Data Science Laboratory – R Programming

Subject Code: DMC6311

|  |  |
| --- | --- |
| Name | Haryish E |
| Degree and Batch | MCA – Batch 1 |
| Roll No | 2232MCA0058 |

INDEX

[1. Calculate the number if occurrence of a character in each row of R DataFrameCalculate the number if occurrence of a character in each row of R DataFrame. 2](#_Toc159693941)

[2. Find row and column index of the maximum and minimum value in a matrix in R 4](#_Toc159693942)

[3: Multiply a matrix by its transpose while ignore missing values in R. 6](#_Toc159693943)

[4. Remove rows with NA in one column of R DataFrame. 7](#_Toc159693944)

[5. How to find common rows and columns between two dataframe in R? 8](#_Toc159693945)

[6. Adding Colors to Charts in R Programming. 10](#_Toc159693946)

[7. Create a Stacked Dot Plot in R. 11](#_Toc159693947)

[8. Plot Shaded Area between vertical lines in R. 12](#_Toc159693948)

[9. Add Mean and Median to Histogram in. 13](#_Toc159693949)

[10. Create Scatter plot from CSV in R. 14](#_Toc159693950)

# 1. Calculate the number if occurrence of a character in each row of R DataFrameCalculate the number if occurrence of a character in each row of R DataFrame.

**Program**:

# Sample DataFrame

df <- data.frame(

ID = 1:5,

Text = c("hello", "world", "foo", "bar", "baz")

)

# Character to count

char\_to\_count <- "o"

# Function to count occurrences of a character in a string

count\_char <- function(text, char) {

sum(strsplit(text, "")[[1]] == char)

}

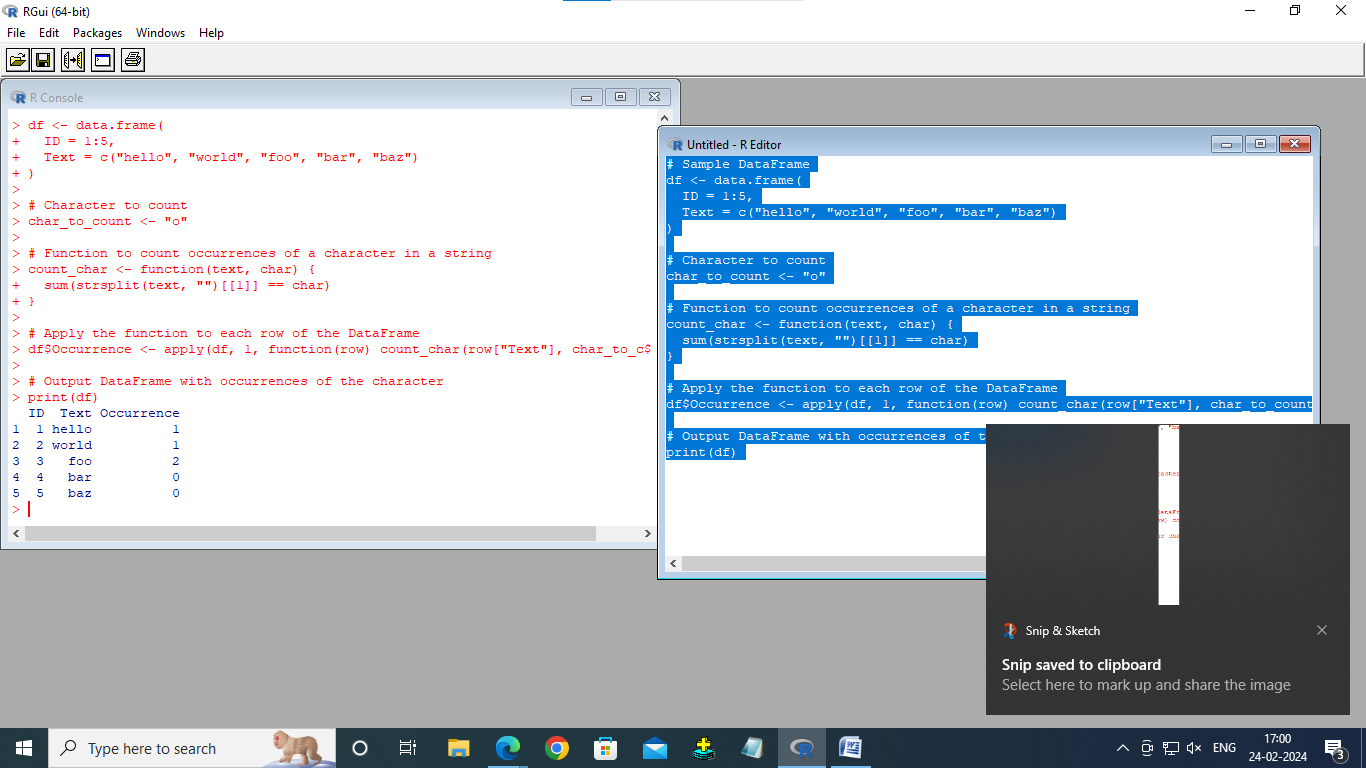
# Apply the function to each row of the DataFrame

df$Occurrence <- apply(df, 1, function(row) count\_char(row["Text"], char\_to\_count))

# Output DataFrame with occurrences of the character

print(df)

**Output**:



# 2. Find row and column index of the maximum and minimum value in a matrix in R

**Program**:

# Create a sample matrix

mat <- matrix(c(1, 2, 3, 4, 5, 6, 7, 8, 9), nrow = 3, byrow = TRUE)

print(mat)

# Find the row and column index of the maximum value

max\_index <- which(mat == max(mat), arr.ind = TRUE)

max\_row <- max\_index[1, 1]

max\_col <- max\_index[1, 2]

# Find the row and column index of the minimum value

min\_index <- which(mat == min(mat), arr.ind = TRUE)

min\_row <- min\_index[1, 1]

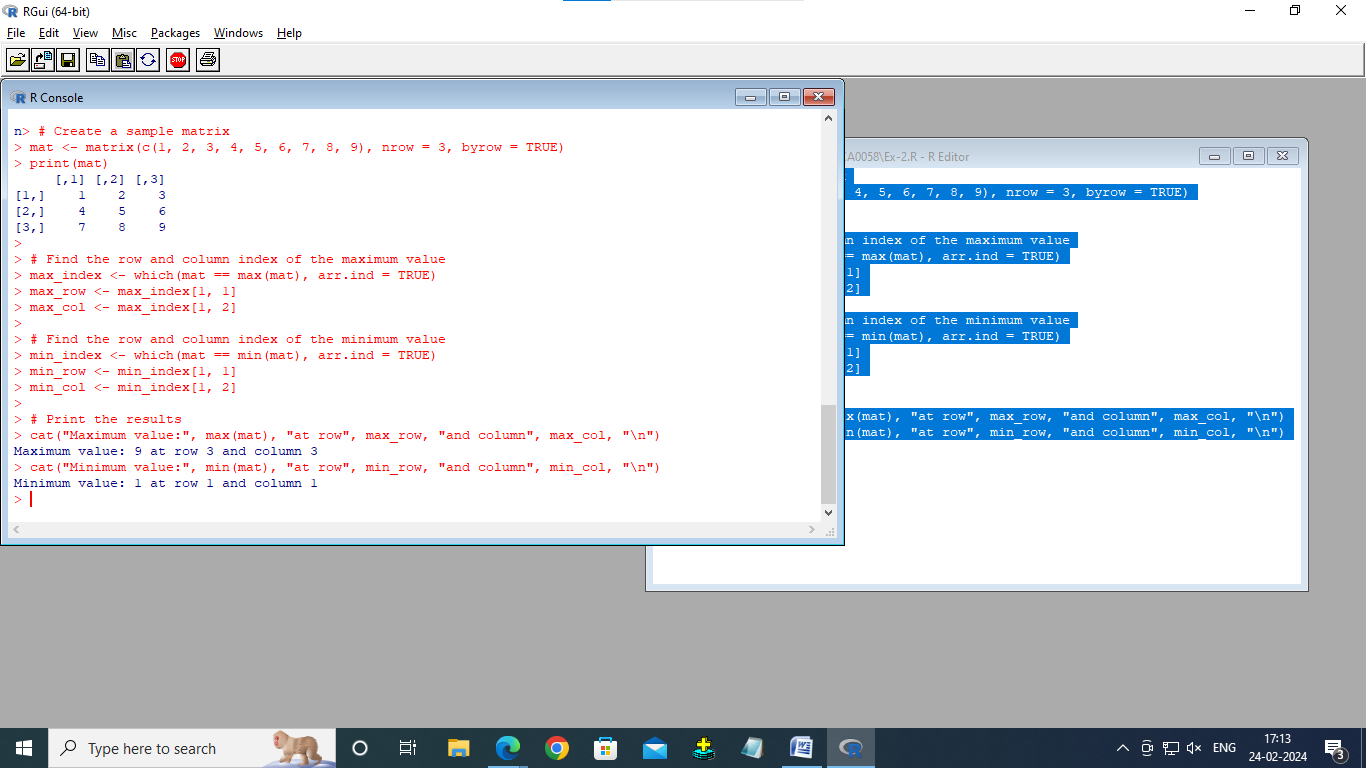
min\_col <- min\_index[1, 2]

# Print the results

cat("Maximum value:", max(mat), "at row", max\_row, "and column", max\_col, "\n")

cat("Minimum value:", min(mat), "at row", min\_row, "and column", min\_col, "\n")

**Output**:



# 3: Multiply a matrix by its transpose while ignore missing values in R.

**Program**:

multiply\_matrix\_transpose <- function(mat) {

mat <- na.omit(mat)

result <- mat %\*% t(mat)

return(result)

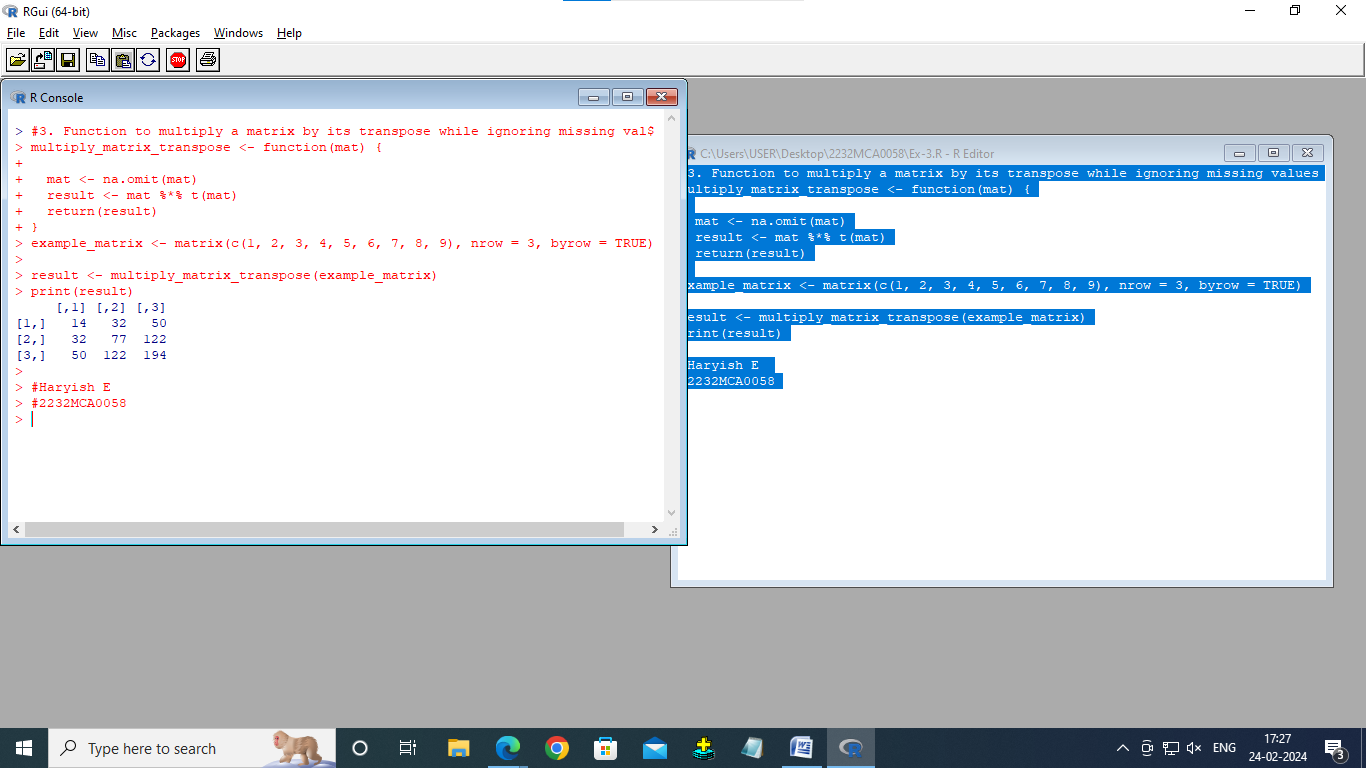
}

example\_matrix <- matrix(c(1, 2, 3, 4, 5, 6, 7, 8, 9), nrow = 3, byrow = TRUE)

result <- multiply\_matrix\_transpose(example\_matrix)

print(result)

**Output**:



# 4. Remove rows with NA in one column of R DataFrame.

**Program**:

# Create a sample DataFrame

df <- data.frame(

col1 = c(1, 2, NA, 4),

col2 = c(5, NA, 7, 8)

)

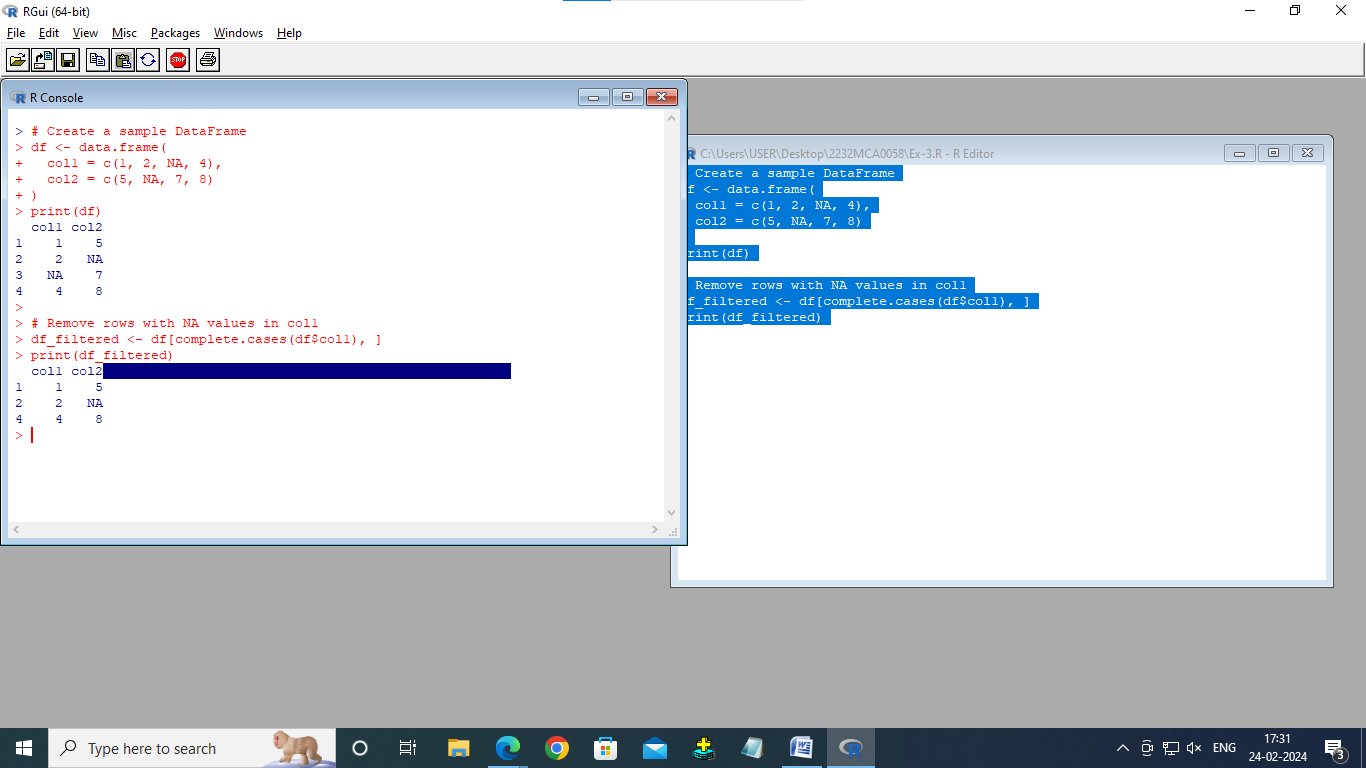
print(df)

# Remove rows with NA values in col1

df\_filtered <- df[complete.cases(df$col1), ]

print(df\_filtered)

**Output**:



# 5. How to find common rows and columns between two dataframe in R?

**Program**:

# Create sample dataframes

df1 <- data.frame(A = c(1, 2, 3), B = c(4, 5, 6))

df2 <- data.frame(A = c(2, 3, 4), B = c(5, 6, 7))

# Find common rows

common\_rows <- intersect(rownames(df1), rownames(df2))

# Find common columns

common\_cols <- intersect(names(df1), names(df2))

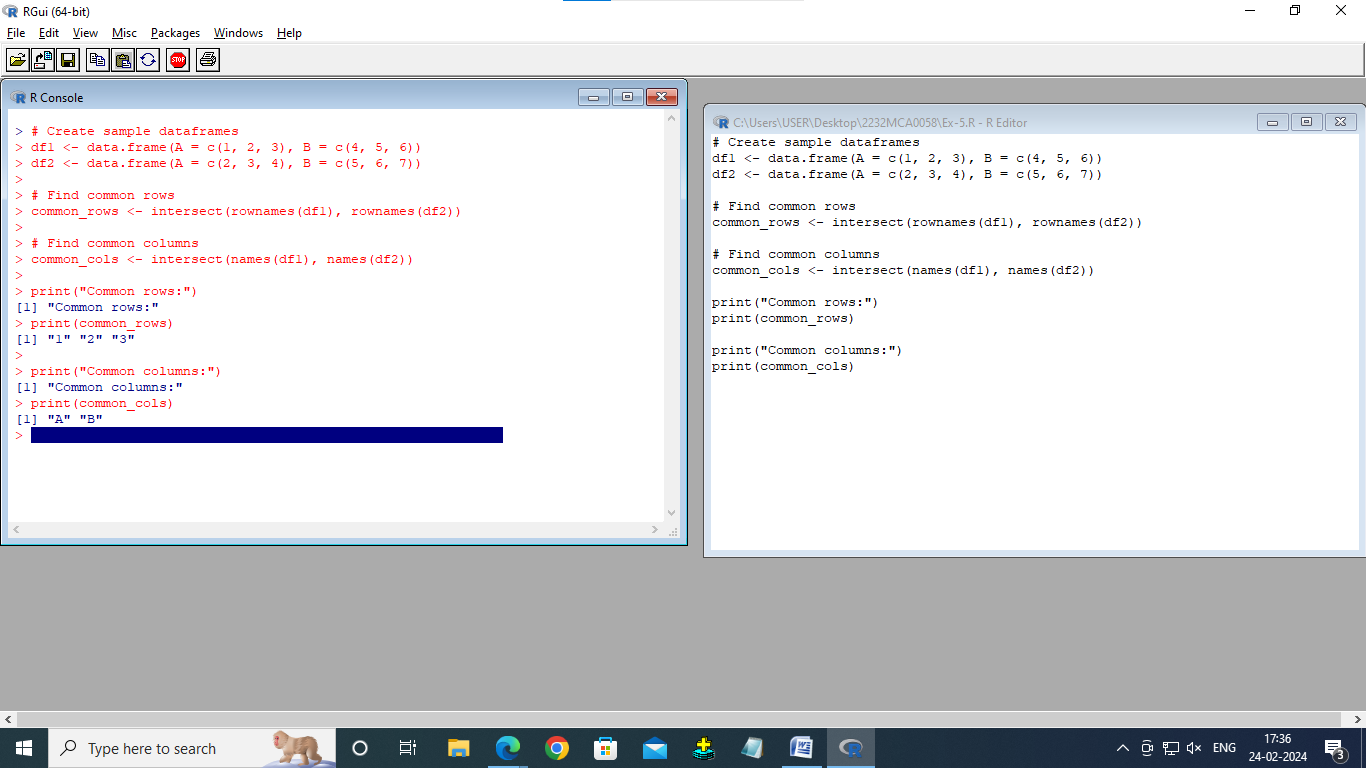
print("Common rows:")

print(common\_rows)

print("Common columns:")

print(common\_cols)

**Output**:



# 6. Adding Colors to Charts in R Programming.

**Program**:

# Create a simple plot with colors

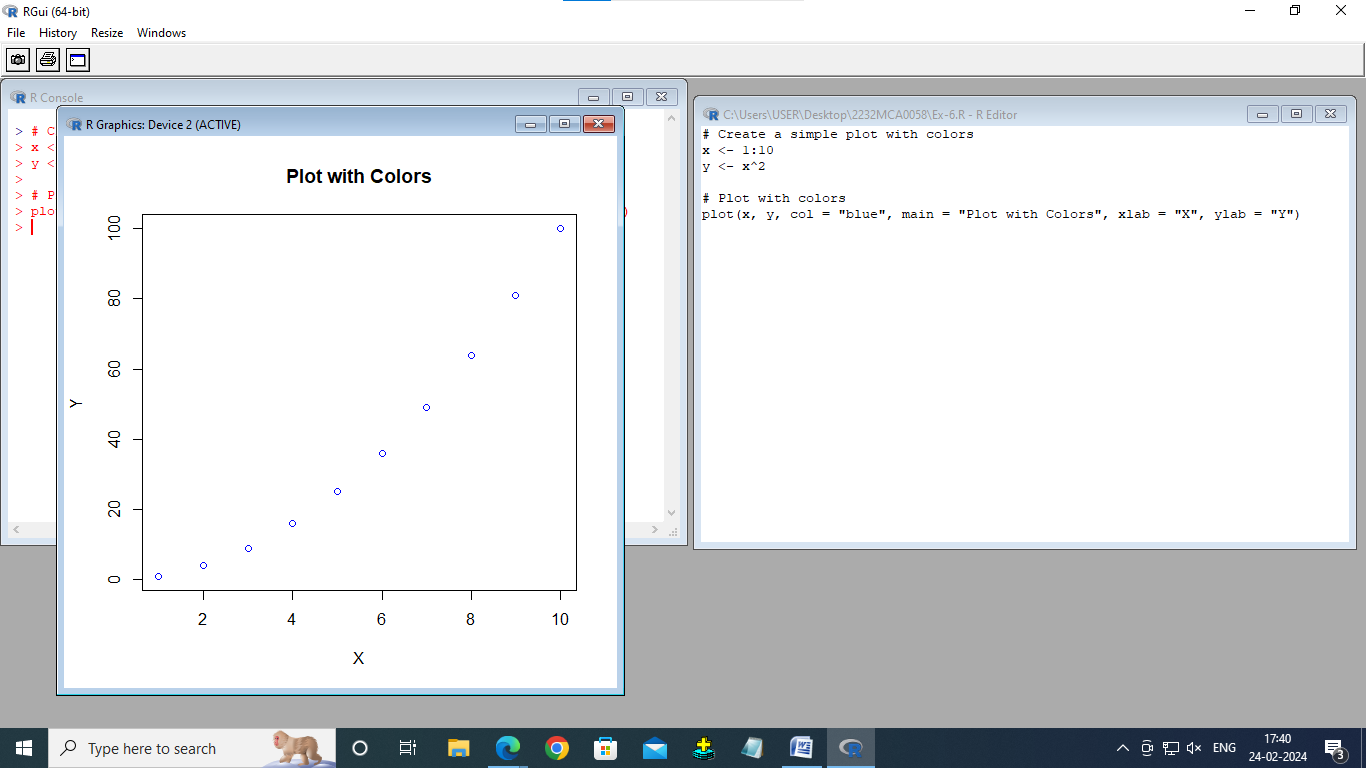
x <- 1:10

y <- x^2

# Plot with colors

plot(x, y, col = "blue", main = "Plot with Colors", xlab = "X", ylab = "Y")

**Output**:



# 7. Create a Stacked Dot Plot in R.

**Program**:

# Create sample data

data <- data.frame(Category = c("A", "B", "C"),

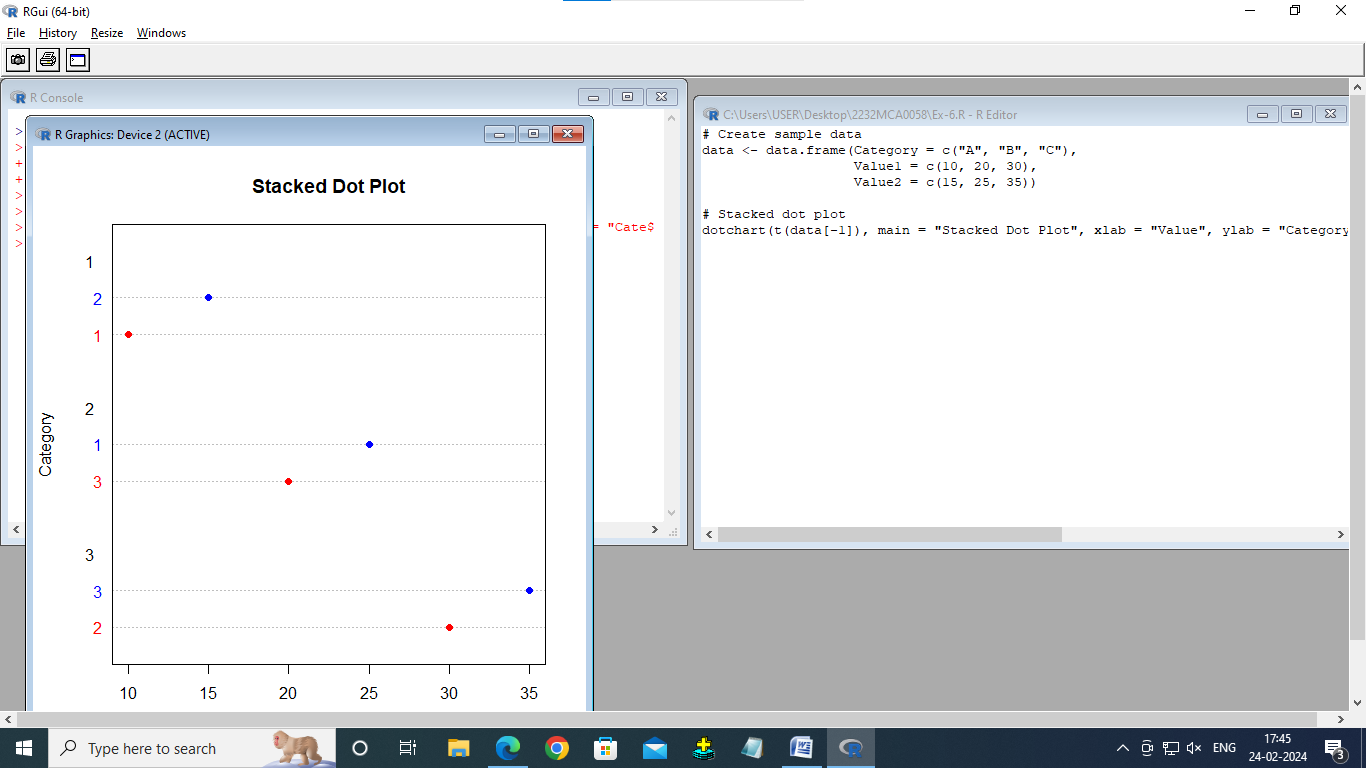
Value1 = c(10, 20, 30),

Value2 = c(15, 25, 35))

# Stacked dot plot

dotchart(t(data[-1]), main = "Stacked Dot Plot", xlab = "Value", ylab = "Category", color = c("red", "blue"), pch = 19, labels = rownames(data))

**Output**:



# 8. Plot Shaded Area between vertical lines in R.

Program:

# Create sample data

x <- 1:10

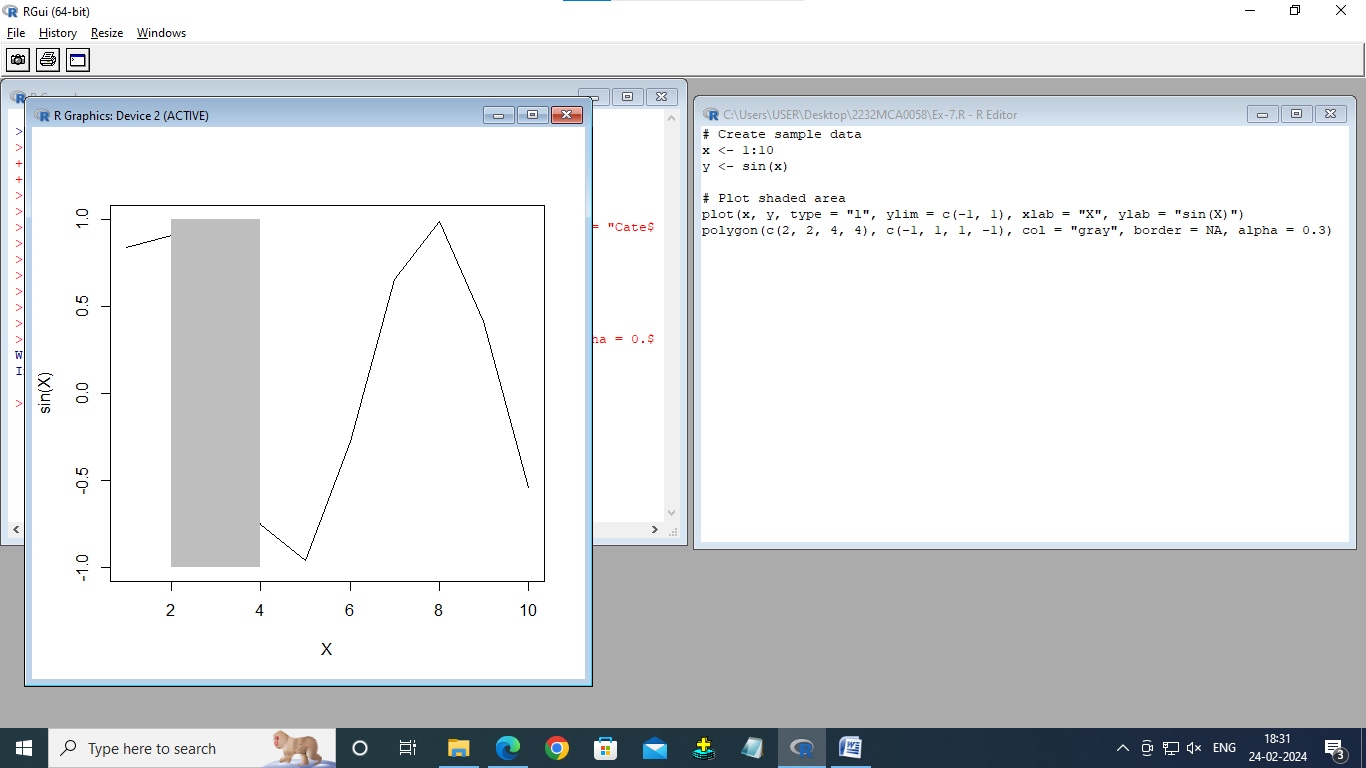
y <- sin(x)

# Plot shaded area

plot(x, y, type = "l", ylim = c(-1, 1), xlab = "X", ylab = "sin(X)")

polygon(c(2, 2, 4, 4), c(-1, 1, 1, -1), col = "gray", border = NA, alpha = 0.3)

Output:



# 9. Add Mean and Median to Histogram in.

**Program:**

# Create sample data

data <- rnorm(100)

# Histogram

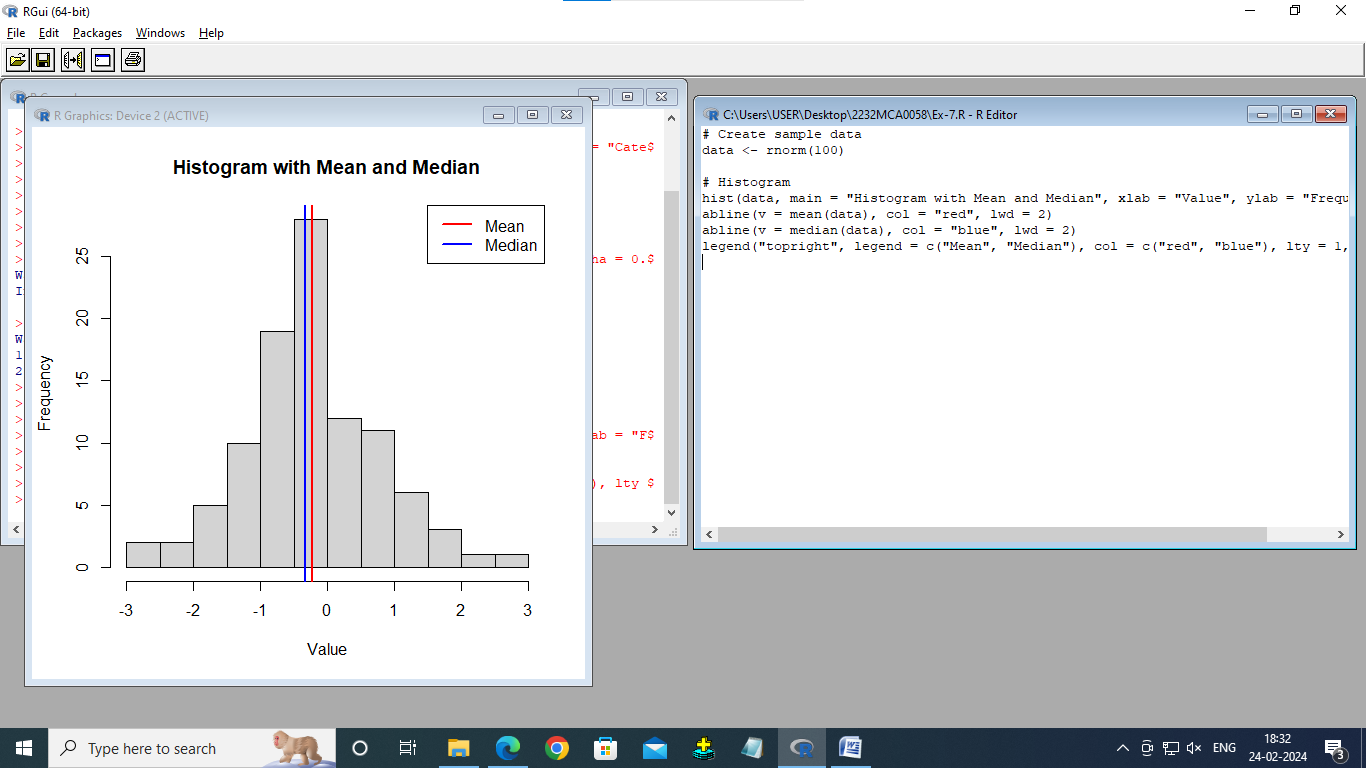
hist(data, main = "Histogram with Mean and Median", xlab = "Value", ylab = "Frequency")

abline(v = mean(data), col = "red", lwd = 2)

abline(v = median(data), col = "blue", lwd = 2)

legend("topright", legend = c("Mean", "Median"), col = c("red", "blue"), lty = 1, lwd = 2)

**OutPut:**



# 10. Create Scatter plot from CSV in R.

**Program:**

data <- read.csv("C:\\Users\\USER\\Desktop\\2232MCA0058\\data-Ex-10.csv")

# Create scatter plot

plot(data$X, data$Y,

main = "Scatter Plot",

xlab = "X-axis",

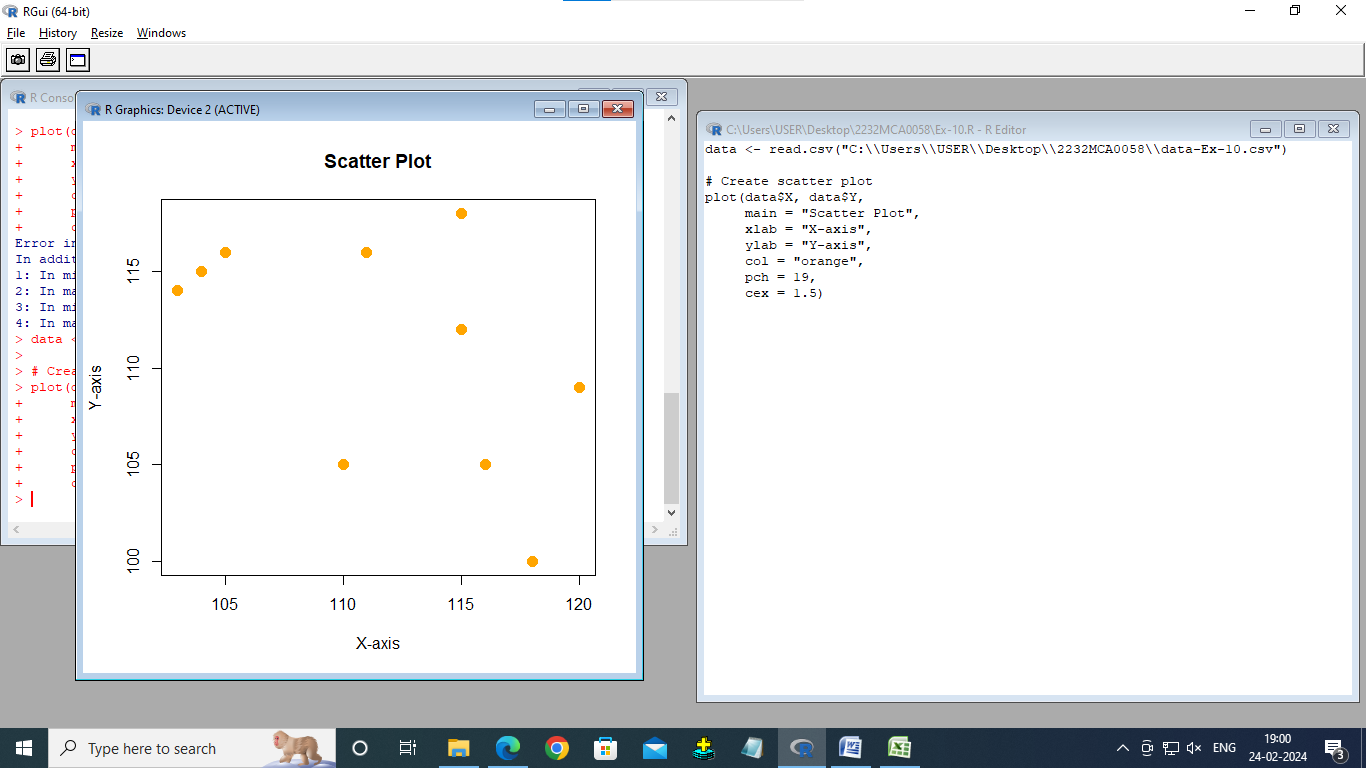
ylab = "Y-axis",

col = "orange",

pch = 19,

cex = 1.5)

**Output:**

****