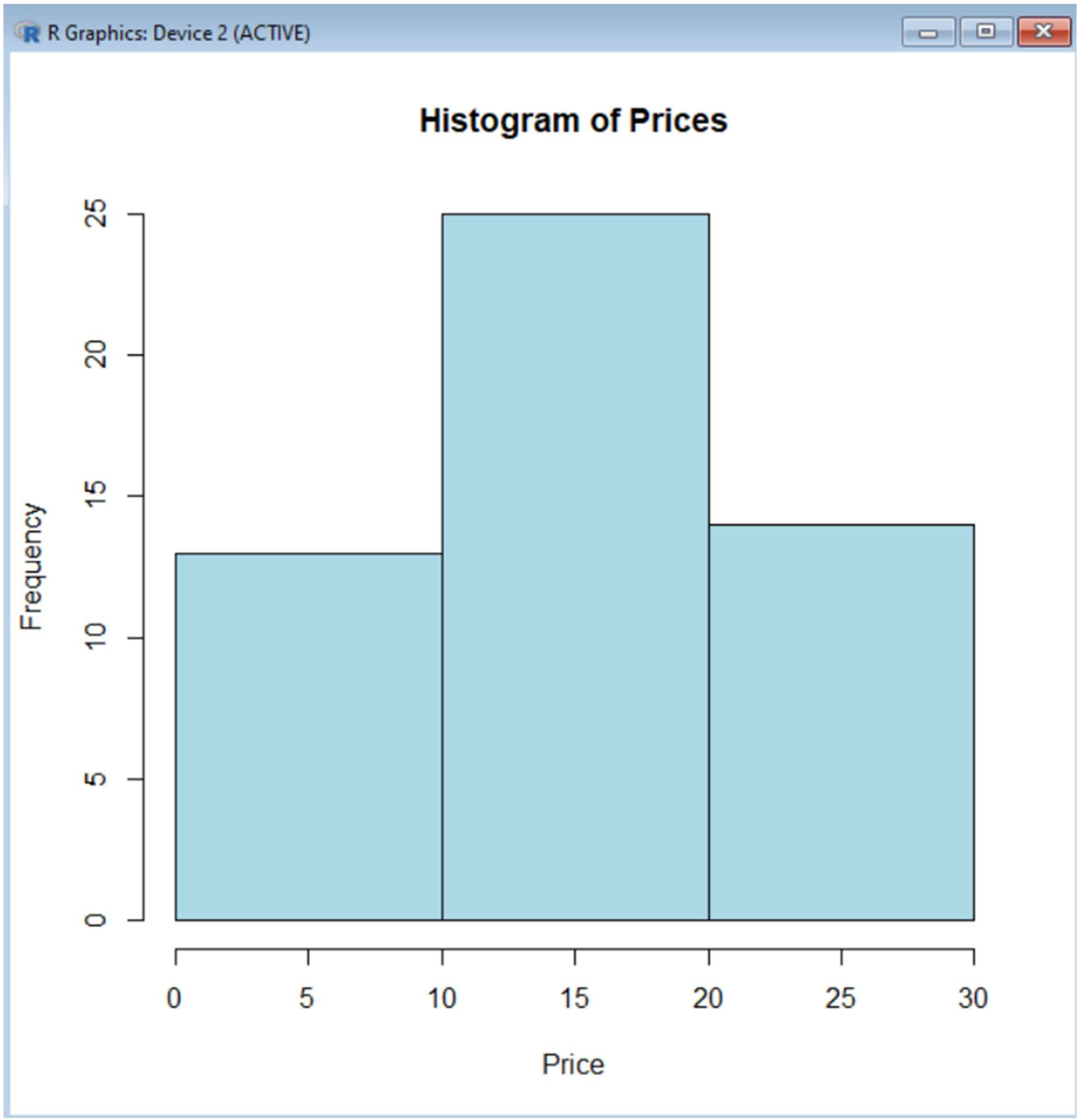


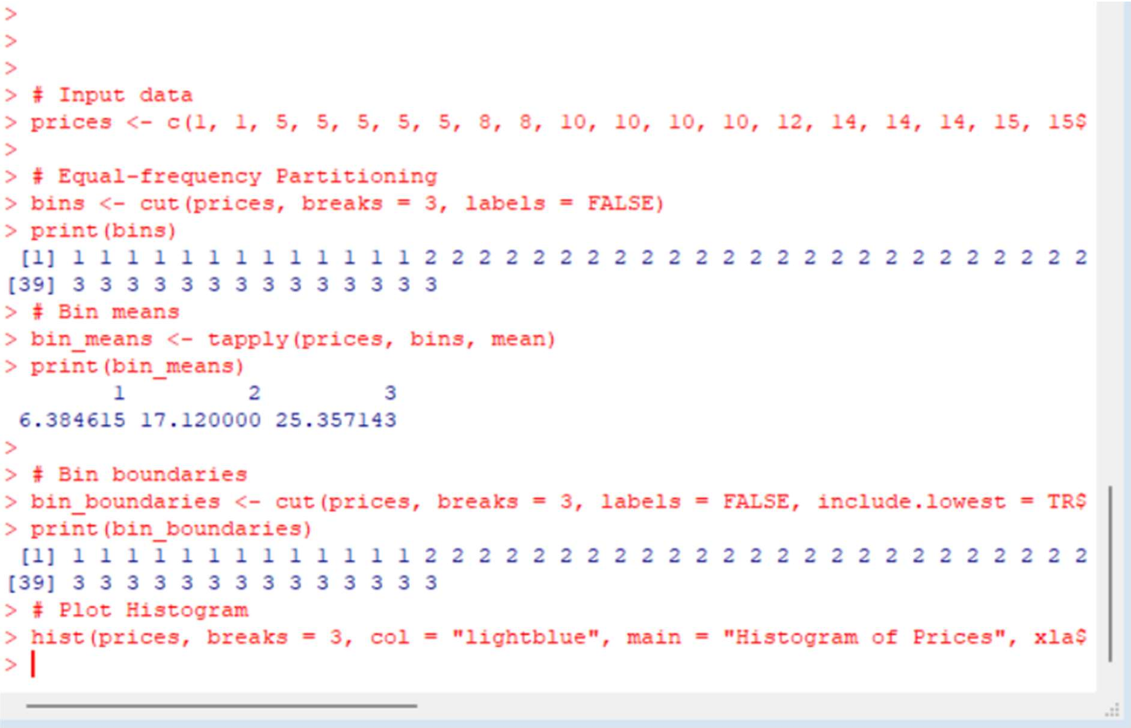
```
> # Given values
> min_value <- 50000
> max_value <- 100000
> v <- 80000
>
> # Min-Max Normalization
> min_max_normalized <- (v - min_value) / (max_value - min_value)
> print(paste("Min-Max Normalized Value:", min_max_normalized))
[1] "Min-Max Normalized Value: 0.6"
> # Given data
> data <- c(200, 300, 400, 600, 1000)
>
> # Z-Score Normalization
> z_scores <- scale(data)
> print("Z-Score Normalized Values:")
[1] "Z-Score Normalized Values:"
> print(z_scores)
           [,1]
[1,] -0.9486833
[2,] -0.6324555
[3,] -0.3162278
[4,]  0.3162278
[5,]  1.5811388
attr(,"scaled:center")
[1] 500
attr(,"scaled:scale")
[1] 316.2278
>
```


R Graphics: Device 2 (ACTIVE)
Boxplot of Class A and Class B
Scores
40
50
60
70
80
90
Class A
Class B

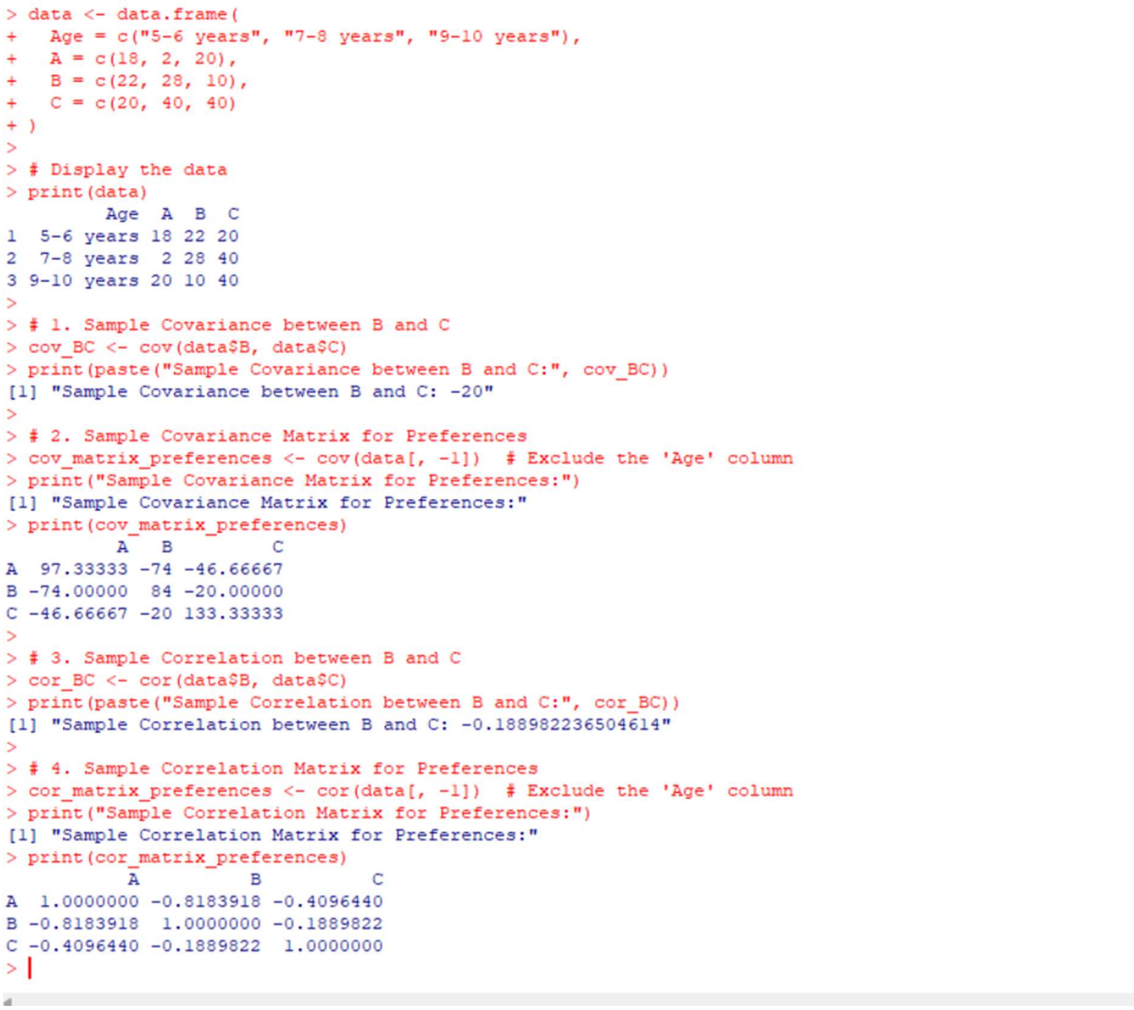


```
>
> # Data for Class A and Class B
> classA <- c(76, 35, 47, 64, 95, 66, 89, 36, 84)
> classB <- c(51, 56, 84, 60, 59, 70, 63, 66, 50)
>
> # Mean
> meanA <- mean(classA)
> meanB <- mean(classB)
>
> # Median
> medianA <- median(classA)
> medianB <- median(classB)
>
> # Range
> rangeA <- max(classA) - min(classA)
> rangeB <- max(classB) - min(classB)
>
> # Display results
> cat("Class A:\nMean:", meanA, "\nMedian:", medianA, "\nRange:", rangeA, "\n\n")
Class A:
Mean: 65.77778
Median: 66
Range: 60

> cat("Class B:\nMean:", meanB, "\nMedian:", medianB, "\nRange:", rangeB, "\n")
Class B:
Mean: 62.11111
Median: 60
Range: 34
> # Combine data for boxplot
> combined_data <- list(ClassA = classA, ClassB = classB)
>
> # Create boxplot
> boxplot(combined_data, col = c("lightblue", "lightgreen"), main = "Boxplot of Class A and Class B",
+         names = c("Class A", "Class B"), ylab = "Scores")
>
```

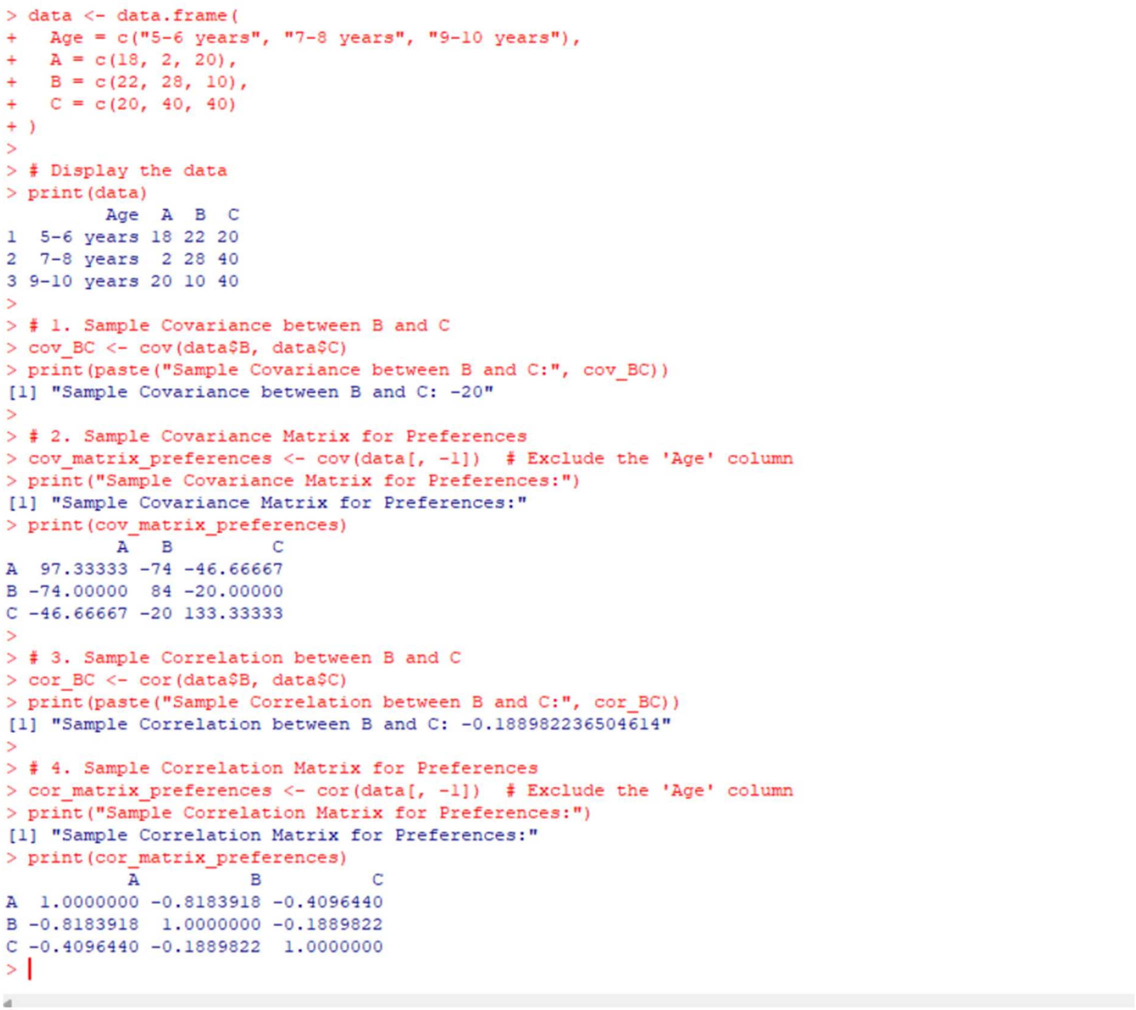

R Graphics: Device 2 (ACTIVE)
Histogram of Prices
Frequency
0
5
10
15
20
25
0
5
10
15
20
25
30
Price



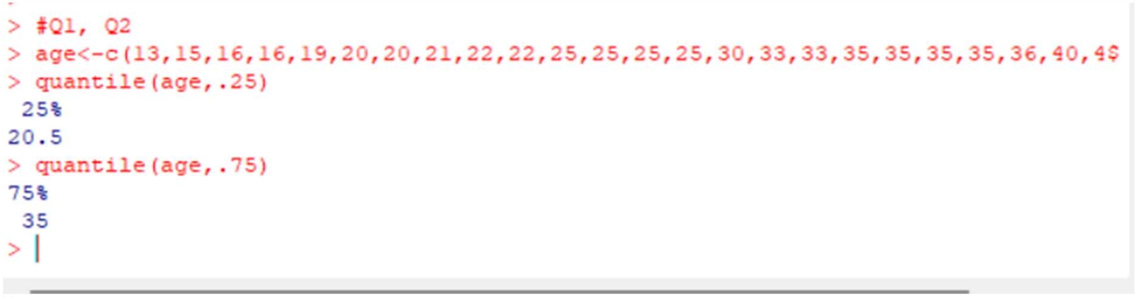
```
>
>
>
> # Input data
> prices <- c(1, 1, 5, 5, 5, 5, 5, 8, 8, 10, 10, 10, 10, 12, 14, 14, 14, 15, 15$
>
> # Equal-frequency Partitioning
> bins <- cut(prices, breaks = 3, labels = FALSE)
> print(bins)
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
[39] 3 3 3 3 3 3 3 3 3 3 3 3 3 3
> # Bin means
> bin_means <- tapply(prices, bins, mean)
> print(bin_means)
        1         2         3
 6.384615 17.120000 25.357143
>
> # Bin boundaries
> bin_boundaries <- cut(prices, breaks = 3, labels = FALSE, include.lowest = TR$
> print(bin_boundaries)
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
[39] 3 3 3 3 3 3 3 3 3 3 3 3 3 3
> # Plot Histogram
> hist(prices, breaks = 3, col = "lightblue", main = "Histogram of Prices", xla$
> 
```



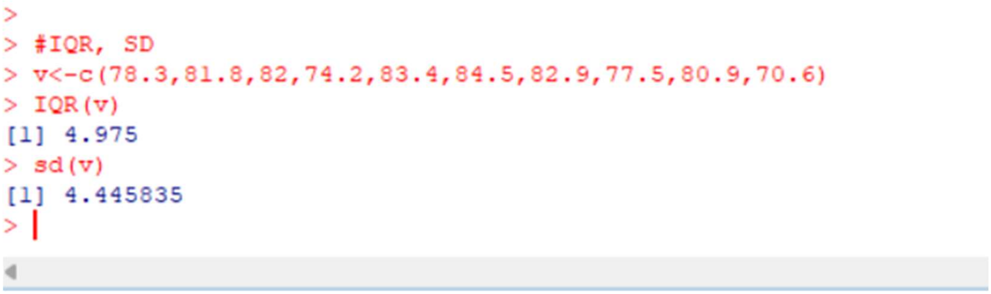
```
> data <- data.frame(
+   Age = c("5-6 years", "7-8 years", "9-10 years"),
+   A = c(18, 2, 20),
+   B = c(22, 28, 10),
+   C = c(20, 40, 40)
+ )
>
> # Display the data
> print(data)
         Age  A  B  C
1  5-6 years 18 22 20
2  7-8 years  2 28 40
3 9-10 years 20 10 40
>
> # 1. Sample Covariance between B and C
> cov_BC <- cov(data$B, data$C)
> print(paste("Sample Covariance between B and C:", cov_BC))
[1] "Sample Covariance between B and C: -20"
>
> # 2. Sample Covariance Matrix for Preferences
> cov_matrix_preferences <- cov(data[, -1])  # Exclude the 'Age' column
> print("Sample Covariance Matrix for Preferences:")
[1] "Sample Covariance Matrix for Preferences:"
> print(cov_matrix_preferences)
          A   B         C
A  97.33333 -74 -46.66667
B -74.00000  84 -20.00000
C -46.66667 -20 133.33333
>
> # 3. Sample Correlation between B and C
> cor_BC <- cor(data$B, data$C)
> print(paste("Sample Correlation between B and C:", cor_BC))
[1] "Sample Correlation between B and C: -0.188982236504614"
>
> # 4. Sample Correlation Matrix for Preferences
> cor_matrix_preferences <- cor(data[, -1])  # Exclude the 'Age' column
> print("Sample Correlation Matrix for Preferences:")
[1] "Sample Correlation Matrix for Preferences:"
> print(cor_matrix_preferences)
           A          B          C
A  1.0000000 -0.8183918 -0.4096440
B -0.8183918  1.0000000 -0.1889822
C -0.4096440 -0.1889822  1.0000000
> 
```



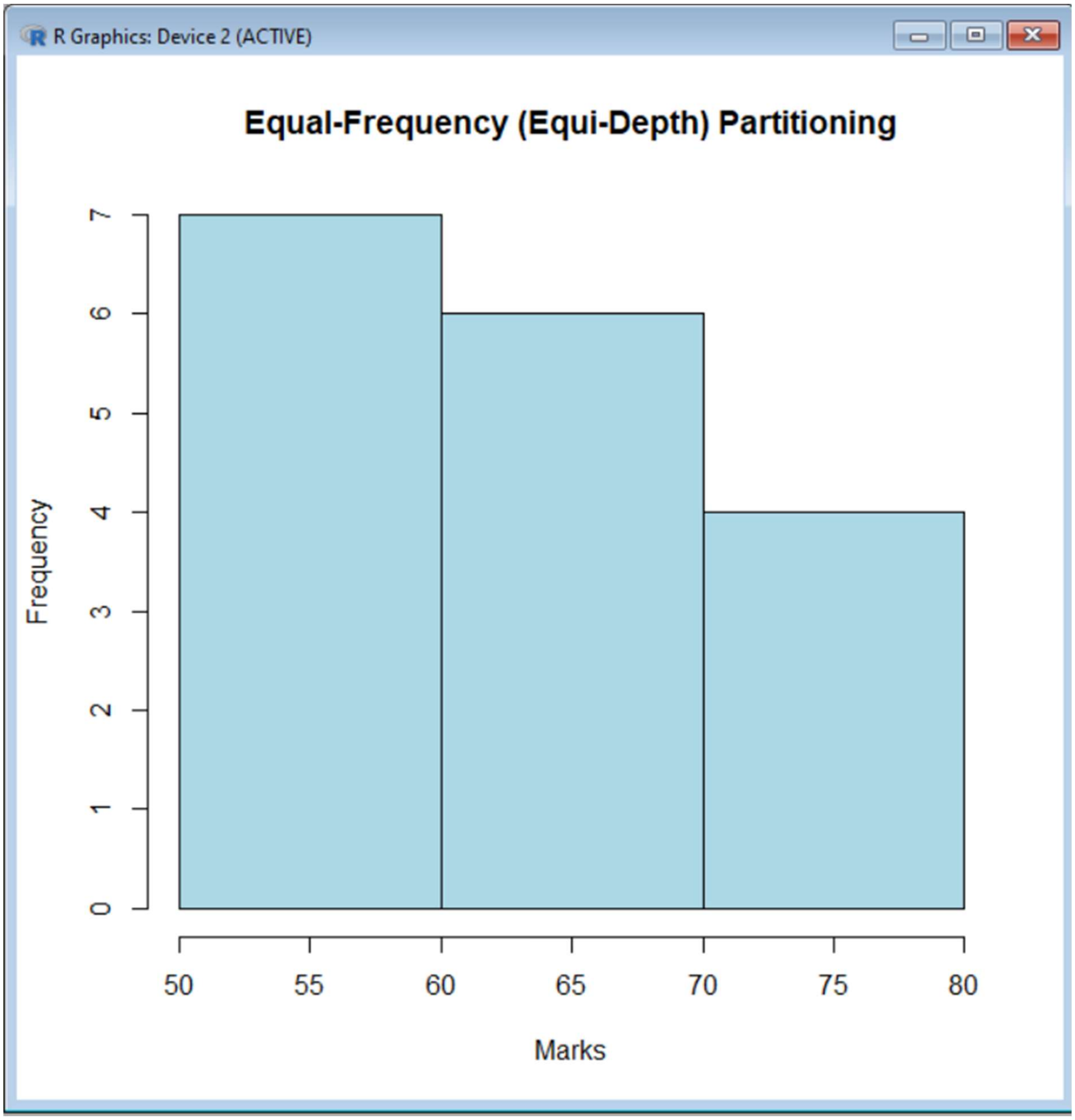
```
> data <- data.frame(
+   Age = c("5-6 years", "7-8 years", "9-10 years"),
+   A = c(18, 2, 20),
+   B = c(22, 28, 10),
+   C = c(20, 40, 40)
+ )
>
> # Display the data
> print(data)
         Age  A  B  C
1  5-6 years 18 22 20
2  7-8 years  2 28 40
3 9-10 years 20 10 40
>
> # 1. Sample Covariance between B and C
> cov_BC <- cov(data$B, data$C)
> print(paste("Sample Covariance between B and C:", cov_BC))
[1] "Sample Covariance between B and C: -20"
>
> # 2. Sample Covariance Matrix for Preferences
> cov_matrix_preferences <- cov(data[, -1])  # Exclude the 'Age' column
> print("Sample Covariance Matrix for Preferences:")
[1] "Sample Covariance Matrix for Preferences:"
> print(cov_matrix_preferences)
          A   B         C
A  97.33333 -74 -46.66667
B -74.00000  84 -20.00000
C -46.66667 -20 133.33333
>
> # 3. Sample Correlation between B and C
> cor_BC <- cor(data$B, data$C)
> print(paste("Sample Correlation between B and C:", cor_BC))
[1] "Sample Correlation between B and C: -0.188982236504614"
>
> # 4. Sample Correlation Matrix for Preferences
> cor_matrix_preferences <- cor(data[, -1])  # Exclude the 'Age' column
> print("Sample Correlation Matrix for Preferences:")
[1] "Sample Correlation Matrix for Preferences:"
> print(cor_matrix_preferences)
           A          B          C
A  1.0000000 -0.8183918 -0.4096440
B -0.8183918  1.0000000 -0.1889822
C -0.4096440 -0.1889822  1.0000000
> 
```

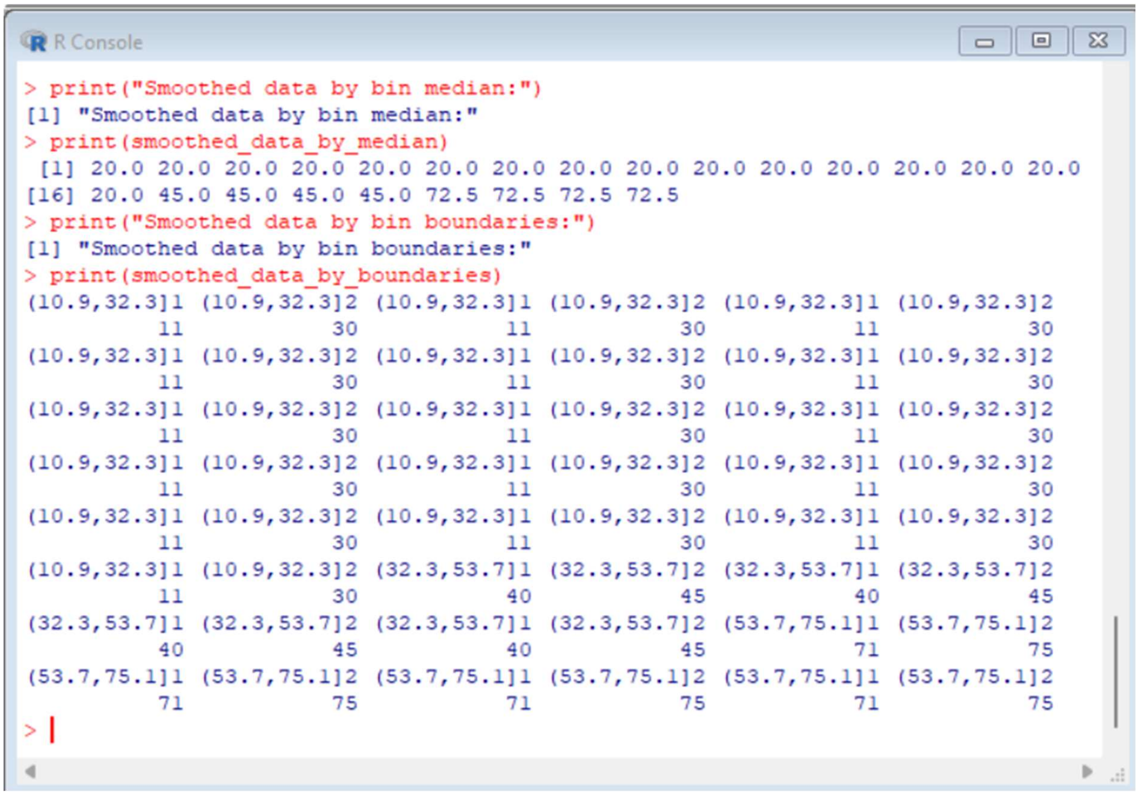


```
> #Q1, Q2
> age<-c(13,15,16,16,19,20,20,21,22,22,25,25,25,25,30,33,33,35,35,35,35,36,40,4$
> quantile(age,.25)
 25%
20.5
> quantile(age,.75)
75%
 35
> 
```

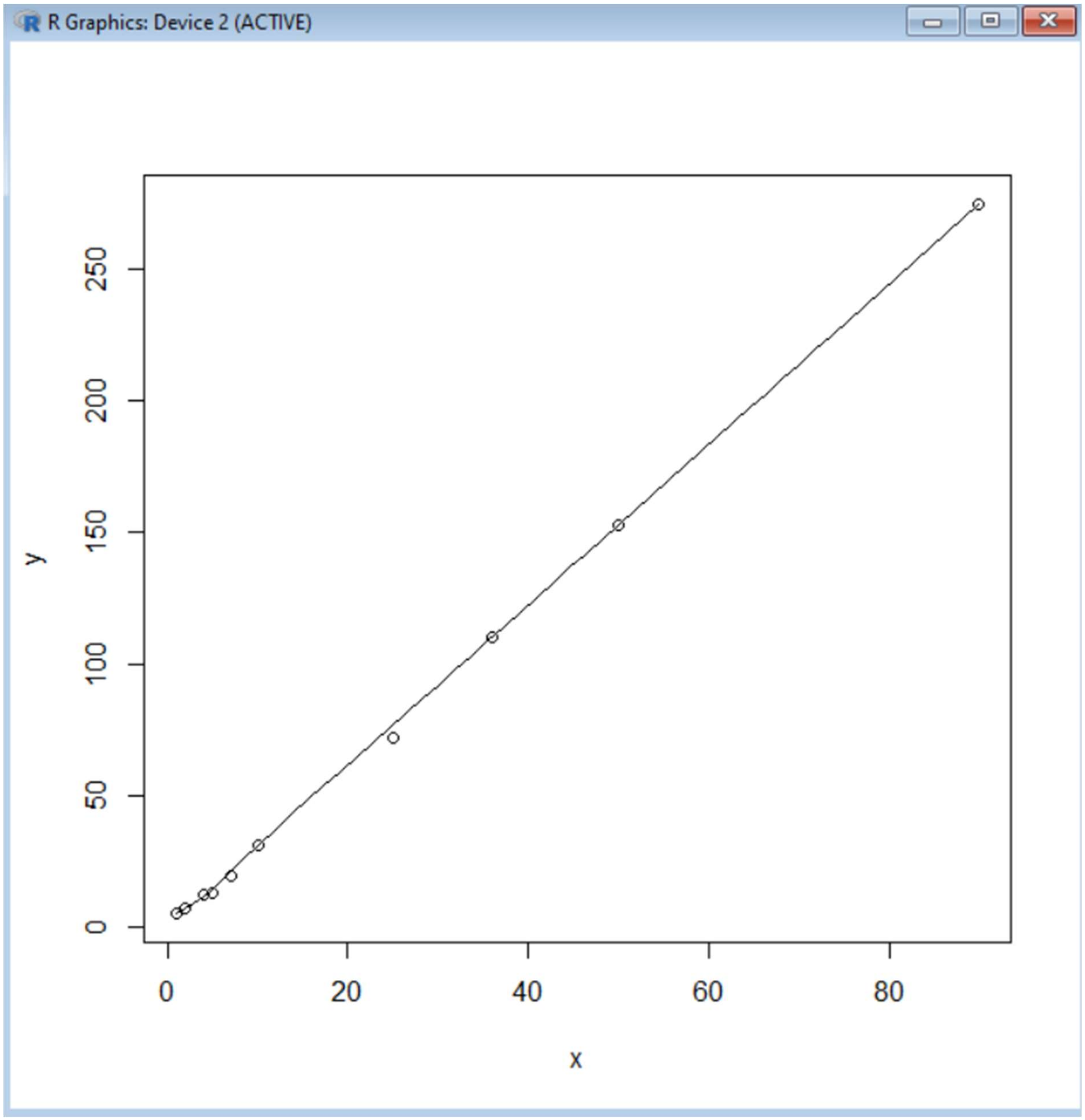


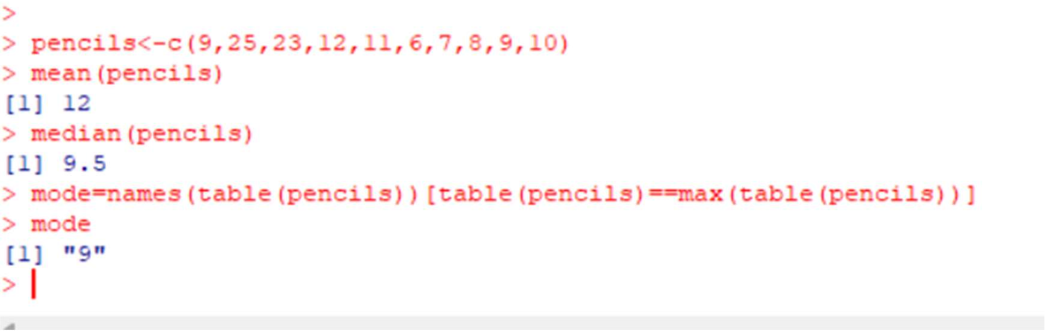
```
>
> #IQR, SD
> v<-c(78.3,81.8,82,74.2,83.4,84.5,82.9,77.5,80.9,70.6)
> IQR(v)
[1] 4.975
> sd(v)
[1] 4.445835
> 
```


R Graphics: Device 2 (ACTIVE)
Equal-Frequency (Equi-Depth) Partitioning
Frequency
0
1
2
3
4
5
6
7
50
55
60
65
70
75
80
Marks

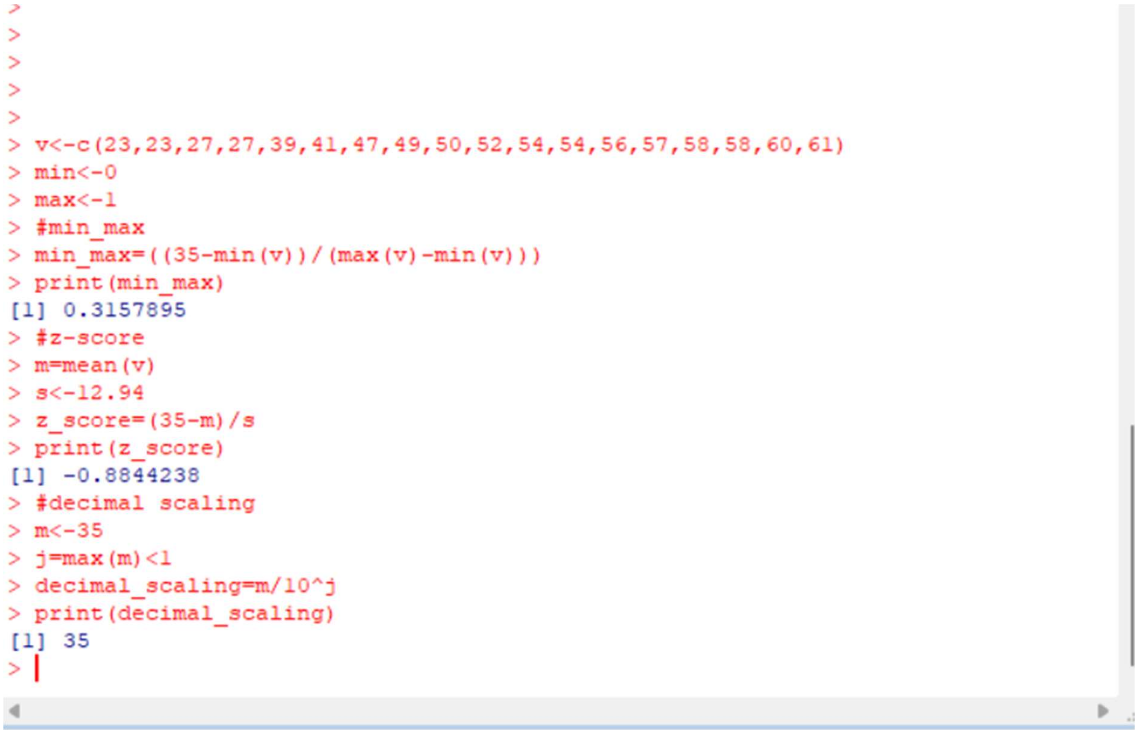


```
> print("Smoothed data by bin median:")
[1] "Smoothed data by bin median:"
> print(smoothed_data_by_median)
 [1] 20.0 20.0 20.0 20.0 20.0 20.0 20.0 20.0 20.0 20.0 20.0 20.0 20.0 20.0 20.0
[16] 20.0 45.0 45.0 45.0 45.0 72.5 72.5 72.5 72.5
> print("Smoothed data by bin boundaries:")
[1] "Smoothed data by bin boundaries:"
> print(smoothed_data_by_boundaries)
(10.9,32.3]1 (10.9,32.3]2 (10.9,32.3]1 (10.9,32.3]2 (10.9,32.3]1 (10.9,32.3]2 
          11           30           11           30           11           30 
(10.9,32.3]1 (10.9,32.3]2 (10.9,32.3]1 (10.9,32.3]2 (10.9,32.3]1 (10.9,32.3]2 
          11           30           11           30           11           30 
(10.9,32.3]1 (10.9,32.3]2 (10.9,32.3]1 (10.9,32.3]2 (10.9,32.3]1 (10.9,32.3]2 
          11           30           11           30           11           30 
(10.9,32.3]1 (10.9,32.3]2 (10.9,32.3]1 (10.9,32.3]2 (10.9,32.3]1 (10.9,32.3]2 
          11           30           11           30           11           30 
(10.9,32.3]1 (10.9,32.3]2 (10.9,32.3]1 (10.9,32.3]2 (10.9,32.3]1 (10.9,32.3]2 
          11           30           11           30           11           30 
(10.9,32.3]1 (10.9,32.3]2 (32.3,53.7]1 (32.3,53.7]2 (32.3,53.7]1 (32.3,53.7]2 
          11           30           40           45           40           45 
(32.3,53.7]1 (32.3,53.7]2 (32.3,53.7]1 (32.3,53.7]2 (53.7,75.1]1 (53.7,75.1]2 
          40           45           40           45           71           75 
(53.7,75.1]1 (53.7,75.1]2 (53.7,75.1]1 (53.7,75.1]2 (53.7,75.1]1 (53.7,75.1]2 
          71           75           71           75           71           75 
> 
```

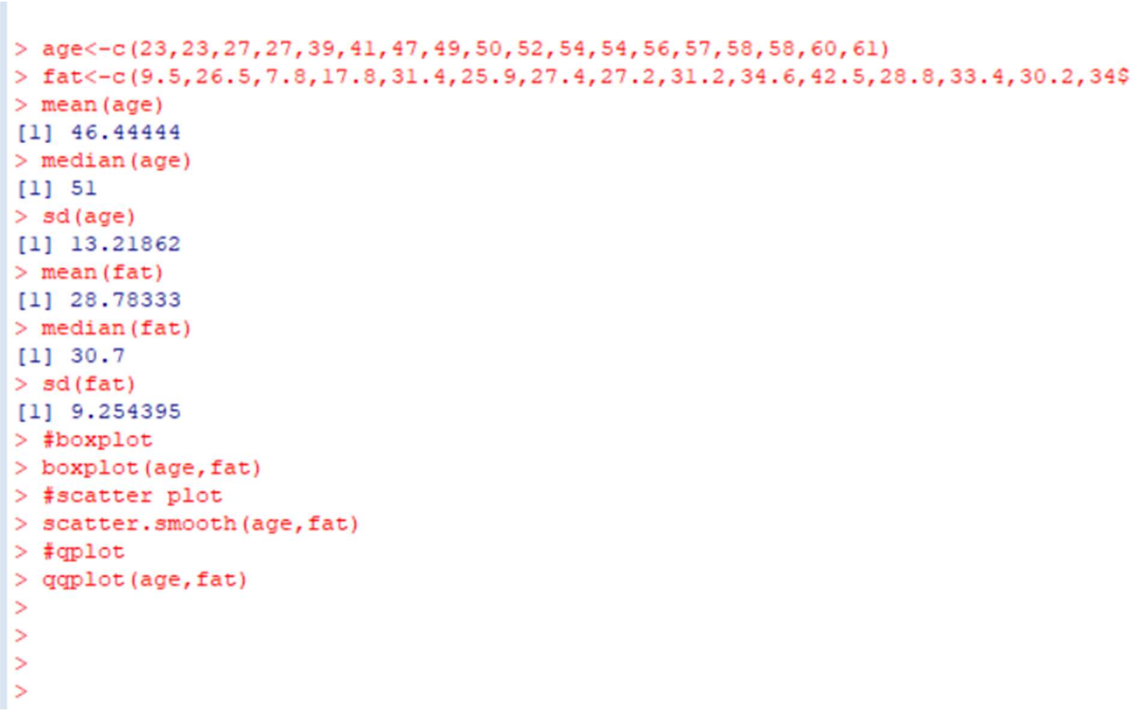




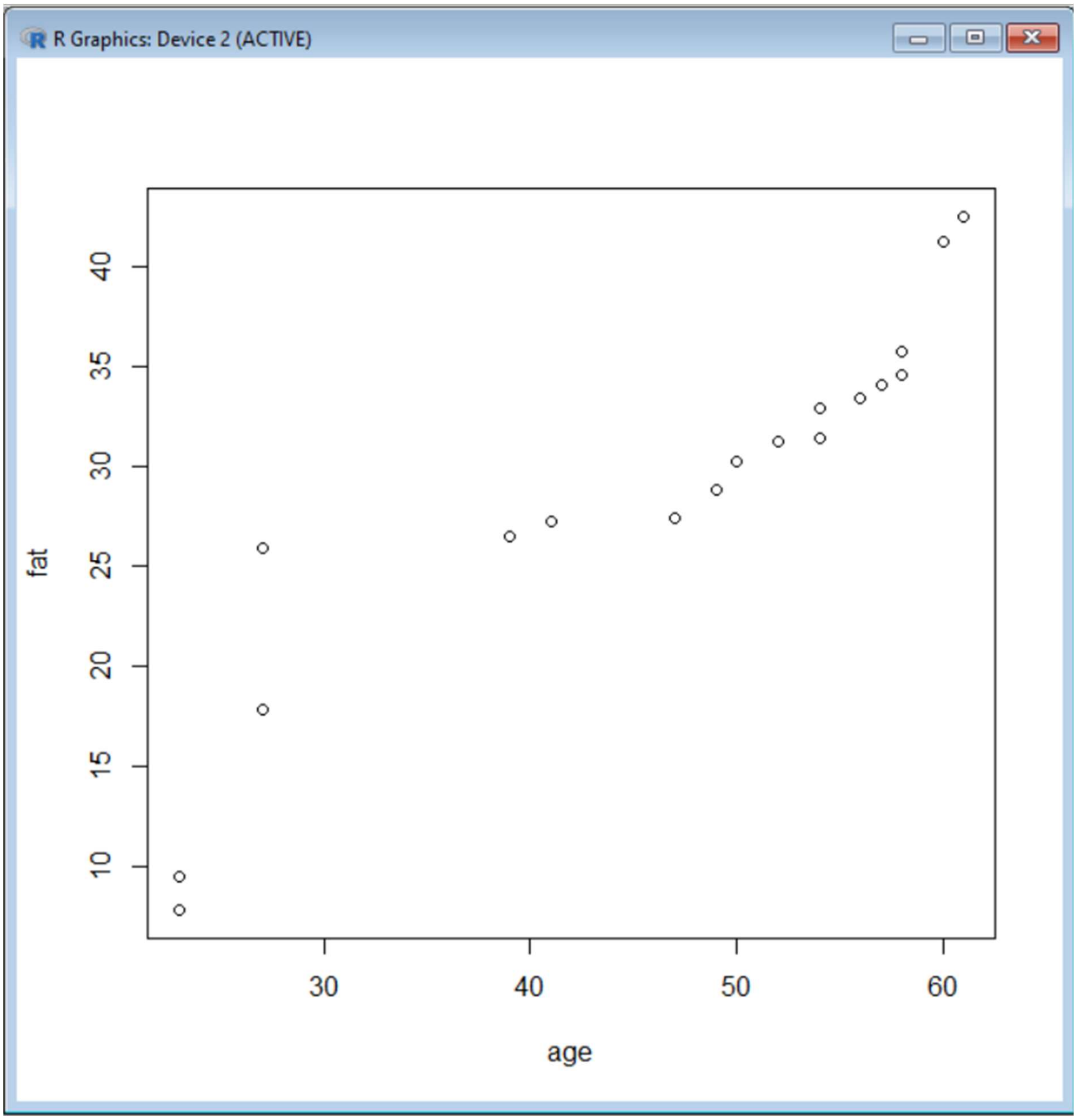
```
>
> pencils<-c(9,25,23,12,11,6,7,8,9,10)
> mean(pencils)
[1] 12
> median(pencils)
[1] 9.5
> mode=names(table(pencils))[table(pencils)==max(table(pencils))]
> mode
[1] "9"
> 
```

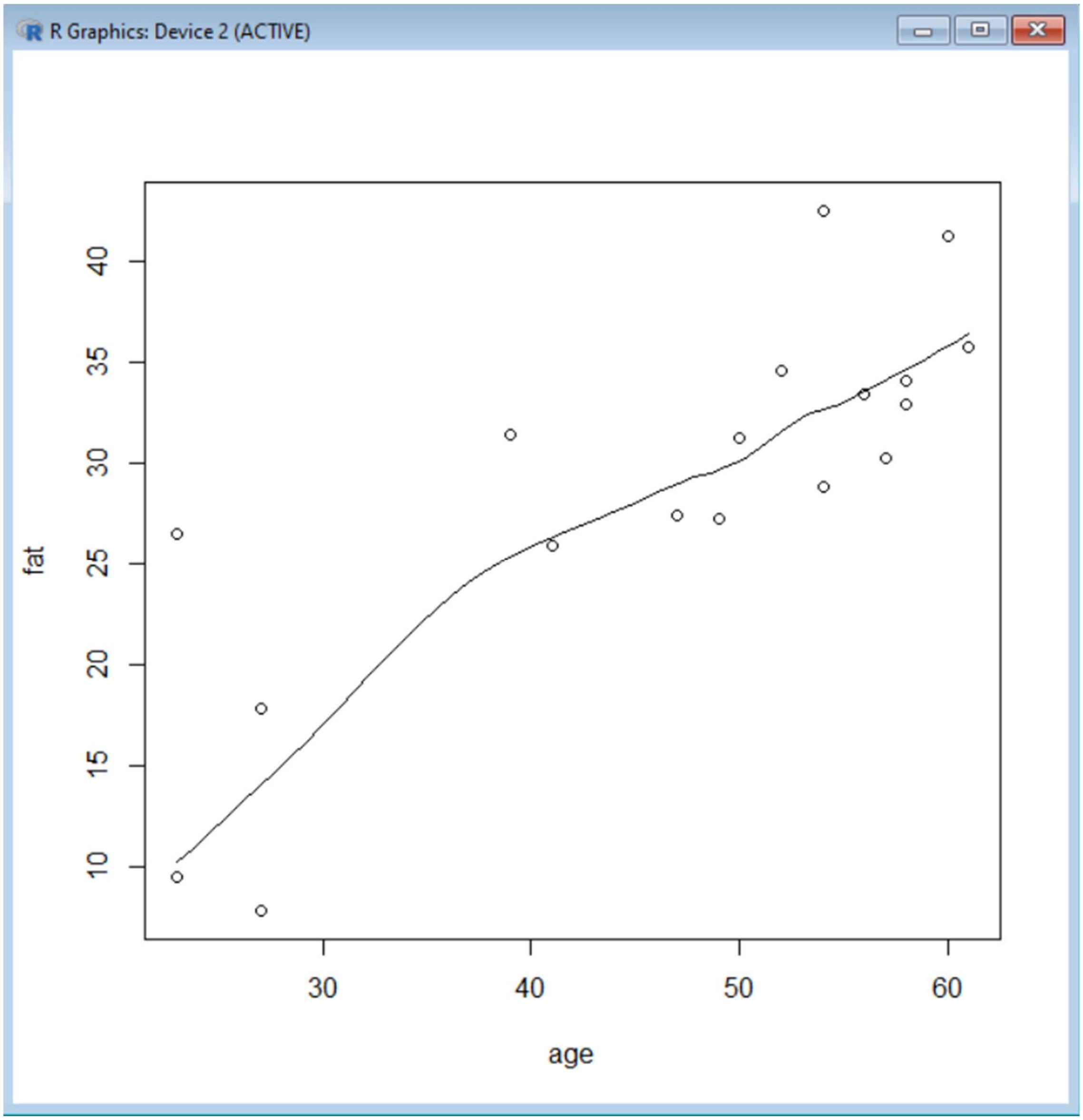


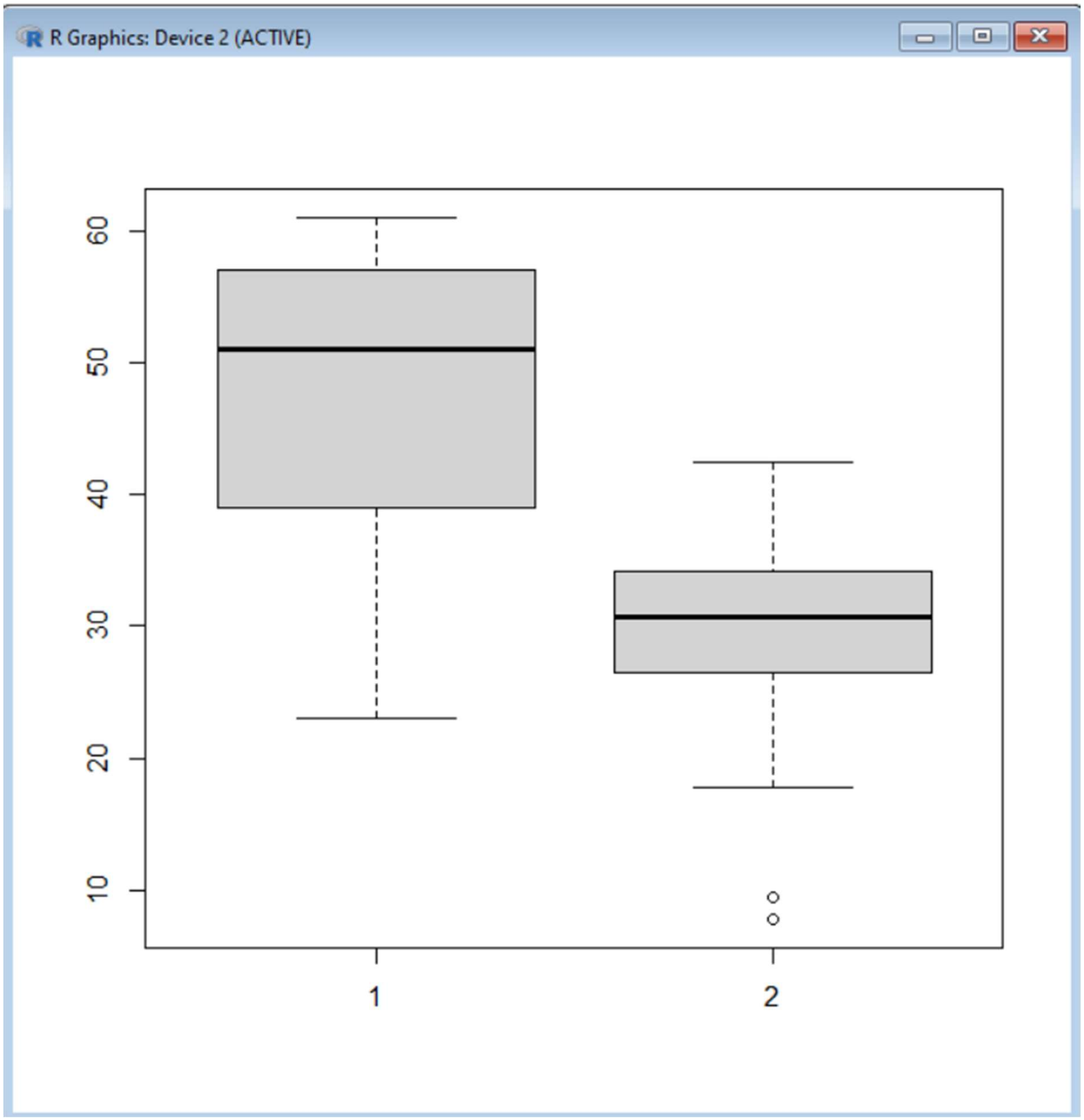
```
>
>
>
>
>
> v<-c(23,23,27,27,39,41,47,49,50,52,54,54,56,57,58,58,60,61)
> min<-0
> max<-1
> #min_max
> min_max=((35-min(v))/(max(v)-min(v)))
> print(min_max)
[1] 0.3157895
> #z-score
> m=mean(v)
> s<-12.94
> z_score=(35-m)/s
> print(z_score)
[1] -0.8844238
> #decimal scaling
> m<-35
> j=max(m)<1
> decimal_scaling=m/10^j
> print(decimal_scaling)
[1] 35
> 
```

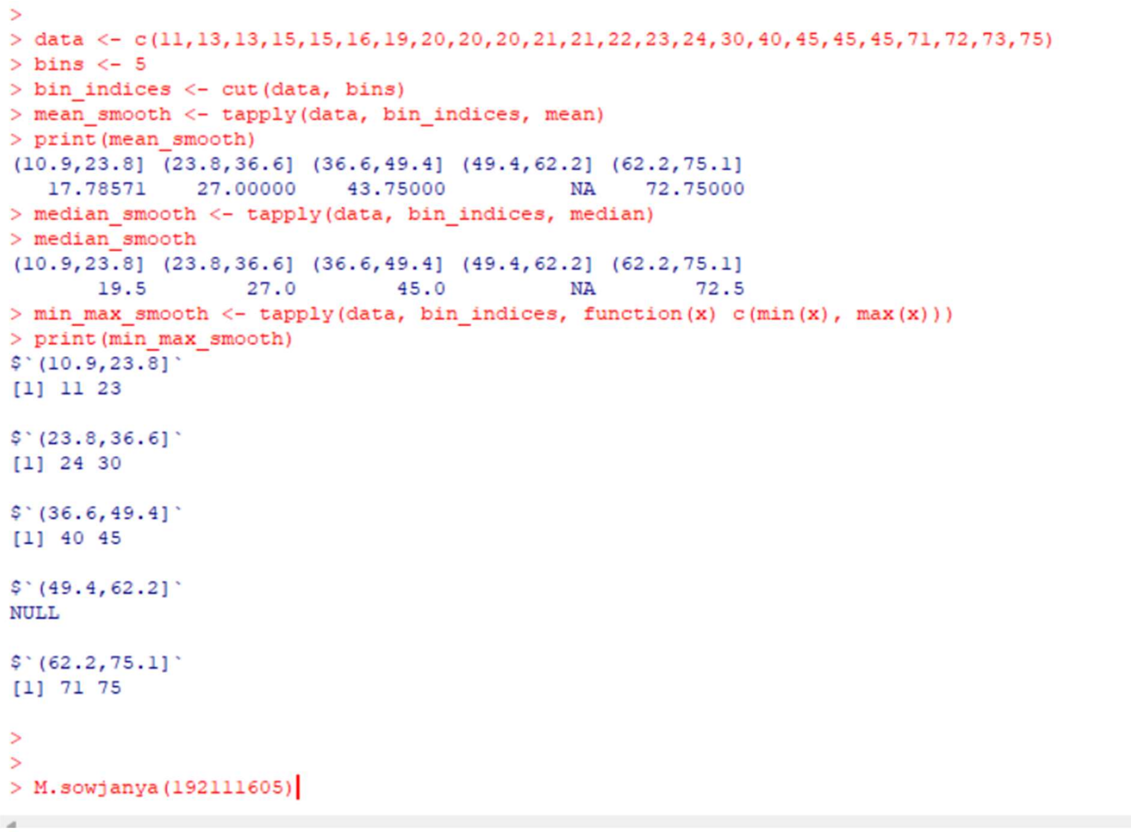


```
> age<-c(23,23,27,27,39,41,47,49,50,52,54,54,56,57,58,58,60,61)
> fat<-c(9.5,26.5,7.8,17.8,31.4,25.9,27.4,27.2,31.2,34.6,42.5,28.8,33.4,30.2,34$
> mean(age)
[1] 46.44444
> median(age)
[1] 51
> sd(age)
[1] 13.21862
> mean(fat)
[1] 28.78333
> median(fat)
[1] 30.7
> sd(fat)
[1] 9.254395
> #boxplot
> boxplot(age,fat)
> #scatter plot
> scatter.smooth(age,fat)
> #qplot
> qqplot(age,fat)
>
>
>
>
```

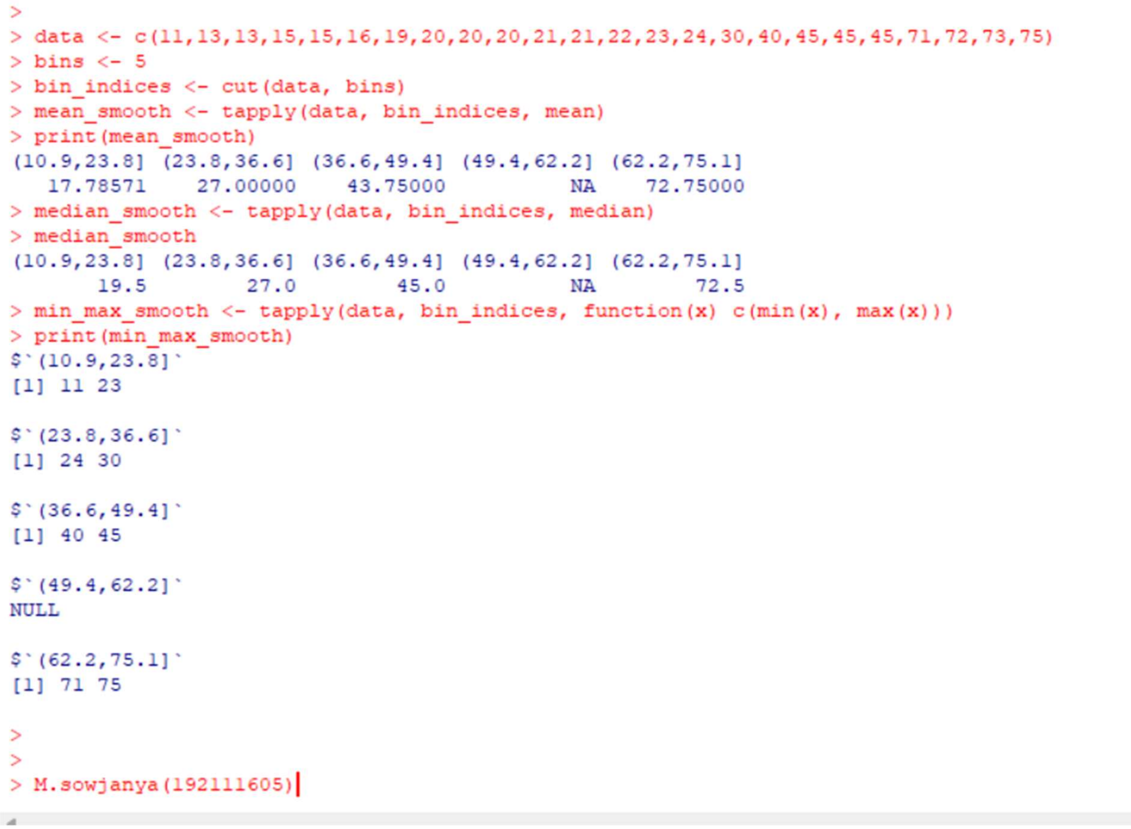

R Graphics: Device 2 (ACTIVE)
fat
10
15
20
25
30
35
40
30
40
50
60
age


R Graphics: Device 2 (ACTIVE)
fat
10
15
20
25
30
35
40
age
30
40
50
60

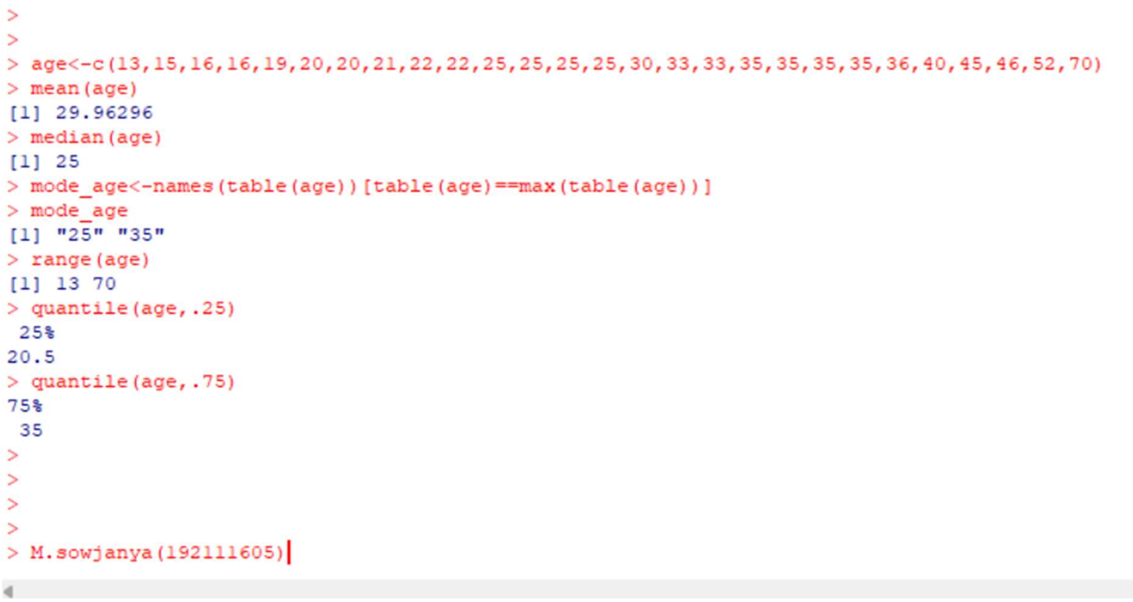

R Graphics: Device 2 (ACTIVE)
10
20
30
40
50
60
1
2



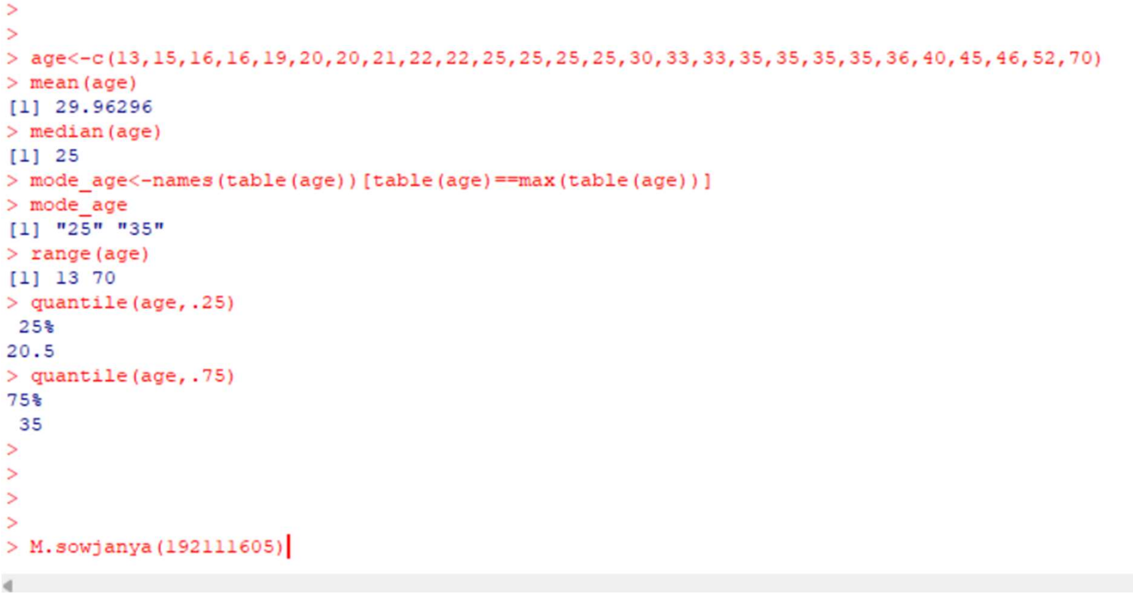
```
>
> data <- c(11,13,13,15,15,16,19,20,20,20,21,21,22,23,24,30,40,45,45,45,71,72,73,75)
> bins <- 5
> bin_indices <- cut(data, bins)
> mean_smooth <- tapply(data, bin_indices, mean)
> print(mean_smooth)
(10.9,23.8] (23.8,36.6] (36.6,49.4] (49.4,62.2] (62.2,75.1]
   17.78571    27.00000    43.75000          NA    72.75000
> median_smooth <- tapply(data, bin_indices, median)
> median_smooth
(10.9,23.8] (23.8,36.6] (36.6,49.4] (49.4,62.2] (62.2,75.1]
       19.5        27.0        45.0          NA        72.5
> min_max_smooth <- tapply(data, bin_indices, function(x) c(min(x), max(x)))
> print(min_max_smooth)
$`(10.9,23.8]`
[1] 11 23

$`(23.8,36.6]`
[1] 24 30

$`(36.6,49.4]`
[1] 40 45

$`(49.4,62.2]`
NULL

$`(62.2,75.1]`
[1] 71 75

>
>
> M.sowjanya(192111605)
```



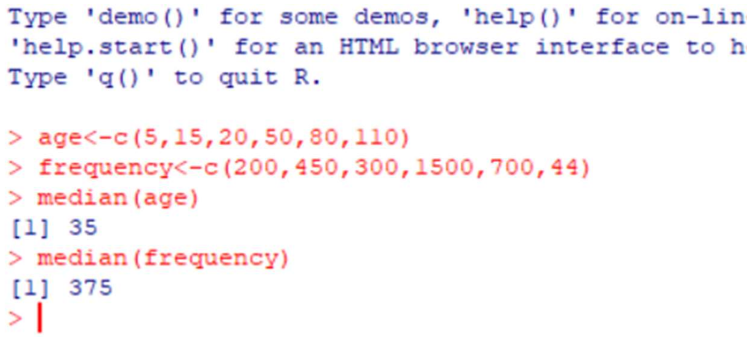
```
>
> data <- c(11,13,13,15,15,16,19,20,20,20,21,21,22,23,24,30,40,45,45,45,71,72,73,75)
> bins <- 5
> bin_indices <- cut(data, bins)
> mean_smooth <- tapply(data, bin_indices, mean)
> print(mean_smooth)
(10.9,23.8] (23.8,36.6] (36.6,49.4] (49.4,62.2] (62.2,75.1]
   17.78571    27.00000    43.75000          NA    72.75000
> median_smooth <- tapply(data, bin_indices, median)
> median_smooth
(10.9,23.8] (23.8,36.6] (36.6,49.4] (49.4,62.2] (62.2,75.1]
       19.5        27.0        45.0          NA        72.5
> min_max_smooth <- tapply(data, bin_indices, function(x) c(min(x), max(x)))
> print(min_max_smooth)
$`(10.9,23.8]`
[1] 11 23

$`(23.8,36.6]`
[1] 24 30

$`(36.6,49.4]`
[1] 40 45

$`(49.4,62.2]`
NULL

$`(62.2,75.1]`
[1] 71 75

>
>
> M.sowjanya(192111605)
```



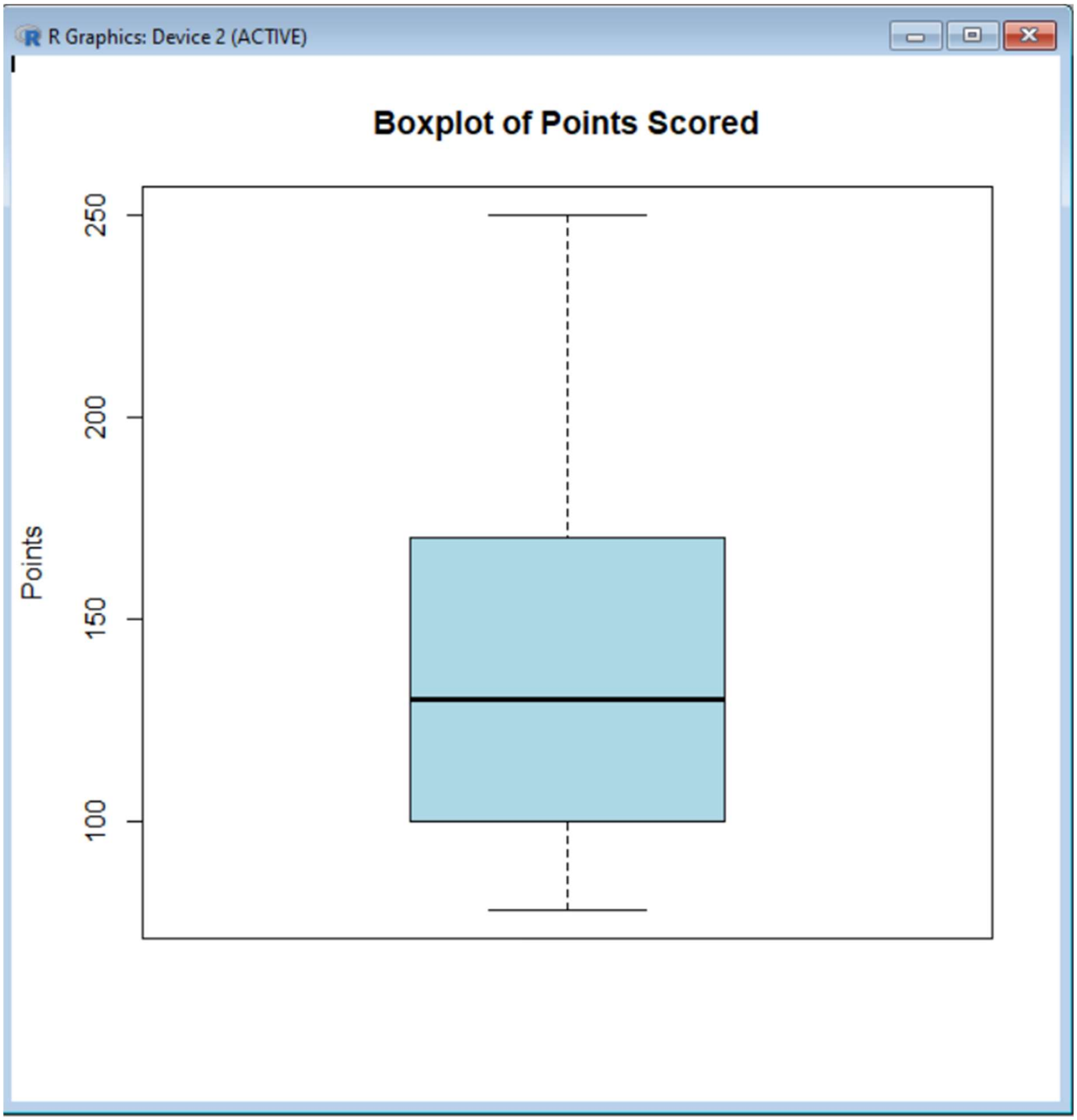
```
>
>
> age<-c(13,15,16,16,19,20,20,21,22,22,25,25,25,25,30,33,33,35,35,35,35,36,40,45,46,52,70)
> mean(age)
[1] 29.96296
> median(age)
[1] 25
> mode_age<-names(table(age))[table(age)==max(table(age))]
> mode_age
[1] "25" "35"
> range(age)
[1] 13 70
> quantile(age,.25)
 25%
20.5
> quantile(age,.75)
75%
 35
>
>
>
>
> M.sowjanya(192111605)
```



```
>
>
> age<-c(13,15,16,16,19,20,20,21,22,22,25,25,25,25,30,33,33,35,35,35,35,36,40,45,46,52,70)
> mean(age)
[1] 29.96296
> median(age)
[1] 25
> mode_age<-names(table(age))[table(age)==max(table(age))]
> mode_age
[1] "25" "35"
> range(age)
[1] 13 70
> quantile(age,.25)
 25%
20.5
> quantile(age,.75)
75%
 35
>
>
>
>
> M.sowjanya(192111605)
```



```
Type 'demo()' for some demos, 'help()' for on-lin
'help.start()' for an HTML browser interface to h
Type 'q()' to quit R.

> age<-c(5,15,20,50,80,110)
> frequency<-c(200,450,300,1500,700,44)
> median(age)
[1] 35
> median(frequency)
[1] 375
> 
```


R Graphics: Device 2 (ACTIVE)
Boxplot of Points Scored
Points