A close-up of a school of engineering

Description automatically generated

**LAB PROGRAMS (61-80)**

**ON**

**ITA0402-Statistics with R Programming for Data Visualization**

**SLOT B**

**Submitted by**

**192321047 – KAUSHIK NARAYANAN V**

**To**

**DR.P. Jesu Jayarin**

**Saveetha School of Engineering**

**SIMATS, Thandalam.**

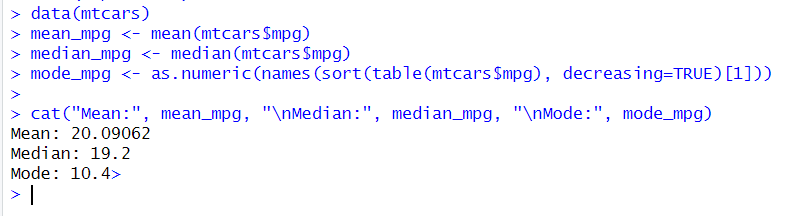
**61. Mean, Median, and Mode of mpg in mtcars**

**Aim:** To calculate the mean, median, and mode of the mpg variable in the mtcars dataset.

r

data(mtcars) mean\_mpg <- mean(mtcars$mpg) median\_mpg <- median(mtcars$mpg) mode\_mpg <- as.numeric(names(sort(table(mtcars$mpg), decreasing=TRUE)[1])) cat("Mean:", mean\_mpg, "\nMedian:", median\_mpg, "\nMode:", mode\_mpg)

**Output:**

****

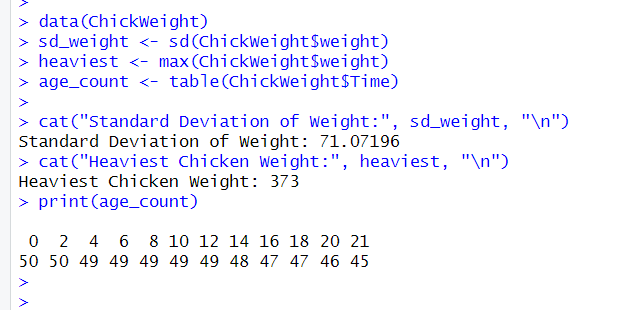
**62. ChickWeight Dataset Analysis**

**Aim:** To analyze the weights, count males and females, find the heaviest chicken, and count chickens by age group.

r

data(ChickWeight) sd\_weight <- sd(ChickWeight$weight) heaviest <- max(ChickWeight$weight) age\_count <- table(ChickWeight$Time) cat("Standard Deviation of Weight:", sd\_weight, "\n") cat("Heaviest Chicken Weight:", heaviest, "\n") print(age\_count)

**Output:**



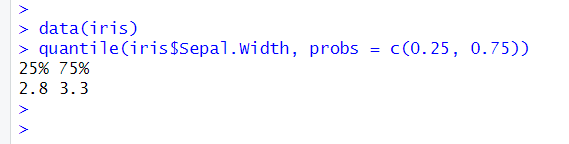
**63. First and Third Quartiles of Sepal.Width in iris Dataset**

**Aim:** To calculate the first and third quartiles of Sepal.Width.

r

data(iris) quantile(iris$Sepal.Width, probs = c(0.25, 0.75))

**Output:**



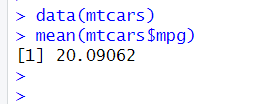
**64. Mean of mpg in mtcars**

**Aim:** To calculate the mean of the mpg variable in mtcars.

r

data(mtcars) mean(mtcars$mpg)

**Output:**

****

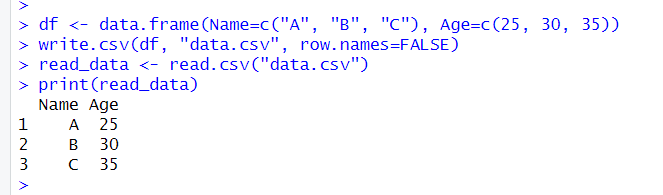
**65. Read and Write CSV File**

**Aim:** To create, write, and read a .csv file.

r

df <- data.frame(Name=c("A", "B", "C"), Age=c(25, 30, 35)) write.csv(df, "data.csv", row.names=FALSE) read\_data <- read.csv("data.csv") print(read\_data)

**Output:**

****

**66. First and Third Quartiles of Sepal.Width in iris Dataset (Repeated)**

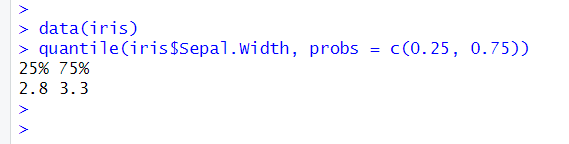
**Aim:** To calculate the first and third quartiles of Sepal.Width.

r

CopyEdit

data(iris) quantile(iris$Sepal.Width, probs = c(0.25, 0.75))

**Output:**



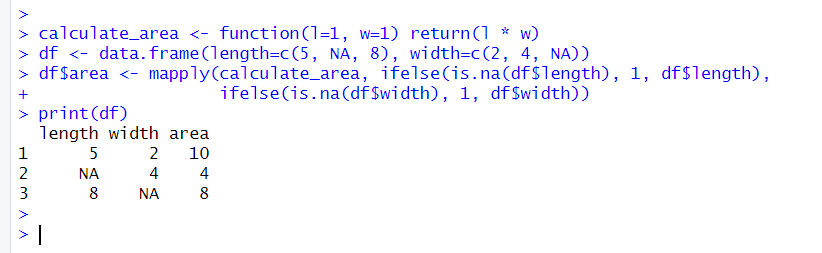
**67. Data Frame with Rectangle Areas**

**Aim:** To create a data frame and calculate the area of rectangles while handling missing values.

r

calculate\_area <- function(l=1, w=1) return(l \* w) df <- data.frame(length=c(5, NA, 8), width=c(2, 4, NA)) df$area <- mapply(calculate\_area, ifelse(is.na(df$length), 1, df$length), ifelse(is.na(df$width), 1, df$width)) print(df)

**Output:**

****

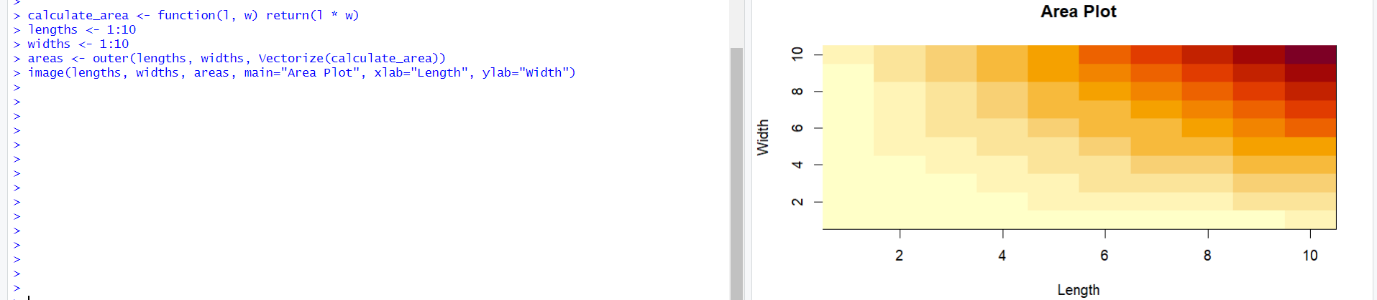
**68. Area Calculation and Plot**

**Aim:** To calculate areas of rectangles and plot them.

r

calculate\_area <- function(l, w) return(l \* w) lengths <- 1:10 widths <- 1:10 areas <- outer(lengths, widths, Vectorize(calculate\_area)) image(lengths, widths, areas, main="Area Plot", xlab="Length", ylab="Width")

**Output:**

****

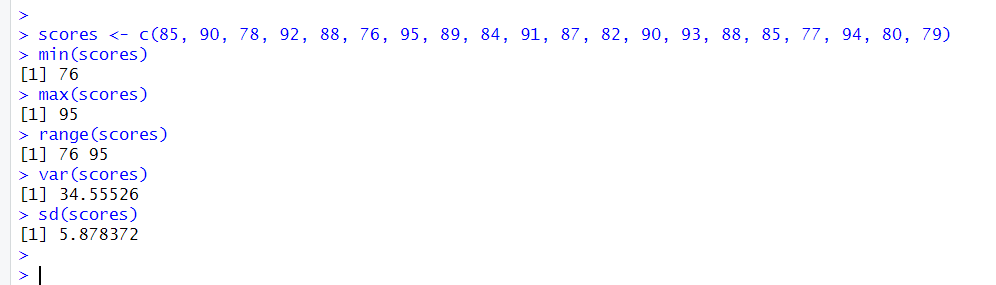
**69. Statistical Measures for Test Scores**

**Aim:** To find min, max, range, variance, and standard deviation.

r

scores <- c(85, 90, 78, 92, 88, 76, 95, 89, 84, 91, 87, 82, 90, 93, 88, 85, 77, 94, 80, 79) min(scores) max(scores) range(scores) var(scores) sd(scores)

**Output:**

****

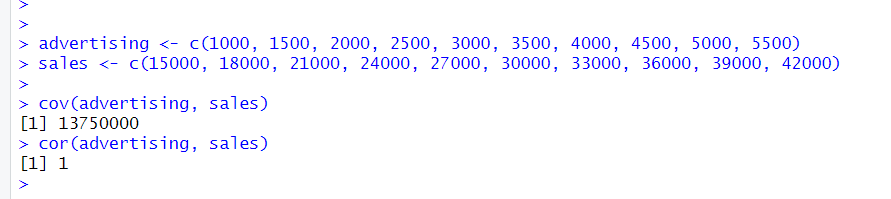
**70. Covariance and Correlation of Advertising and Sales**

**Aim:** To compute covariance and correlation between advertising expenditure and sales revenue.

R

advertising <- c(1000, 1500, 2000, 2500, 3000, 3500, 4000, 4500, 5000, 5500) sales <- c(15000, 18000, 21000, 24000, 27000, 30000, 33000, 36000, 39000, 42000) cov(advertising, sales) cor(advertising, sales)

**Output:**

****

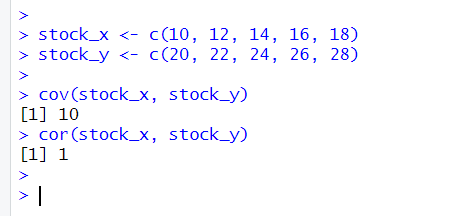
**71. Covariance and Correlation Between Stock Returns**

**Aim:** To compute covariance and correlation between Stock X and Stock Y.

r

stock\_x <- c(10, 12, 14, 16, 18) stock\_y <- c(20, 22, 24, 26, 28) cov(stock\_x, stock\_y) cor(stock\_x, stock\_y)

**Output:**

****

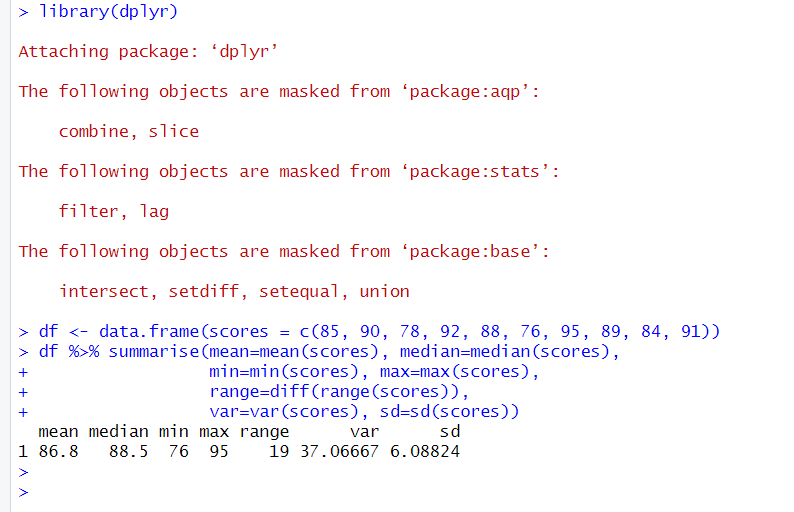
**72. Summary Statistics using dplyr**

**Aim:** To calculate multiple statistical measures using dplyr.

r

library(dplyr) df <- data.frame(scores = c(85, 90, 78, 92, 88, 76, 95, 89, 84, 91)) df %>% summarise(mean=mean(scores), median=median(scores), min=min(scores), max=max(scores), range=diff(range(scores)), var=var(scores), sd=sd(scores))

**Output:**

****

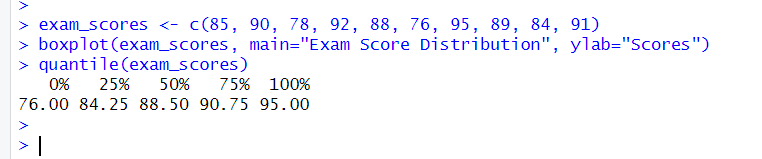
**73. Quartiles and Boxplot for Exam Scores**

**Aim:** To compute quartiles and visualize with a boxplot.

r

exam\_scores <- c(85, 90, 78, 92, 88, 76, 95, 89, 84, 91) boxplot(exam\_scores, main="Exam Score Distribution", ylab="Scores") quantile(exam\_scores)

**Output:**

****

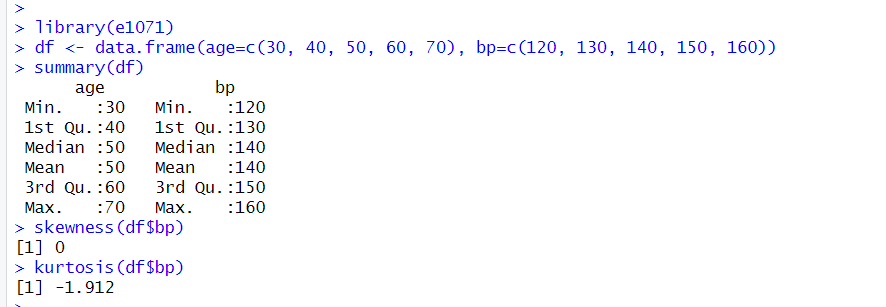
**74. EDA for Patients' Age and Blood Pressure**

**Aim:** To perform EDA on age and blood pressure data.

r

library(e1071) df <- data.frame(age=c(30, 40, 50, 60, 70), bp=c(120, 130, 140, 150, 160)) summary(df) skewness(df$bp) kurtosis(df$bp)

**Output:**

****

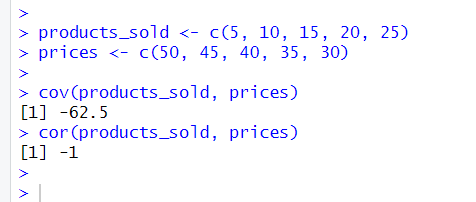
**75. Covariance and Correlation of Products Sold and Prices**

**Aim:** To compute covariance and correlation between products sold and prices.

r

products\_sold <- c(5, 10, 15, 20, 25) prices <- c(50, 45, 40, 35, 30) cov(products\_sold, prices) cor(products\_sold, prices)

**Output:**

****

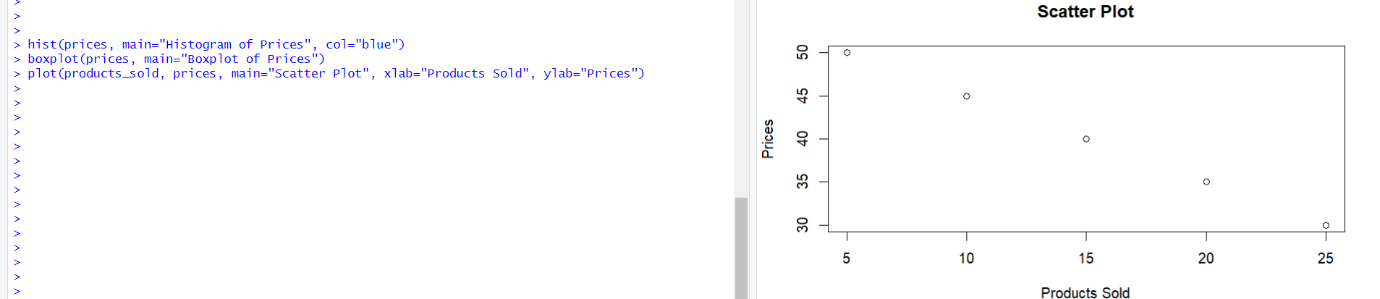
**76. Data Visualization with Multiple Plots**

**Aim:** To create different plots to explore data relationships.

r

hist(prices, main="Histogram of Prices", col="blue") boxplot(prices, main="Boxplot of Prices") plot(products\_sold, prices, main="Scatter Plot", xlab="Products Sold", ylab="Prices")

**Output:**



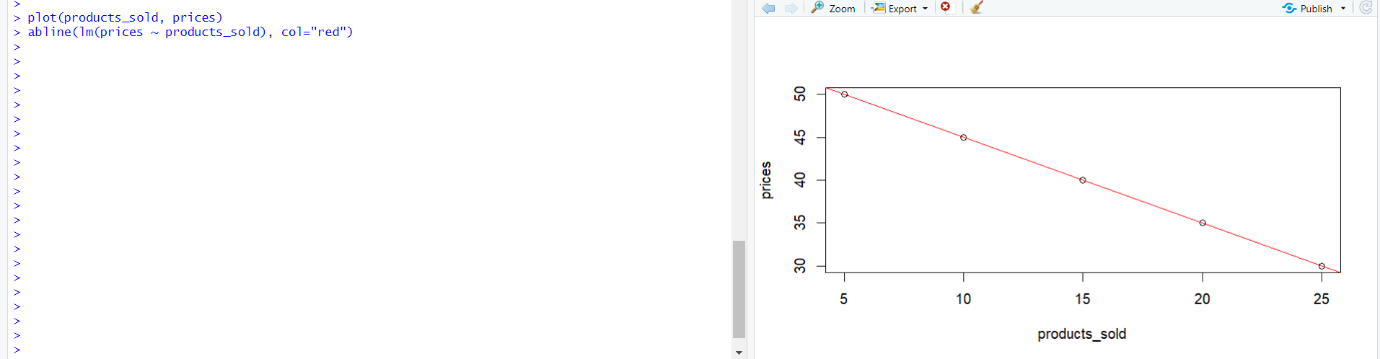
**77. Covariance, Correlation, and Scatter Plot with Trend Line**

**Aim:** To analyze relationships between two variables.

r

plot(products\_sold, prices) abline(lm(prices ~ products\_sold), col="red")

**Output:**



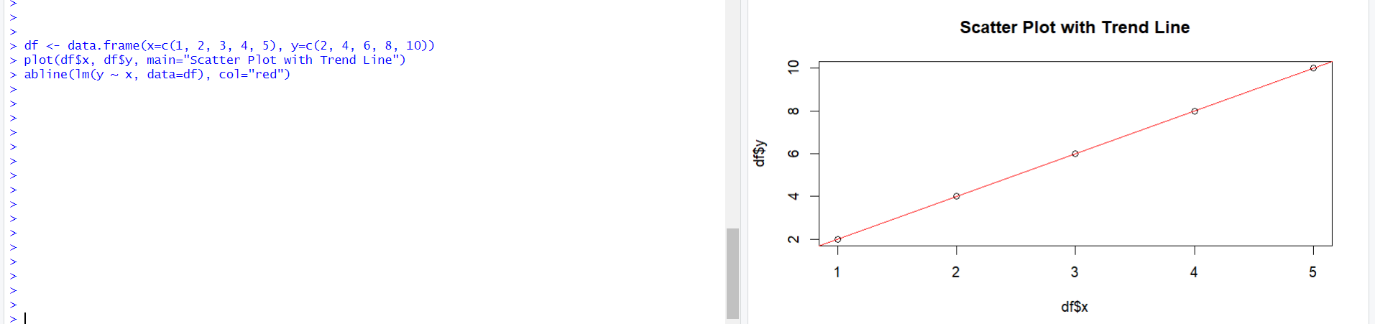
**78. Scatter Plot with Trend Line**

**Aim:** To visualize relationships using a scatter plot.

r

df <- data.frame(x=c(1, 2, 3, 4, 5), y=c(2, 4, 6, 8, 10)) plot(df$x, df$y, main="Scatter Plot with Trend Line") abline(lm(y ~ x, data=df), col="red")

**Output:**

****

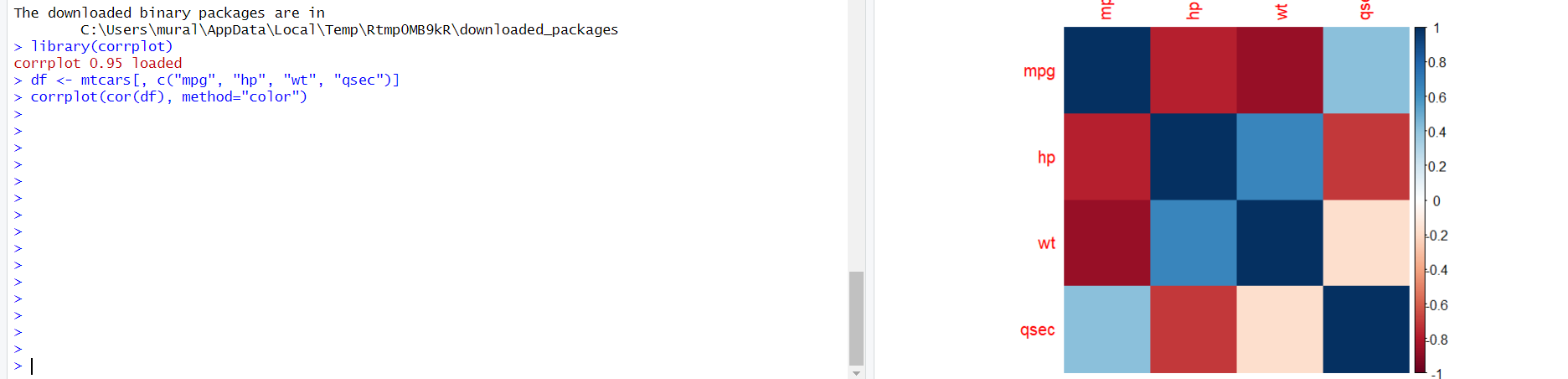
**79. Correlation Matrix with Heatmap**

**Aim:** To compute and visualize a correlation matrix.

r

library(corrplot) df <- mtcars[, c("mpg", "hp", "wt", "qsec")] corrplot(cor(df), method="color")

**Output:**



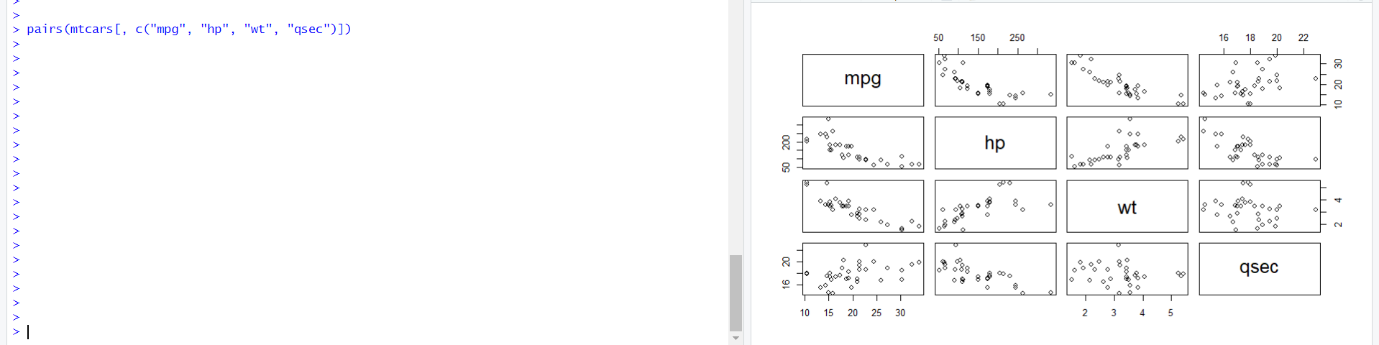
**80. Pair Plot for Dataframe**

**Aim:** To create a scatter plot matrix.

r

pairs(mtcars[, c("mpg", "hp", "wt", "qsec")])

**Output:**

****