**SAVEETHA SCHOOL OF ENGINEERING**

**SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES**

**ITA 0451 - STATISTICS WITH R PROGRAMMING**

**DAY 4 – LAB ASSESSMENT Part 4**

**Reg No:192011216**

**Name:Vamshi.R**

**1.   a. Create multi regression model to find a weight of the chicken , by “Time” and “Diet” as  as  
 predictor variables  
          b. Predict weight for Time=10 and Diet=1  
           c. Find the error in model for same**

**Code:**

# Assuming you have a dataset with weight, time, and diet variables

# Step 1: Load the dataset

data <- read.csv("chicken\_data.csv") # Replace "chicken\_data.csv" with your dataset file name

# Step 2: Create the regression model

model <- lm(weight ~ time + diet, data=data)

# Step 3: View the model summary

summary(model)

# Step 4: Predict weight for Time=10 and Diet=1

time\_new <- 10

diet\_new <- 1

new\_data <- data.frame(time=time\_new, diet=diet\_new)

weight\_pred <- predict(model, newdata=new\_data)

# Step 5: Find the error in the model for the same prediction

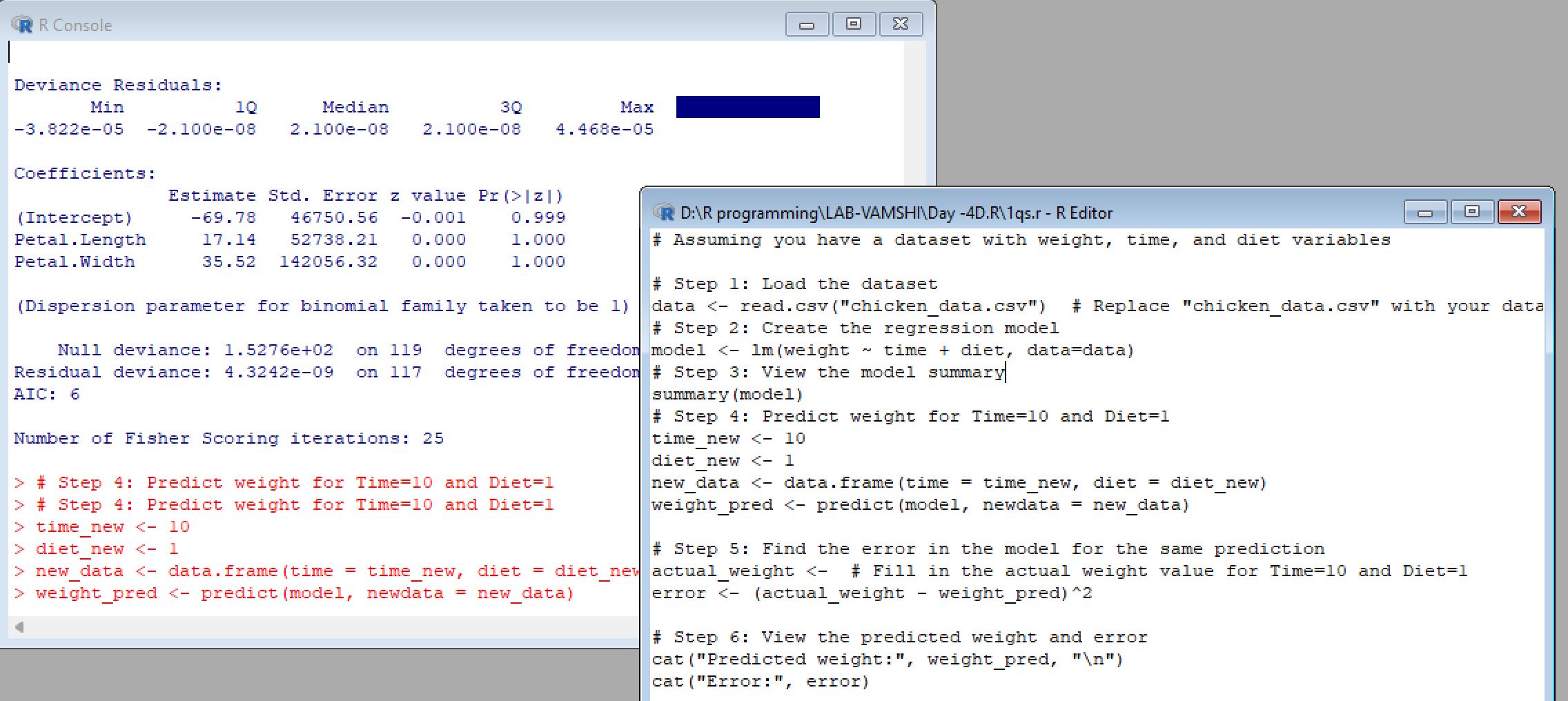
actual\_weight <- # Fill in the actual weight value for Time=10 and Diet=1

error <- (actual\_weight - weight\_pred)^2

# Step 6: View the predicted weight and error

cat("Predicted weight:", weight\_pred, "\n")

cat("Error:", error)



**2 .For this exercise, use the (built-in) dataset Titanic.  
    a. Draw a Bar chart to show details of “Survived” on the Titanic based on passenger Class  
    b. Modify the above plot based on gender of people who survived  
   c. Draw histogram plot to show distribution of feature “Age”**

**Code:**

# Step 1: Load the Titanic dataset

data(Titanic)

# Step 2a: Draw a Bar chart to show details of "Survived" based on passenger Class

survived\_by\_class <- xtabs(Freq ~ Class + Survived, data = Titanic)

barplot(survived\_by\_class, beside = TRUE, legend = TRUE,

col = c("red", "green"),

names.arg = c("1st Class", "2nd Class", "3rd Class"),

main = "Survival on Titanic by Passenger Class",

xlab = "Passenger Class", ylab = "Count")

# Step 2b: Modify the above plot based on gender of people who survived

survived\_by\_class\_gender <- xtabs(Freq ~ Class + Sex + Survived, data = Titanic)

barplot(survived\_by\_class\_gender, beside = TRUE, legend = TRUE,

col = c("red", "green", "blue", "yellow"),

names.arg = c("1st Class\nFemale", "2nd Class\nFemale", "3rd Class\nFemale",

"1st Class\nMale", "2nd Class\nMale", "3rd Class\nMale"),

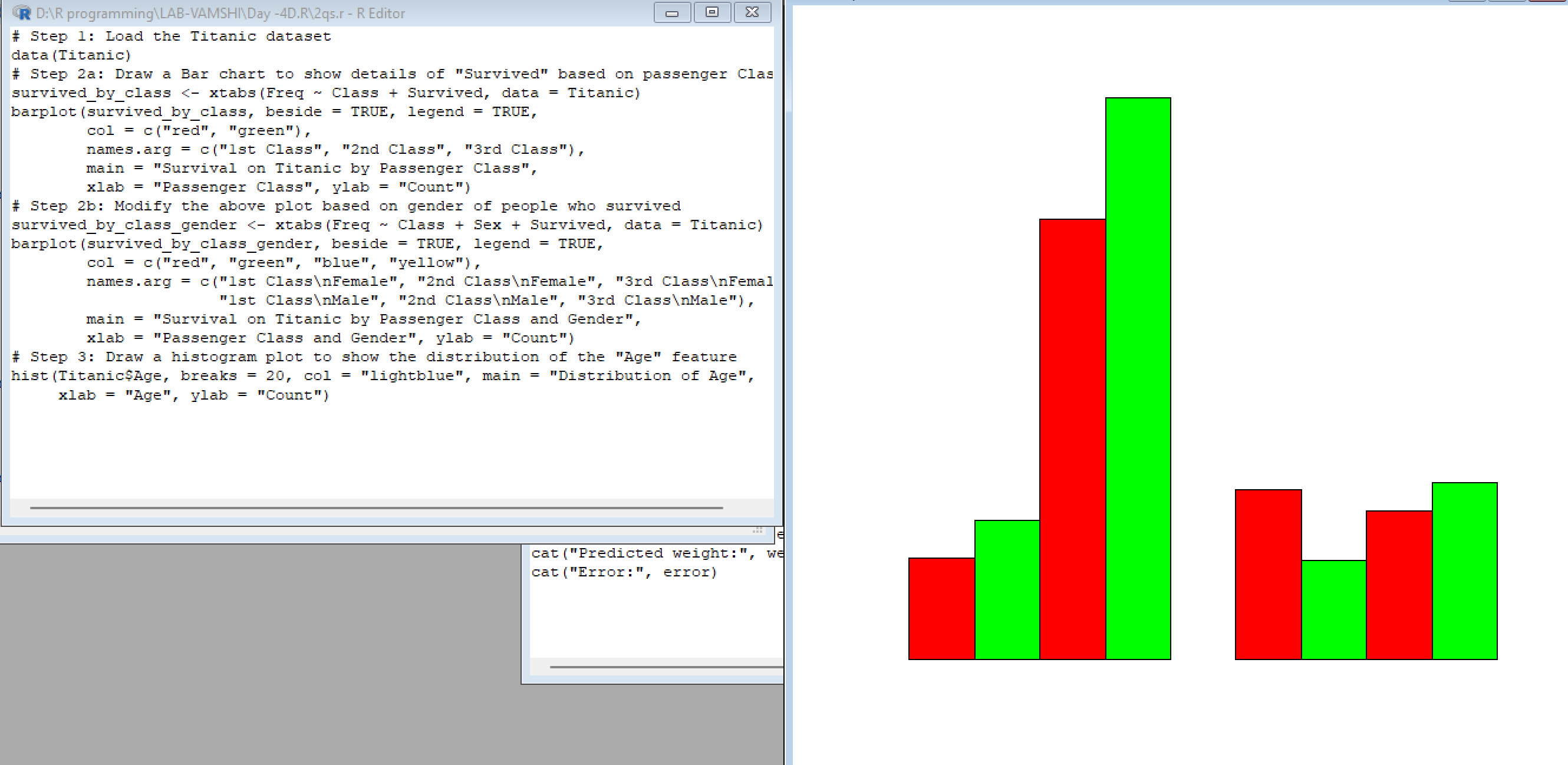
main = "Survival on Titanic by Passenger Class and Gender",

xlab = "Passenger Class and Gender", ylab = "Count")

# Step 3: Draw a histogram plot to show the distribution of the "Age" feature

hist(Titanic$Age, breaks = 20, col = "lightblue", main = "Distribution of Age",

xlab = "Age", ylab = "Count")



**3. Explore the USArrests dataset, contains the number of arrests for murder, assault, and rape for each of the 50 states in 1973. It also contains the percentage of people in the state who live in an urban area.   
 (i) a. Explore the summary of Data set, like number of Features and its type. Find the number         of records for each feature. Print the statistical feature of data  
         b. Print the state which saw the largest total number of rape  
         c. Print the states with the max & min crime rates for murder  
    (ii).a. Find the correlation among the features  
     b. Print the states which have assault arrests more than median of the country  
     c. Print the states are in the bottom 25% of murder  
   (iii). a. Create a histogram and density plot of murder arrests by US stat  
 b. Create the plot that shows the relationship between murder arrest rate and  proportion**

**of the population that is urbanised by state. Then enrich the chart by adding assault**

**arrest rates (by colouring the points from blue (low) to red (high)).  
         c. Draw a bar graph to show the murder rate for each of the 50 states .**

**code:**

# Step 1: Load the USArrests dataset

data(USArrests)

# (i) a. Explore the summary of the dataset

summary(USArrests)

str(USArrests)

sapply(USArrests, length)

# (i) b. Print the state with the largest total number of rape

largest\_rape\_state <- names(USArrests$Rape)[which.max(USArrests$Rape)]

cat("State with the largest total number of rape:", largest\_rape\_state, "\n")

# (i) c. Print the states with the max & min crime rates for murder

max\_murder\_state <- names(USArrests$Murder)[which.max(USArrests$Murder)]

min\_murder\_state <- names(USArrests$Murder)[which.min(USArrests$Murder)]

cat("State with the maximum crime rate for murder:", max\_murder\_state, "\n")

cat("State with the minimum crime rate for murder:", min\_murder\_state, "\n")

# (ii) a. Find the correlation among the features

correlation <- cor(USArrests)

print(correlation)

# (ii) b. Print the states which have assault arrests more than the median of the country

median\_assault <- median(USArrests$Assault)

states\_more\_than\_median <- names(USArrests$Assault)[USArrests$Assault > median\_assault]

cat("States with assault arrests more than the median of the country:", states\_more\_than\_median, "\n")

# (ii) c. Print the states in the bottom 25% of murder

murder\_percentile <- quantile(USArrests$Murder, probs = 0.25)

states\_bottom\_25 <- names(USArrests$Murder)[USArrests$Murder < murder\_percentile]

cat("States in the bottom 25% of murder:", states\_bottom\_25, "\n")

# (iii) a. Create a histogram and density plot of murder arrests by US state

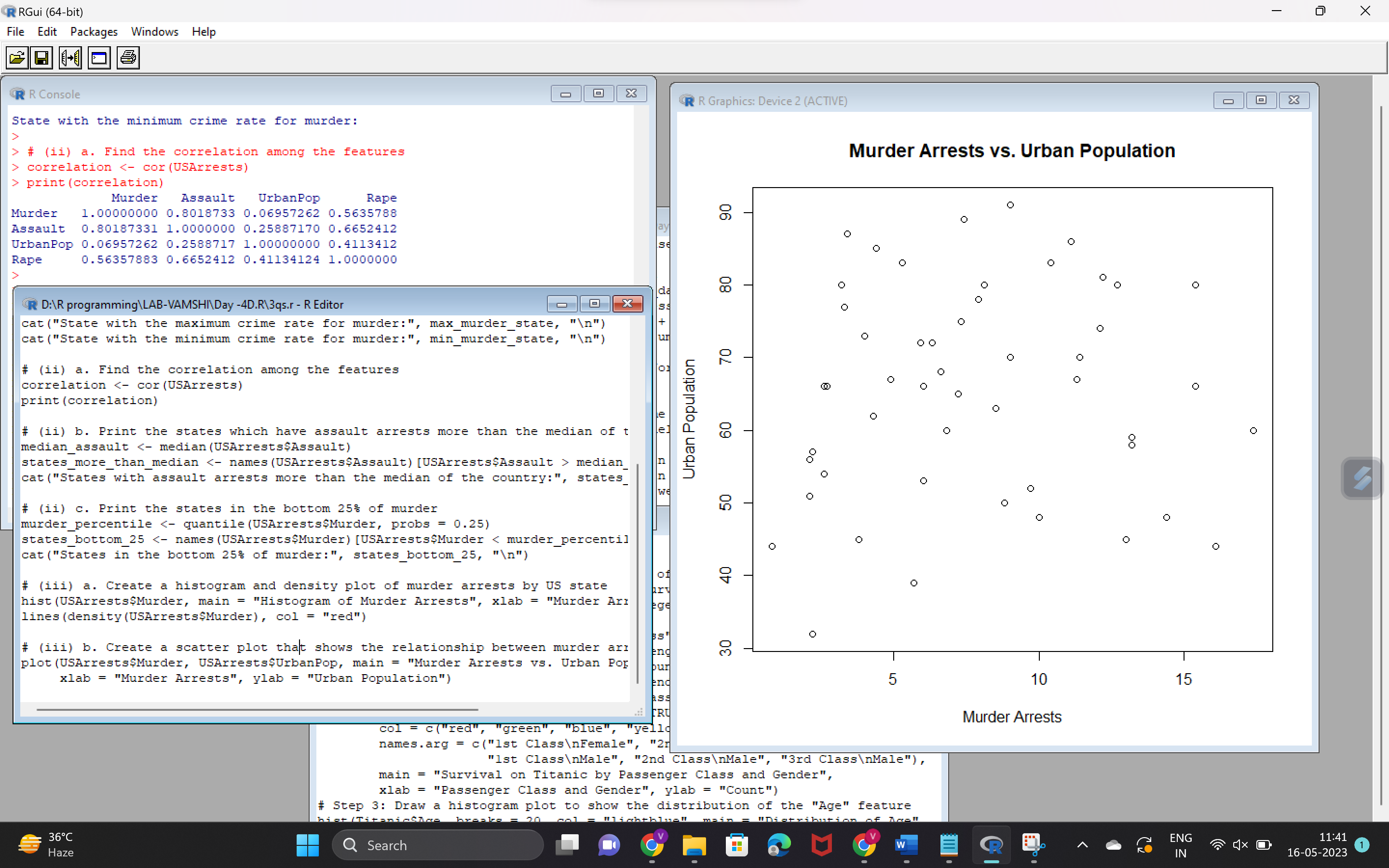
hist(USArrests$Murder, main = "Histogram of Murder Arrests", xlab = "Murder Arrests", ylab = "Frequency")

lines(density(USArrests$Murder), col = "red")

# (iii) b. Create a scatter plot that shows the relationship between murder arrest rate and proportion

plot(USArrests$Murder, USArrests$UrbanPop, main = "Murder Arrests vs. Urban Population",

xlab = "Murder Arrests", ylab = "Urban Population")

****

**4. a. Create a data frame based on below table.**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Month** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** |
| **Spends** | **1000** | **4000** | **5000** | **4500** | **3000** | **4000** | **9000** | **11000** | **15000** | **12000** | **7000** | **3000** |
| **Sales** | **9914** | **40487** | **54324** | **50044** | **34719** | **42551** | **94871** | **118914** | **158484** | **131348** | **78504** | **36284** |

**b. Create a regression model for that data frame table to show the amount of sales(Sales) based on the how much the company spends (Spends) in advertising**

**c. Predict the Sales if Spend=13500**

**CODE:**

# Create the data frame

Month <- 1:12

Spends <- c(1000, 4000, 5000, 4500, 3000, 4000, 9000, 11000, 15000, 12000, 7000, 3000)

Sales <- c(9914, 40487, 54324, 50044, 34719, 42551, 94871, 118914, 158484, 131348, 78504, 36284)

df <- data.frame(Month, Spends, Sales)

# Create the regression model

model <- lm(Sales ~ Spends, data = df)

# Print the model summary

summary(model)

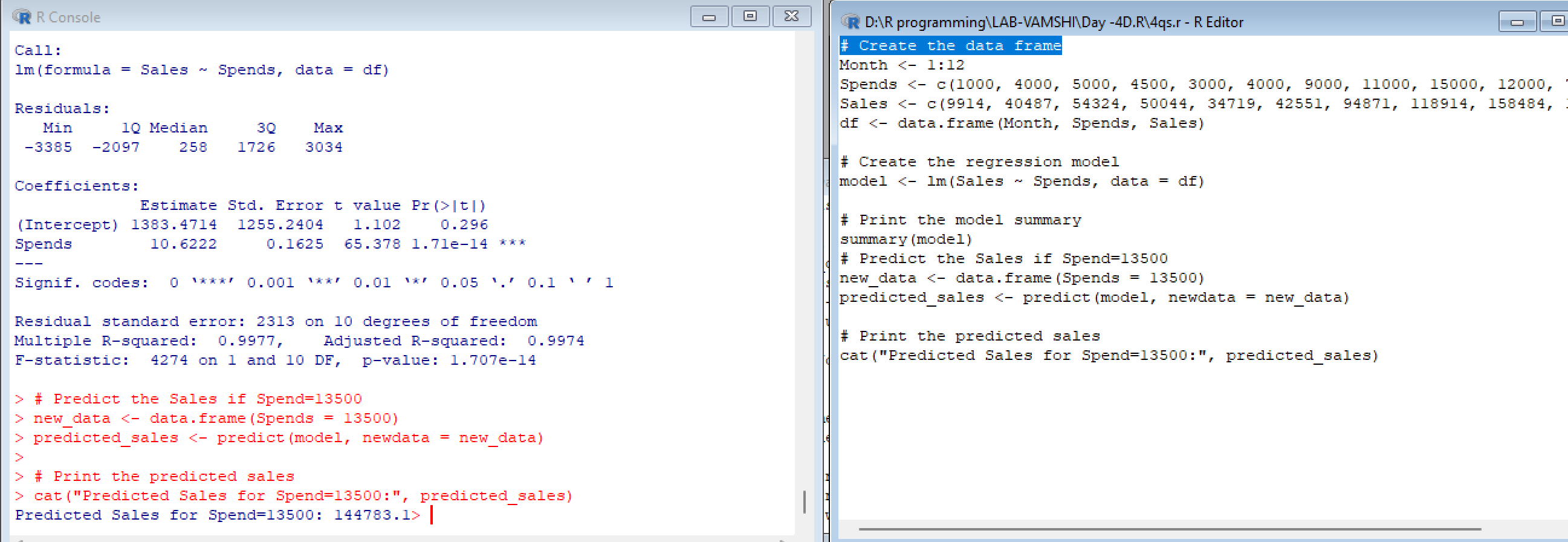
# Predict the Sales if Spend=13500

new\_data <- data.frame(Spends = 13500)

predicted\_sales <- predict(model, newdata = new\_data)

# Print the predicted sales

cat("Predicted Sales for Spend=13500:", predicted\_sales)



**5.(i) Write a R program to extract the five of the levels of factor created from a random sample from the LETTERS (Part of the base R distribution.)**

# Create a random sample from LETTERS

set.seed(123)

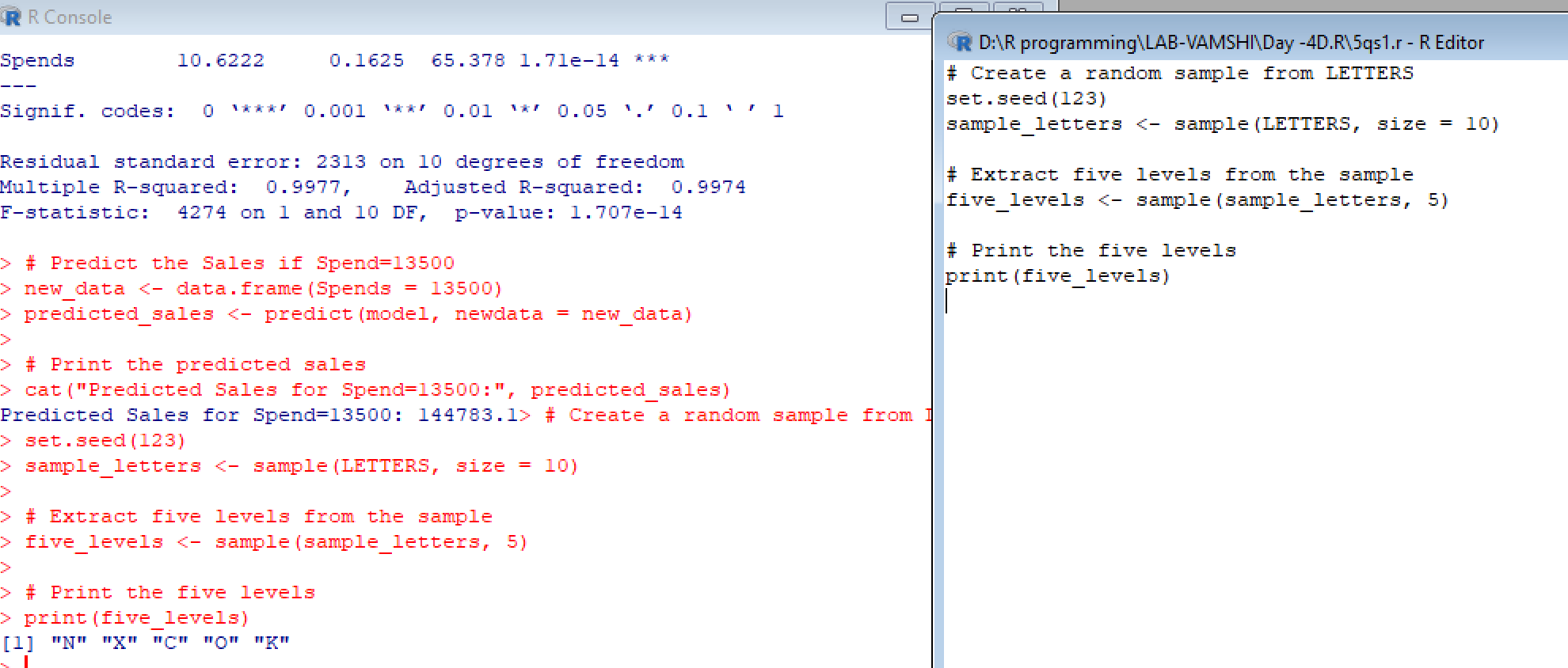
sample\_letters <- sample(LETTERS, size = 10)

# Extract five levels from the sample

five\_levels <- sample(sample\_letters, 5)

# Print the five levels

print(five\_levels)

  
  
  
**(ii)Write R function to find the range of given vector. Range=Max-Min  
Sample input, C<-(9,8,7,6,5,4,3,2,1),  
output=8**

find\_range <- function(vec) {

max\_val <- max(vec)

min\_val <- min(vec)

range\_val <- max\_val - min\_val

return(range\_val)

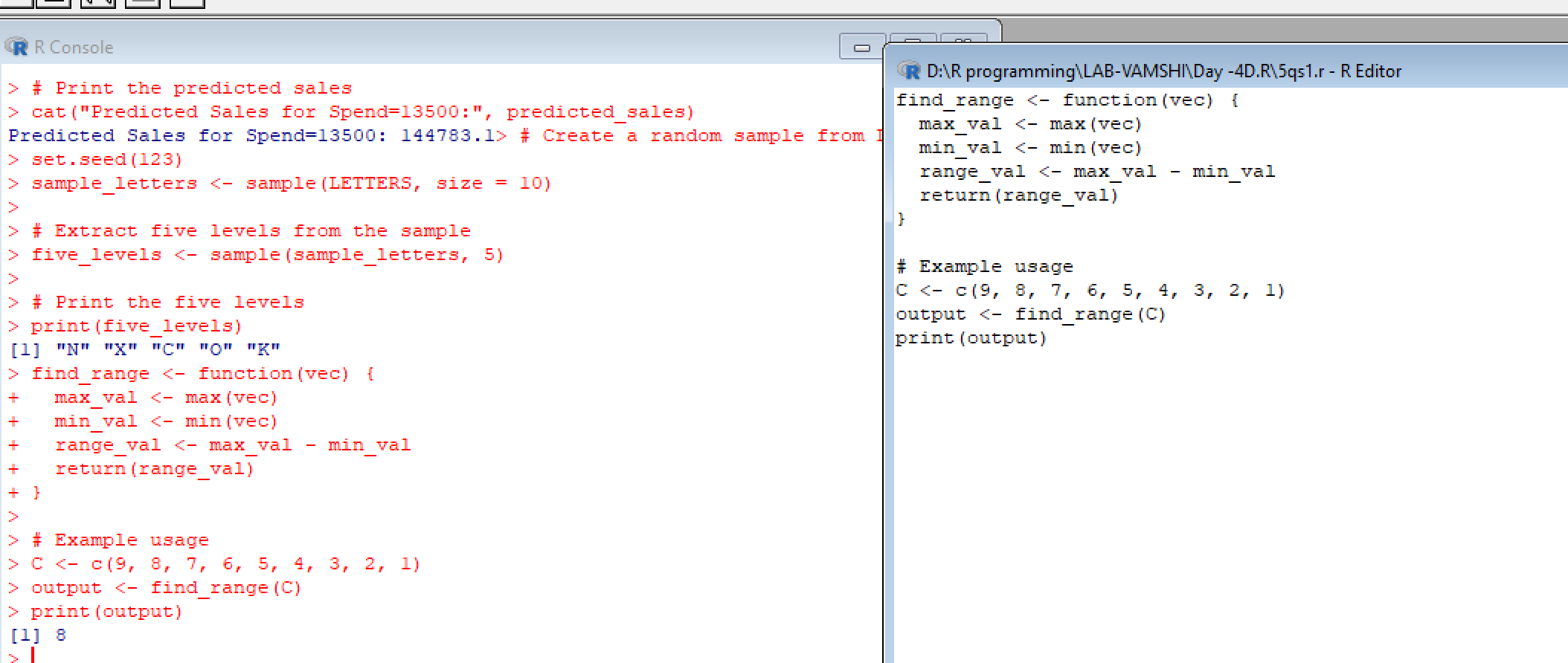
}

# Example usage

C <- c(9, 8, 7, 6, 5, 4, 3, 2, 1)

output <- find\_range(C)

print(output)

  
**(iii)Wirte the R function to find the number of vowels in given string  
 Sample input c<- “matrix”,  output<-2**

count\_vowels <- function(str) {

vowels <- c("a", "e", "i", "o", "u")

count <- sum(strsplit(tolower(str), "")[[1]] %in% vowels)

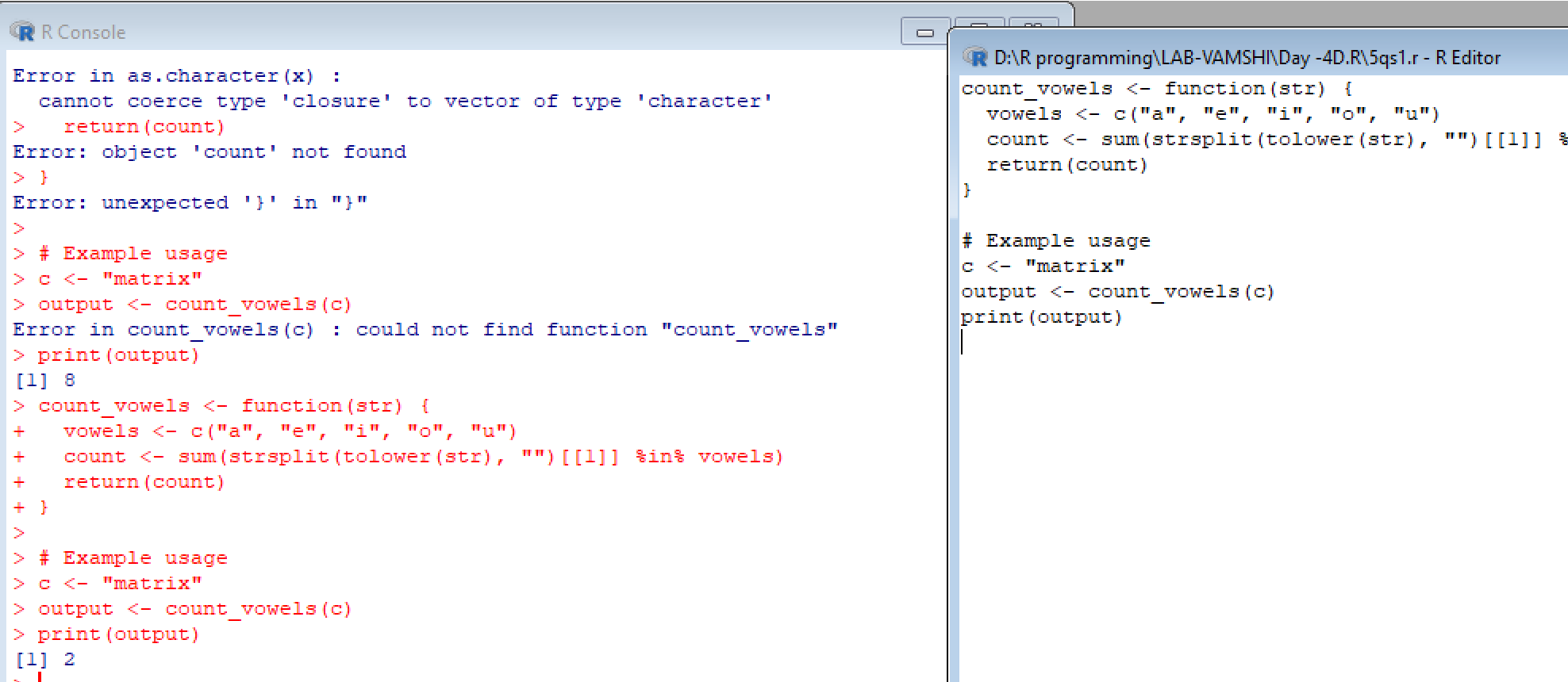
return(count)

}

# Example usage

c <- "matrix"

output <- count\_vowels(c)

print(output)  
  
 **6.Load inbuild dataset “ChickWeight” in R  
(i) Explore the summary of Data set, like number of Features and its type. Fins the number of records for each features  
(ii)Extract last 6 records of dataset  
(iii) order the data frame, in ascending order by feature name  “weight”  grouped by feature “diet”  
(iv)Perform melting function based on “Chick","Time","Diet"   features as ID variables  
(v)Perform cast function to display the mean value of weight grouped by Diet  
  
code:**

# Load the ChickWeight dataset

data(ChickWeight)

# (i) Explore the summary of the dataset

summary(ChickWeight)

str(ChickWeight)

sapply(ChickWeight, length)

# (ii) Extract the last 6 records of the dataset

last\_six <- tail(ChickWeight, 6)

print(last\_six)

# (iii) Order the data frame in ascending order by the "weight" feature grouped by "diet"

ordered\_df <- ChickWeight[order(ChickWeight$weight), ]

print(ordered\_df)

# (iv) Perform melting function based on "Chick", "Time", "Diet" features as ID variables

library(reshape2)

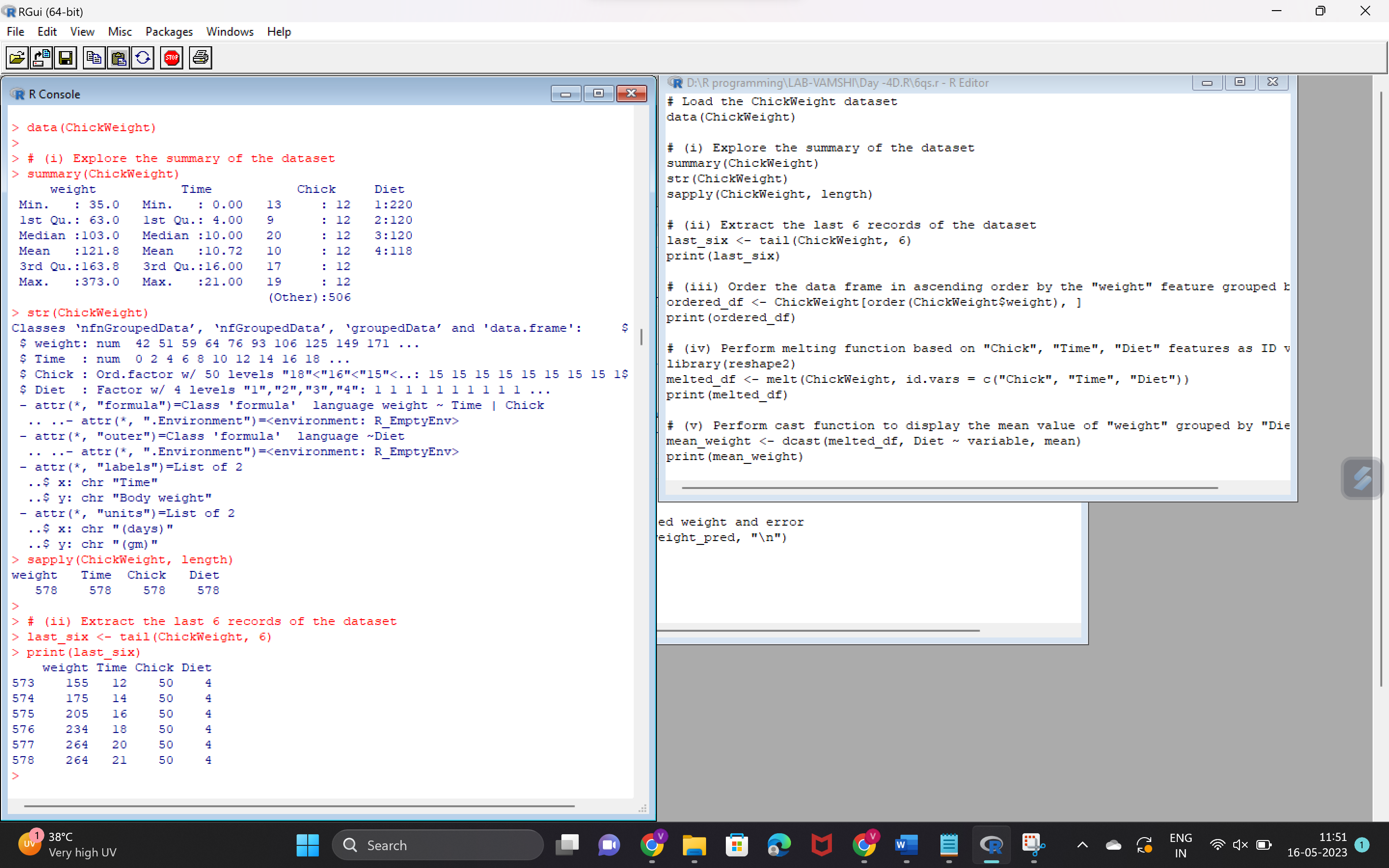
melted\_df <- melt(ChickWeight, id.vars = c("Chick", "Time", "Diet"))

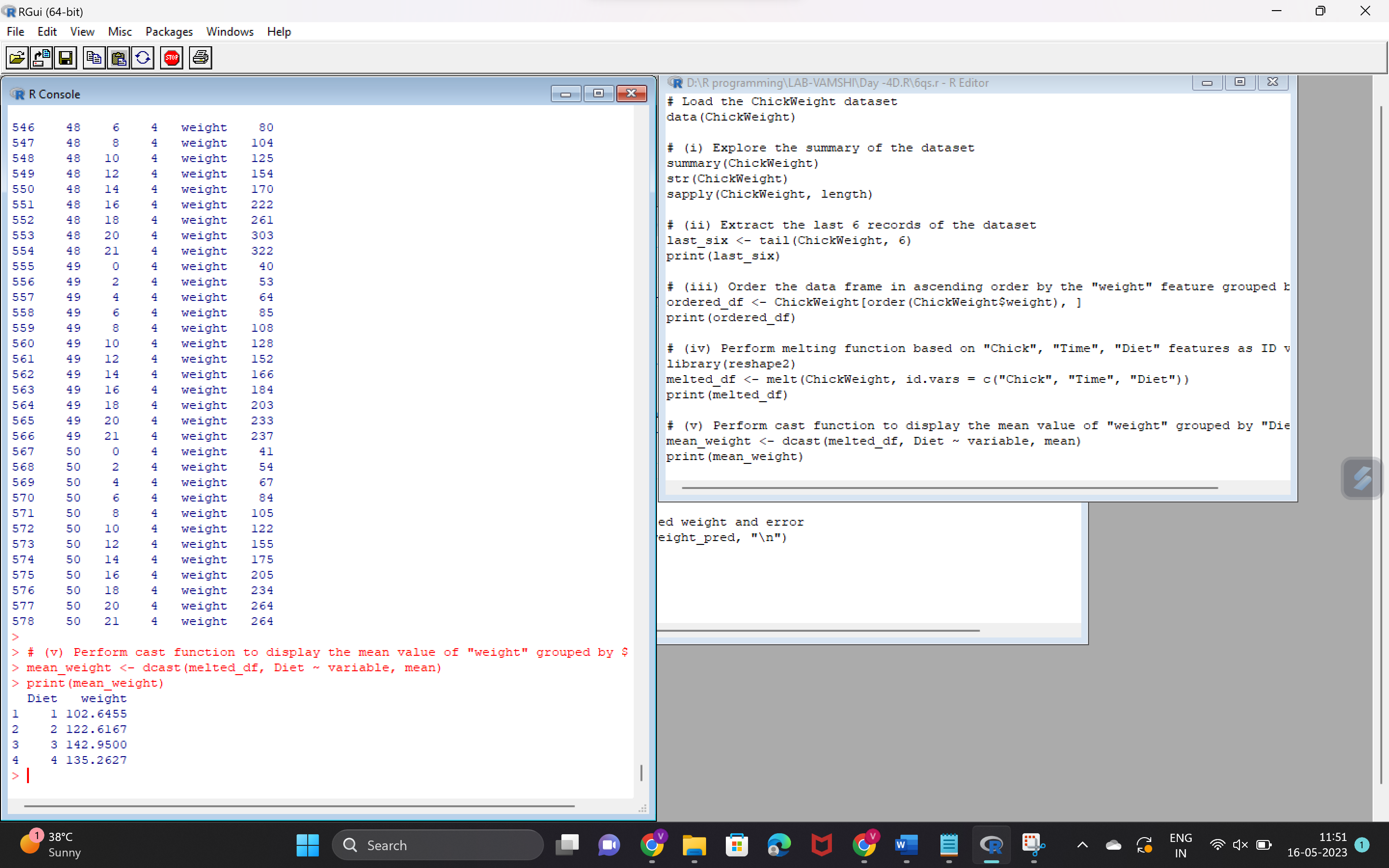
print(melted\_df)

# (v) Perform cast function to display the mean value of "weight" grouped by "Diet"

mean\_weight <- dcast(melted\_df, Diet ~ variable, mean)

print(mean\_weight)

****

**  
7.(i)Get the Statistical  Summary of  “ChickWeight” dataset  
 (ii)Create Box plot for “weight”  grouped by “Diet”  
 (iii)Create a Histogram for  “Weight” features  belong to Diet- 1 category  
 (iv) Create a Histogram for  “Weight” features  belong to Diet- 4 category  
 (v) Create Scatter plot  for weight vs Time grouped by Diet**

**Code:**

# Load ChickWeight dataset

data("ChickWeight")

# (i) Get the Statistical Summary of ChickWeight dataset

summary(ChickWeight)

# (ii) Create Box plot for weight grouped by Diet

library(ggplot2)

ggplot(ChickWeight, aes(x = factor(Diet), y = weight)) +

geom\_boxplot() +

labs(x = "Diet", y = "Weight") +

ggtitle("Box plot of Weight grouped by Diet")

# (iii) Create a Histogram for Weight features belong to Diet-1 category

hist(ChickWeight$weight[ChickWeight$Diet == 1],

main = "Histogram of Weight for Diet-1",

xlab = "Weight")

# (iv) Create a Histogram for Weight features belong to Diet-4 category

hist(ChickWeight$weight[ChickWeight$Diet == 4],

main = "Histogram of Weight for Diet-4",

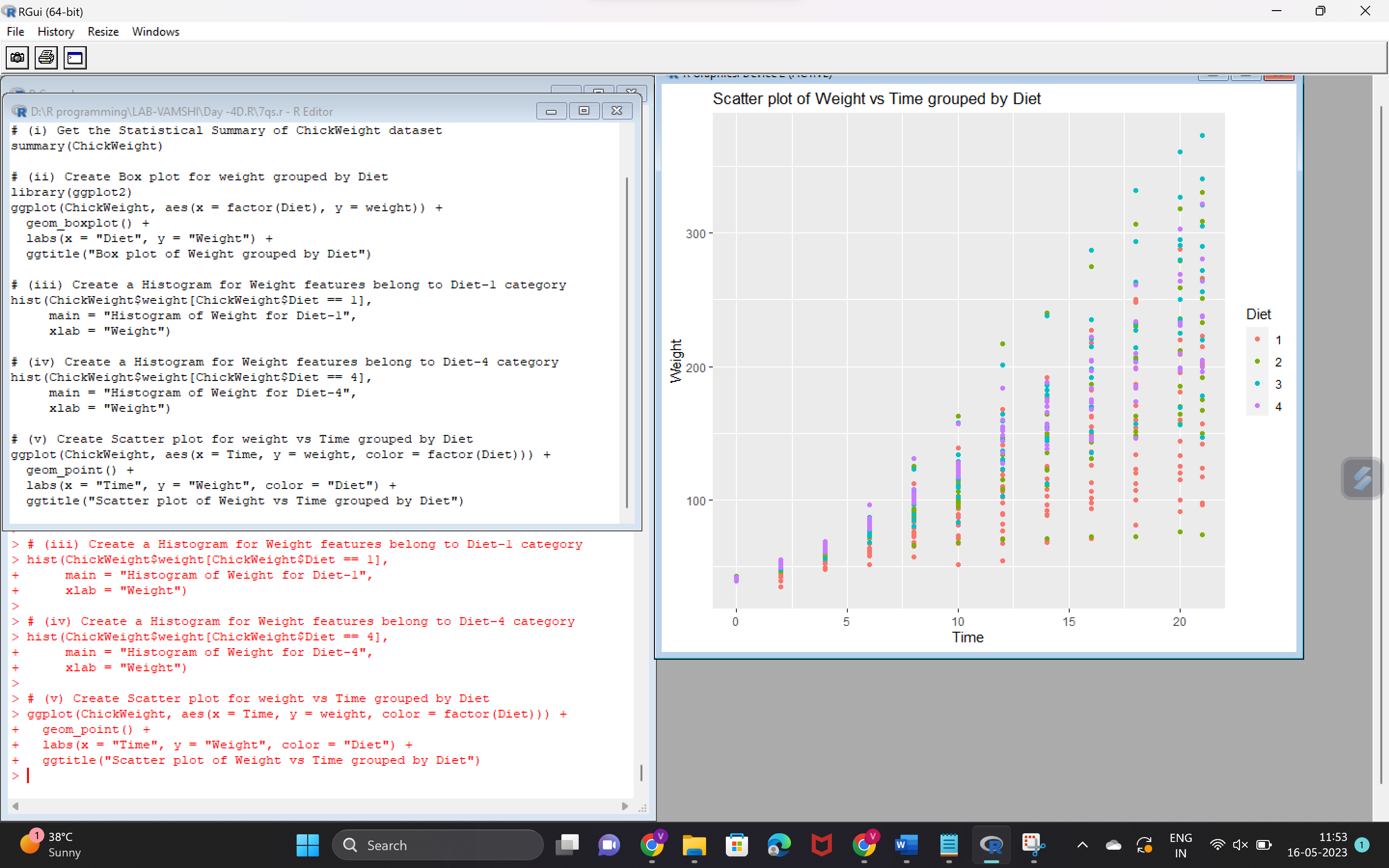
xlab = "Weight")

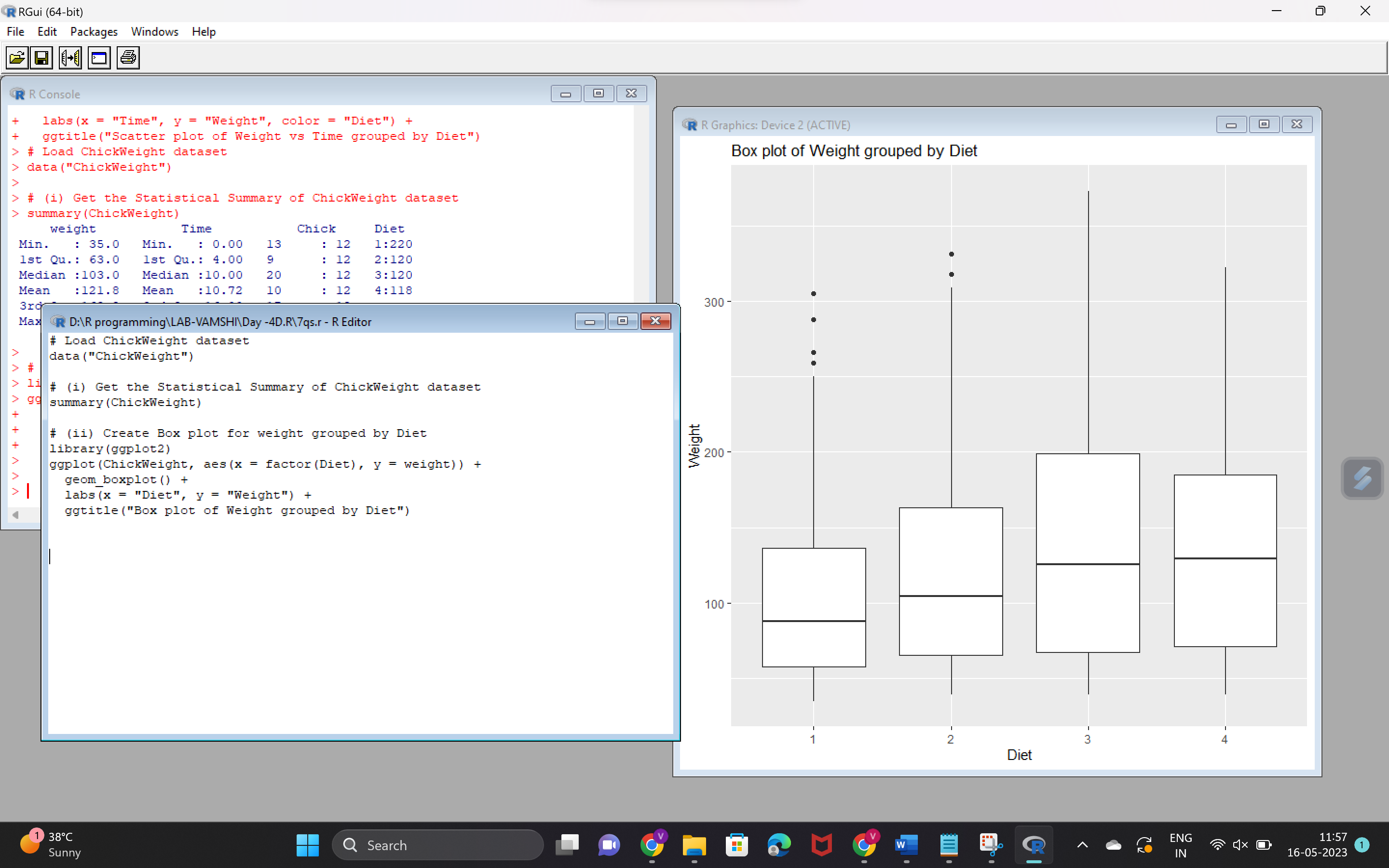
# (v) Create Scatter plot for weight vs Time grouped by Diet

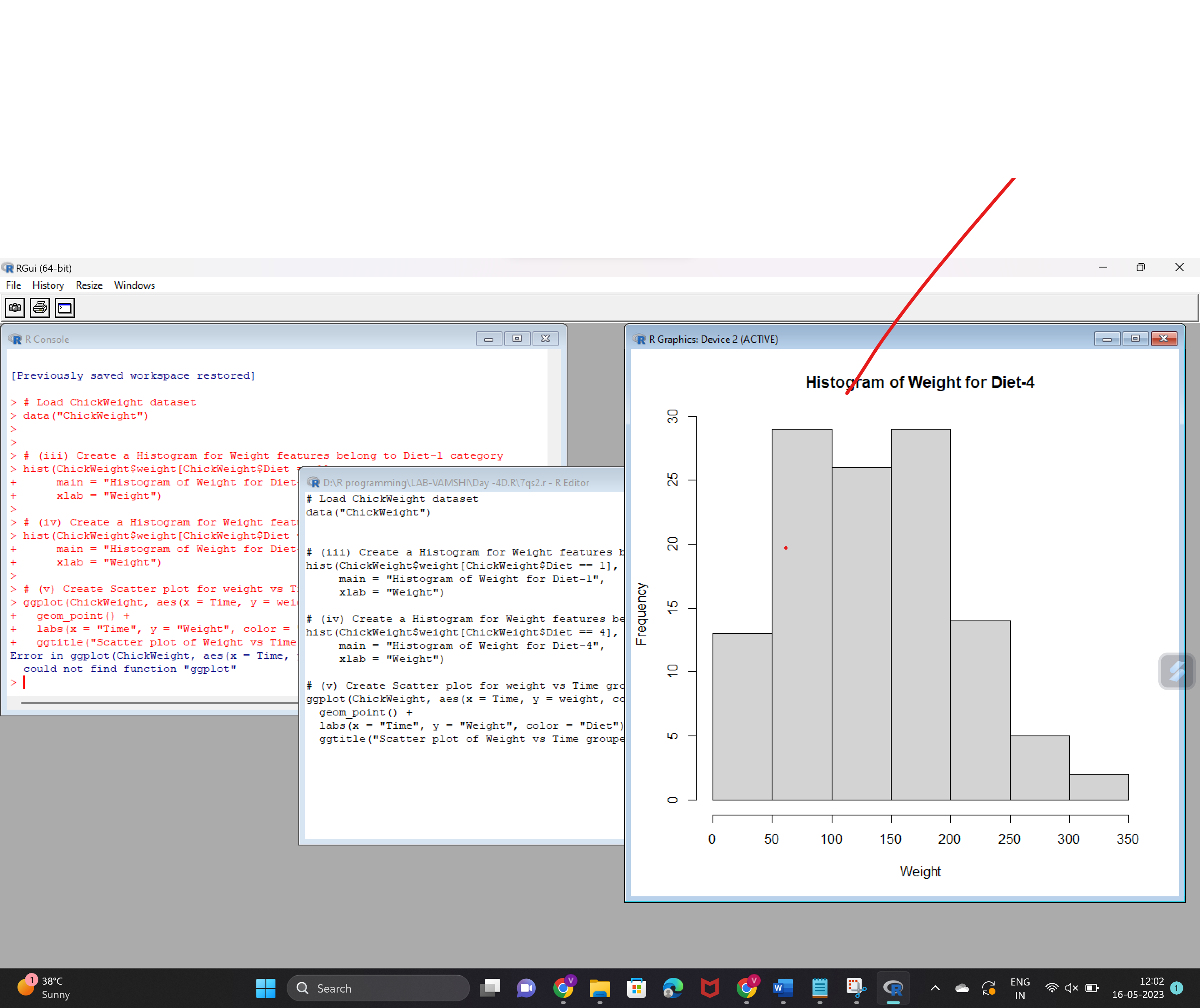
ggplot(ChickWeight, aes(x = Time, y = weight, color = factor(Diet))) +

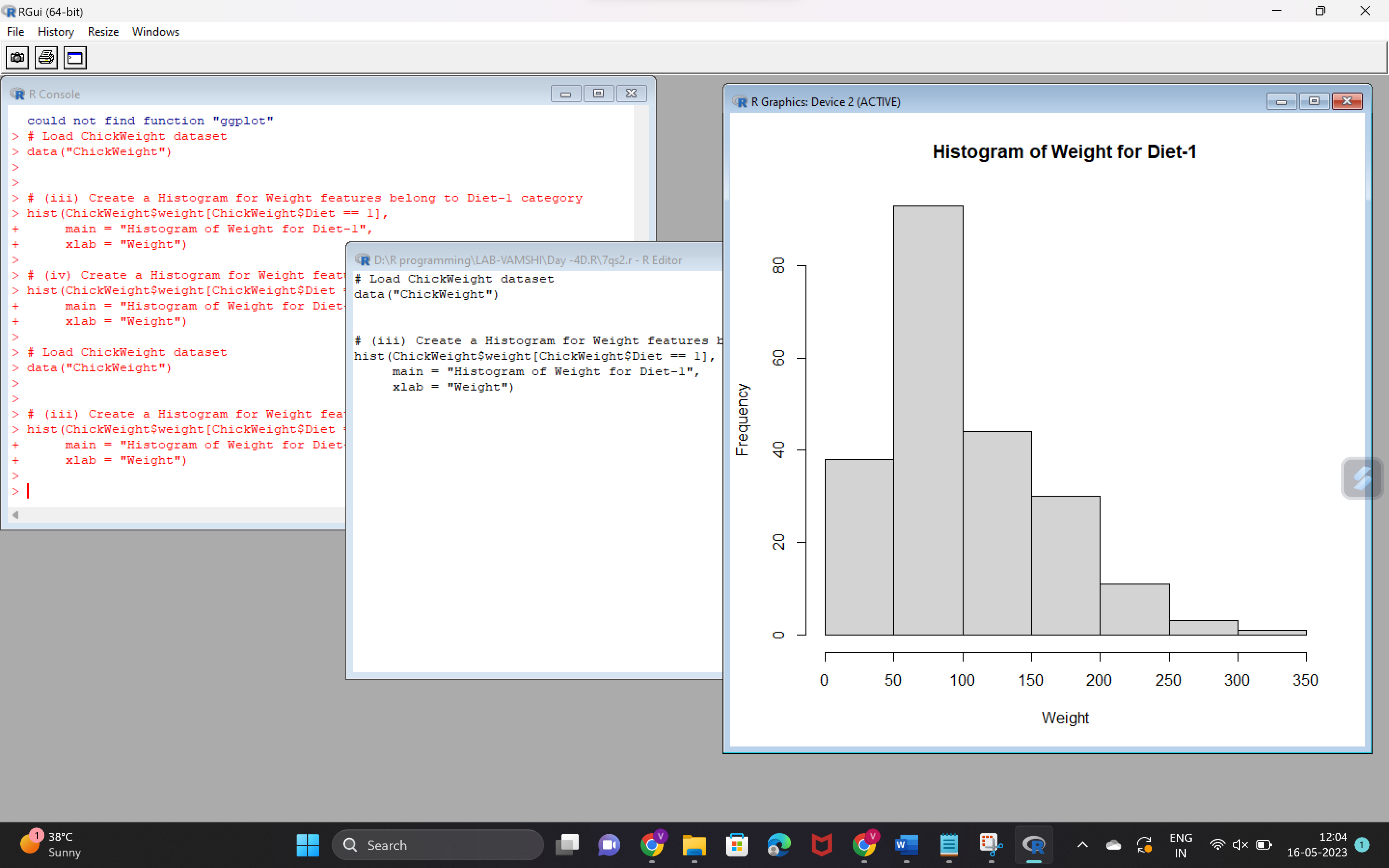
geom\_point() +

labs(x = "Time", y = "Weight", color = "Diet") +

ggtitle("Scatter plot of Weight vs Time grouped by Diet")  


****

****

  
 **8.(i) Create multi regression model to find a weight of the chicken , by “Time” and “Diet” as as predictor variables  
(ii) Predict weight for Time=10 and Diet=1  
(iii)Find the error in model for same**

**Code:**

# Load ChickWeight dataset

data("ChickWeight")

# (i) Create a multiple regression model to predict weight based on time and diet

model <- lm(weight ~ Time + Diet, data = ChickWeight)

# (ii) Predict weight for Time=10 and Diet=1

new\_data <- data.frame(Time = 10, Diet = 1)

predicted\_weight <- predict(model, newdata = new\_data)

# (iii) Calculate the error in the model

actual\_weight <- ChickWeight$weight[ChickWeight$Time == 10 & ChickWeight$Diet == 1]

error <- actual\_weight - predicted\_weight

# Print the results

cat("Predicted Weight for Time=10 and Diet=1:", predicted\_weight, "\n")

cat("Actual Weight for Time=10 and Diet=1:", actual\_weight, "\n")

cat("Error in the Model:", error)

