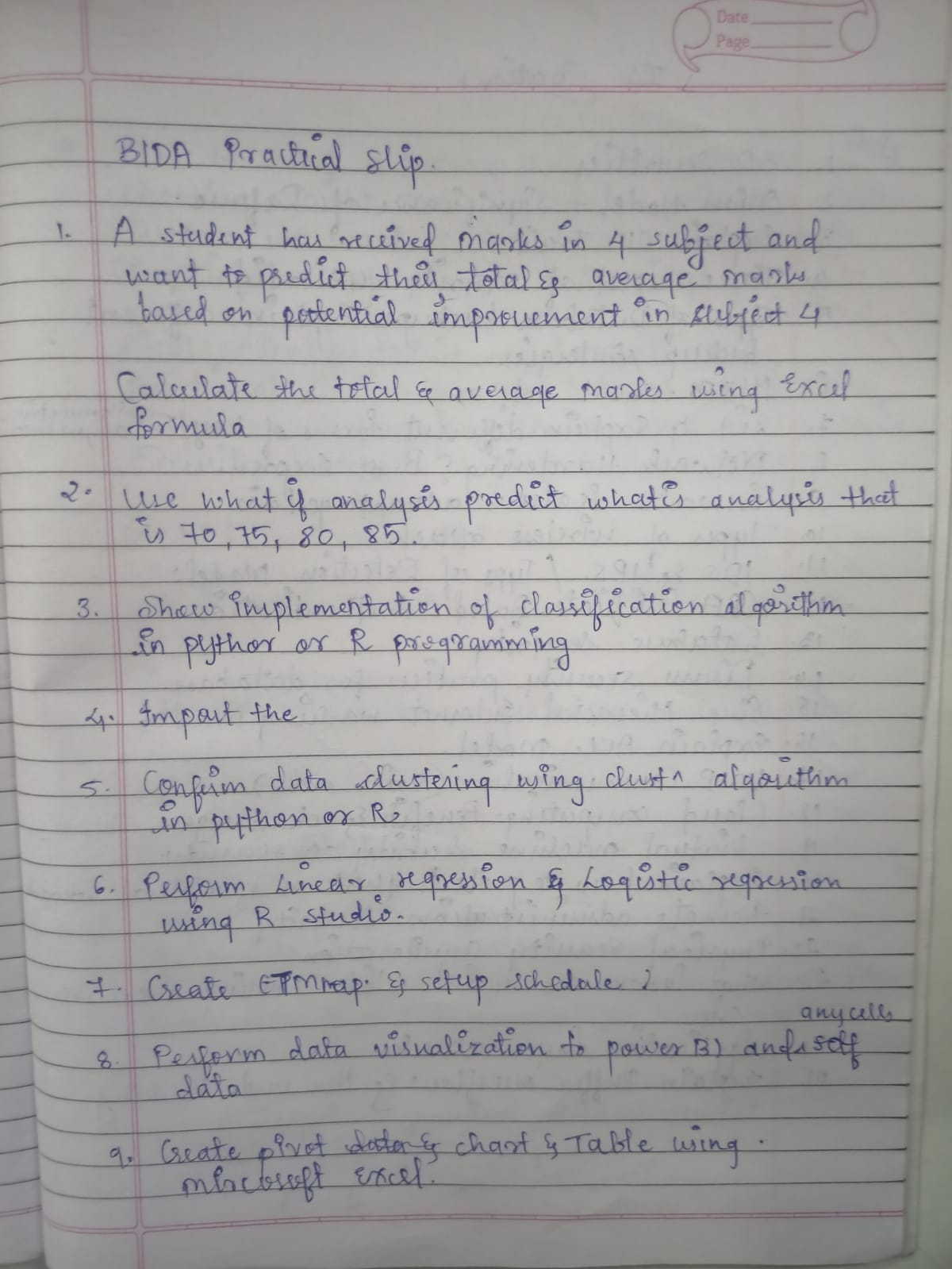
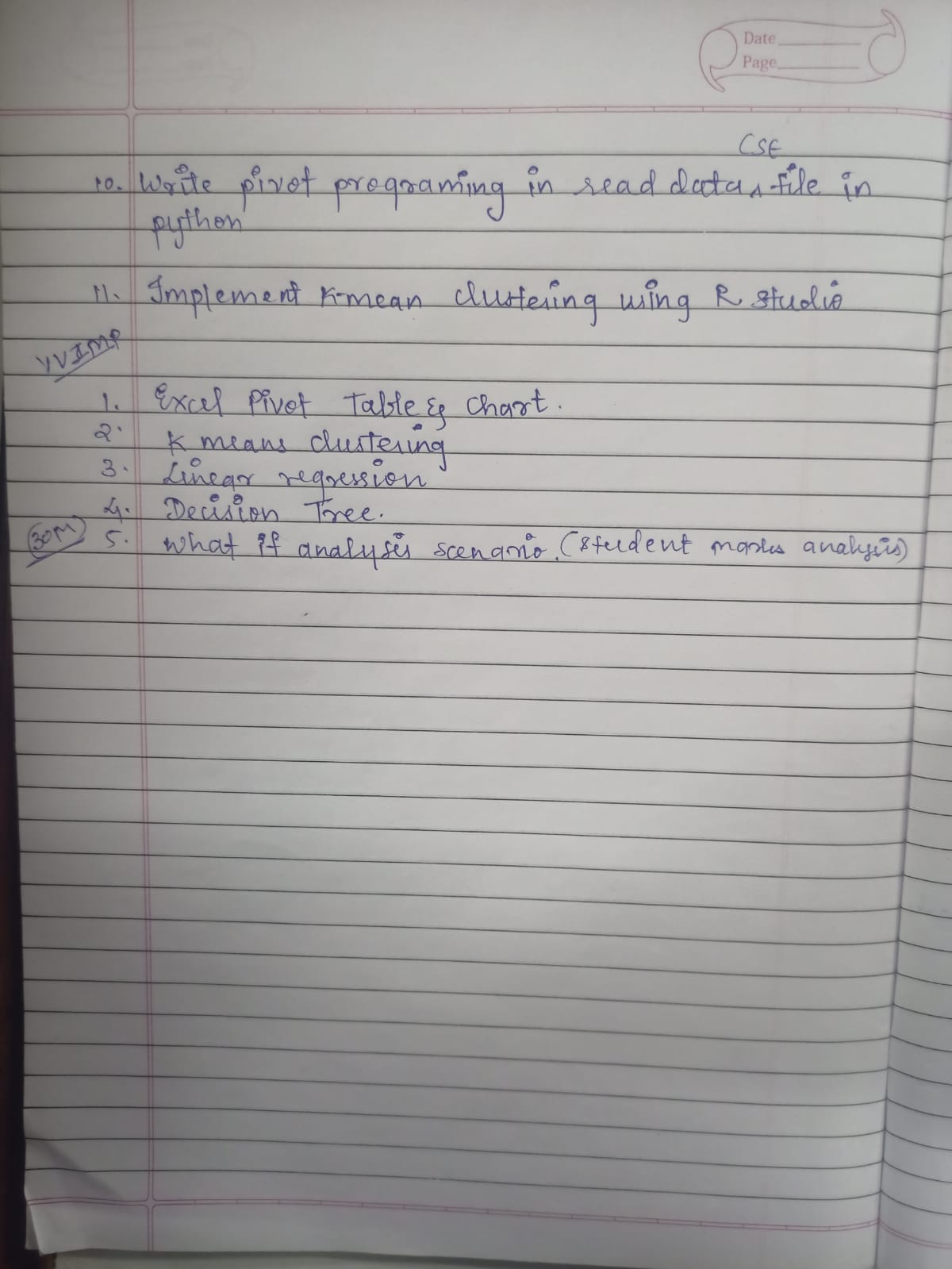
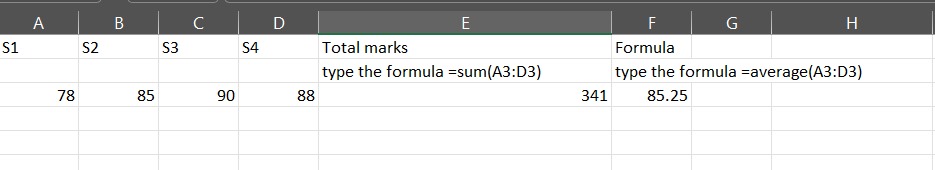
**BIDA Practical Slip**

****



1. **A student has received marks in 4 subjects and wants to predict their total & average marks based on potential improvement in subject 4.  
    Calculate the total & average marks using Excel formula.  
     
   Use what-if analysis to predict. What's the analysis that is 70, 75, 80, 85?**

**Ans:**

****

**SCENARIO MANAGER -> MODIFY A SCENARIO AND SELECT THE CELLS AND MAKE CHANGES -> SHOW SUMMARY**

**excel**

1. **Show implementation of classification algorithm in Python or R programming.**

**Ans:**

**Practical 6**

# Get the data points in form of a R vector.

rainfall <-c(799,1174.8,865.1,1334.6,635.4,918.5,685.5,998.6,784.2,985,882.8,1071)

# Convert it to a time series object.

rainfall.timeseries <- ts(rainfall,start = c(2012,1),frequency = 12)

# Print the timeseries data.

print(rainfall.timeseries)

# Give the chart file a name.

png(file = "rainfall.png")

# Plot a graph of the time series.

plot(rainfall.timeseries)

# Save the file.

dev.off()

Output:

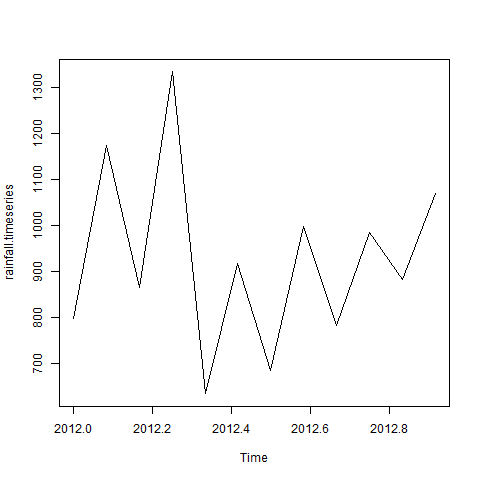
When we execute the above code, it produces the following result and chart −

Jan Feb Mar Apr May Jun Jul Aug Sep

2012 799.0 1174.8 865.1 1334.6 635.4 918.5 685.5 998.6 784.2

Oct Nov Dec

2012 985.0 882.8 1071.0

****

1. **Import the data warehouse in Microsoft Excel and create Pivot table and Pivot Chart**

**Ans:**  
Same as question 8(almost)

1. **Perform data clustering using cluster algorithms in Python or R.**

**Ans:**  
**Practical 7**

First import package by click tools- install package- search “party” install

library(party)

print(head(readingSkills))

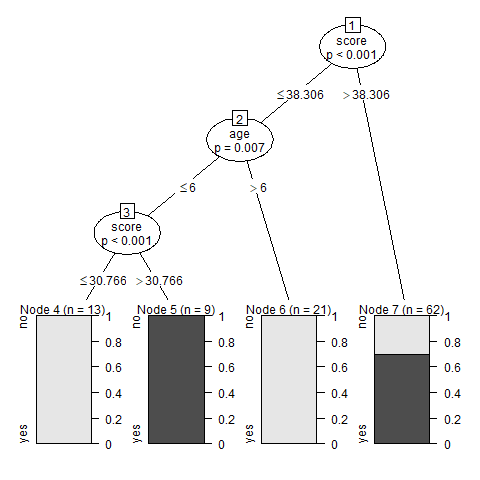
input.dat<-readingSkills[c(1:105),]

png(file="suraj.png")

output.tree<-ctree(nativeSpeaker~age+shoeSize+score,data=input.dat)

plot(output.tree)

dev.off()

****

1. **Perform linear & logistic regression using R studio.**

**Ans:**  
**Practical 9 -linear**

# Create the predictor and response variable.

x <- c(151, 174, 138, 186, 128, 136, 179, 163, 152, 131)

y <- c(63, 81, 56, 91, 47, 57, 76, 72, 62, 48)

relation <- lm(y~x)

# Give the chart file a name.

png(file = "linearregression.png")

# Plot the chart.

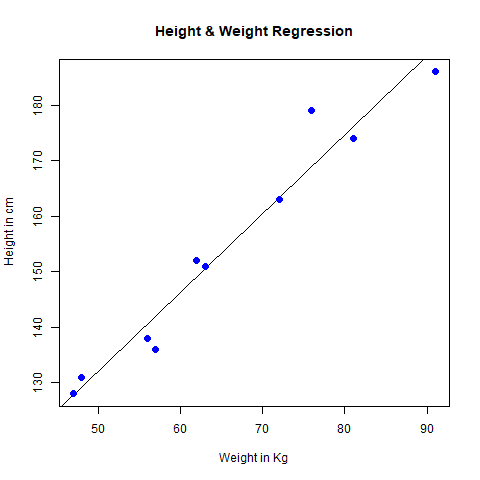
plot(y,x,col = "blue",main = "Height & Weight Regression",

abline(lm(x~y)),cex = 1.3,pch = 16,xlab = "Weight in Kg",ylab = "Height in

cm")

# Save the file.

dev.off()

  
**logistic regression**

# Predictor (x) and response variable (y)

x <- c(2, 3, 5, 7, 9, 10, 12, 15, 18, 20) # Example predictor

y <- c(0, 0, 0, 1, 0, 1, 1, 1, 1, 1) # Binary response (0 or 1)

# Fit logistic regression model

model <- glm(y ~ x, family = binomial)

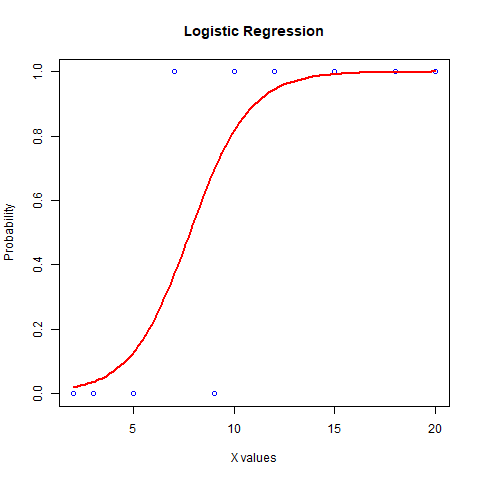
# Plot logistic regression curve

png(file = "logisticregression.png")

plot(x, y, col = "blue", main = "Logistic Regression", xlab = "X values", ylab = "Probability")

curve(predict(model, data.frame(x = x), type = "response"), add = TRUE, col = "red", lwd = 2)

dev.off()



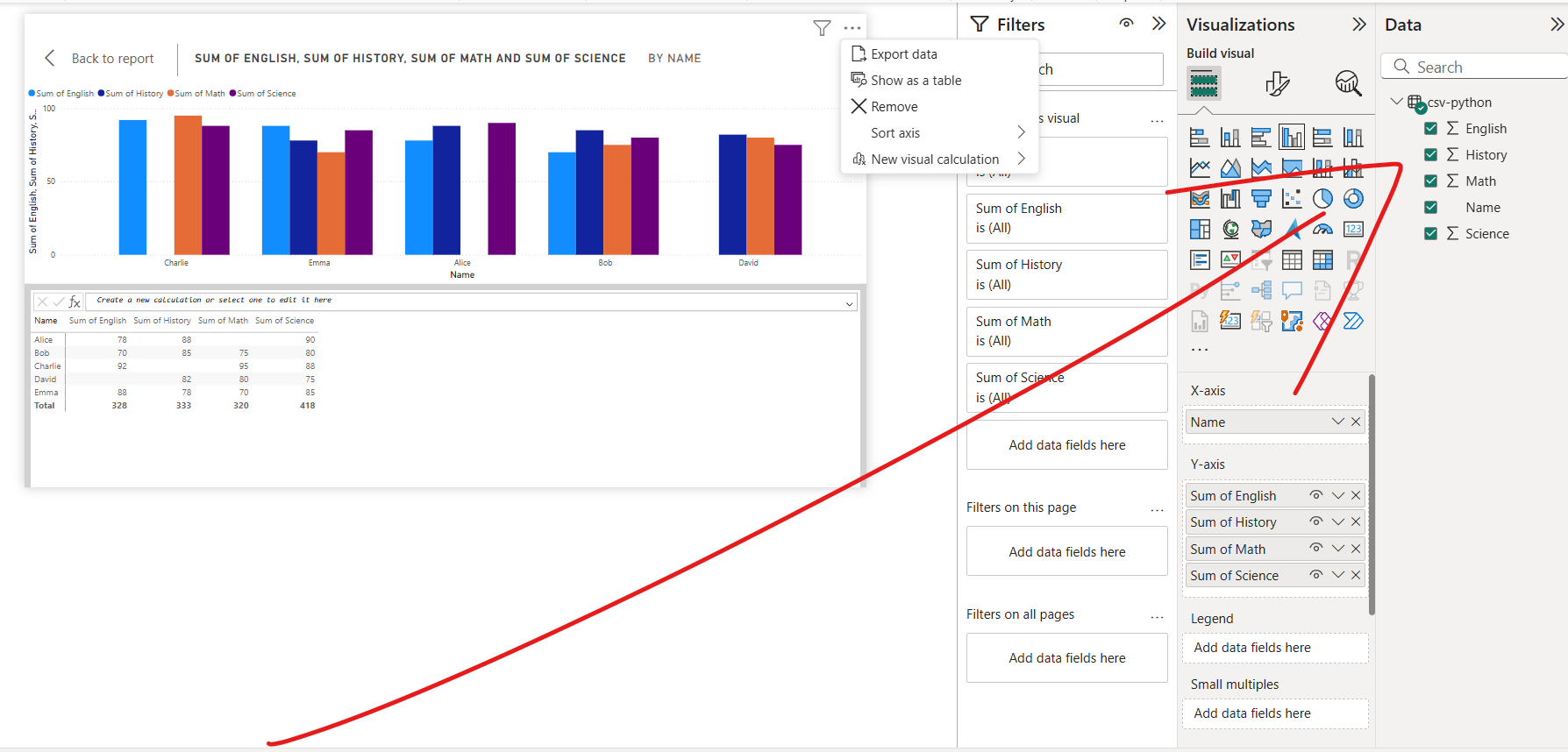
1. Create ETL Map & setup schedule.

Ans: **nhi sikhaya aarya ne aur isko kuch SQL se krna hai**

1. **Perform data visualization in Power BI and Sales data.**

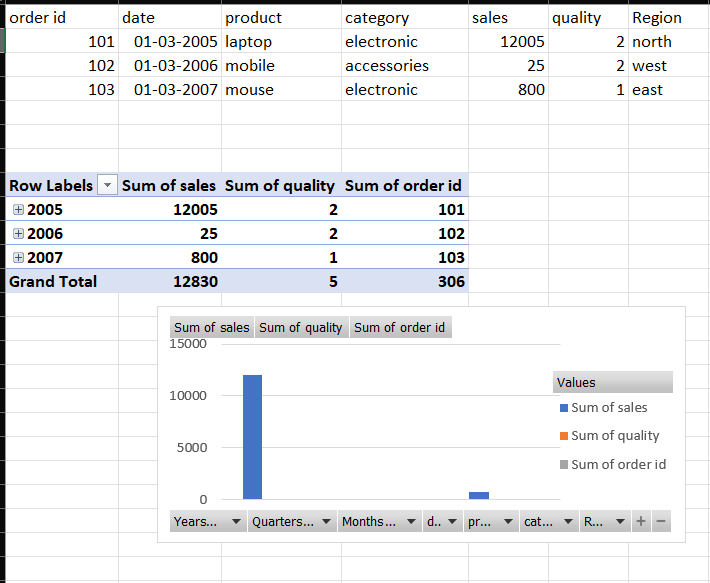
**Ans:**

**application:B1**

**Import data from excel > select excel file and load data > select any chart and double click > select jo bhi chahiye > then on extreme right tick the data you want to display  
  
**

1. **Create pivot table & chart & table using Microsoft Excel.**

**Ans:**



**Sabse pehle upar wala table banao then usko select karo:**  
Go to the **"Insert"** tab and click **TABLE** -> **"PivotTable"**.

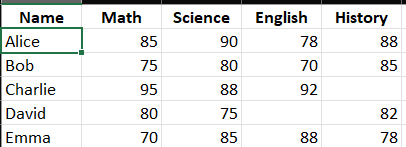
1. Choose where to place the PivotTable (New or Existing Worksheet). CLICK **EXISTING**
2. Click **OK**.
3. Drag fields into the **Rows, Columns, Values, and Filters** areas as needed. (eg order, category etc.)

### **2. Create a Pivot Chart**

1. Click inside the PivotTable.
2. Go to **"Insert" > "PivotChart"**.
3. Choose a chart type (e.g., Column, Pie, Line).
4. Click **OK** to generate the chart.
5. **Write programming in read data CSV file in Python.**

**Ans:  
Python IDLE  
Kya pata CSV file milega ki Nhi  
Path acche se copy karo ( / jo hai kabhi ulta copy hota hai toh usko seedha kr dena)**

(ye csv file khud se bhi bana skte hai excel me same bas **.cvs extension** se save krna hai)



Ye cmd me type karna hai dono cheez

Cmd 1: -m ensurepip --default-pip

Sabse pehle cmd2: “pip install pandas”  
  
Now use **IDLE and paste the code**

import pandas as pd

data = pd.read\_csv("C:/Users/dhari/Desktop/csv-python.csv")

print("First 5 rows of the data:")

print(data.head())

print("\nMissing values in each column:")

print(data.isnull().sum())

print("\nSummary statistics for numerical columns:")

print(data.describe())

**Output:** just shows the missing values from the table

1. **Implement K-means clustering using R Studio.**

**Ans:**

**RSTUDIO > FILE > NEW > RSCRIPT**

Practical 8  
newiris<-iris

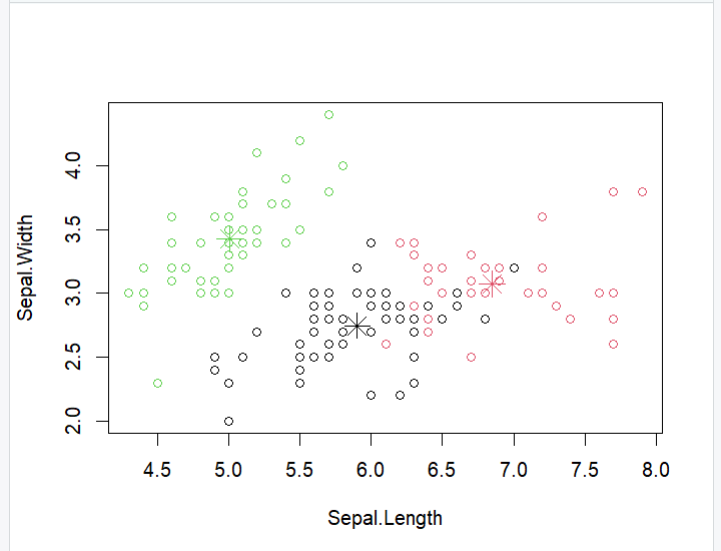
newiris$Species<-NULL

(kc<- kmeans(newiris,3))

table(iris$Species,kc$cluster)

plot(newiris[c("Sepal.Length","Sepal.Width")],col=kc$cluster)

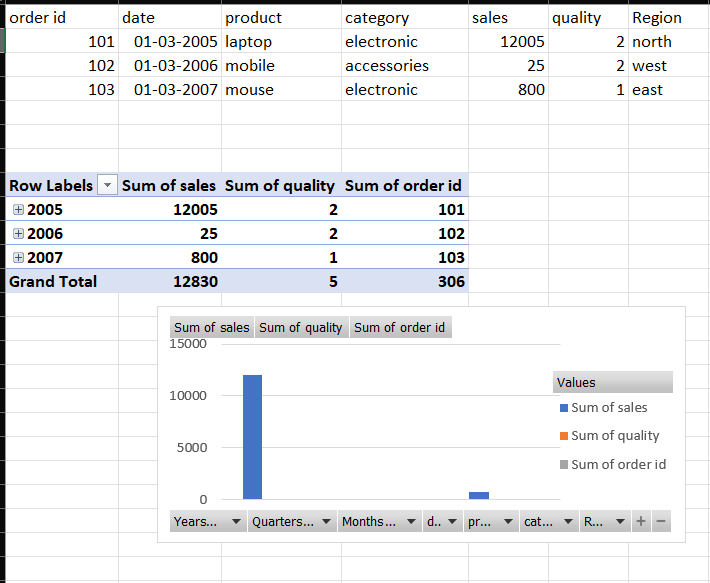
points(kc$centers[,c("Sepal.Length","Sepal.Width")],col=1:3,pch=8,cex=2)



**V.V. IMP**

1. **Excel Pivot Table & Chart.**

**Ans:**



**Sabse pahale uper wala table banao then usko select karo:**  
Go to the **"Insert"** tab and click **"PivotTable"**.

1. Choose where to place the PivotTable (New or Existing Worksheet).
2. Click **OK**.
3. Drag fields into the **Rows, Columns, Values, and Filters** areas as needed.

### **2. Create a Pivot Chart**

1. Click inside the PivotTable.
2. Go to **"Insert" > "PivotChart"**.
3. Choose a chart type (e.g., Column, Pie, Line).
4. Click **OK** to generate the chart.
5. **K-means clustering.**

**Ans:**

**Practical 8**

newiris<-iris

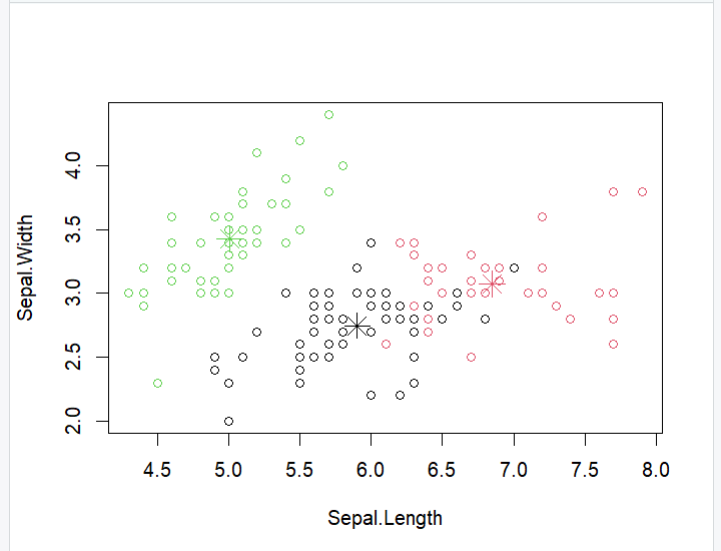
newiris$Species<-NULL

(kc<- kmeans(newiris,3))

table(iris$Species,kc$cluster)

plot(newiris[c("Sepal.Length","Sepal.Width")],col=kc$cluster)

points(kc$centers[,c("Sepal.Length","Sepal.Width")],col=1:3,pch=8,cex=2)



1. **Linear regression.**

**Ans:**

**Practical 9**

# Create the predictor and response variable.

x <- c(151, 174, 138, 186, 128, 136, 179, 163, 152, 131)

y <- c(63, 81, 56, 91, 47, 57, 76, 72, 62, 48)

relation <- lm(y~x)

# Give the chart file a name.

png(file = "linearregression.png")

# Plot the chart.

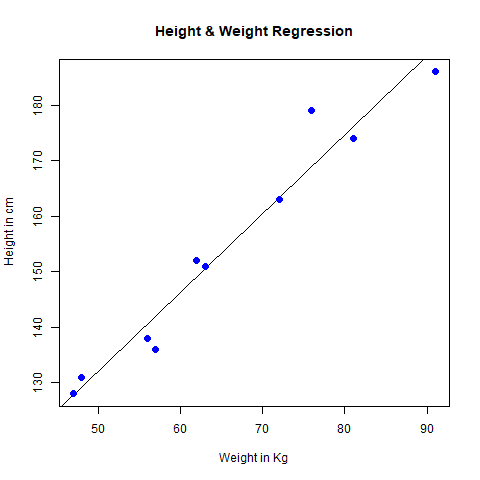
plot(y,x,col = "blue",main = "Height & Weight Regression",

abline(lm(x~y)),cex = 1.3,pch = 16,xlab = "Weight in Kg",ylab = "Height in

cm")

# Save the file.

dev.off()



1. **Decision Tree.**

**Ans:**

**Practical 7**

library(party)

print(head(readingSkills))

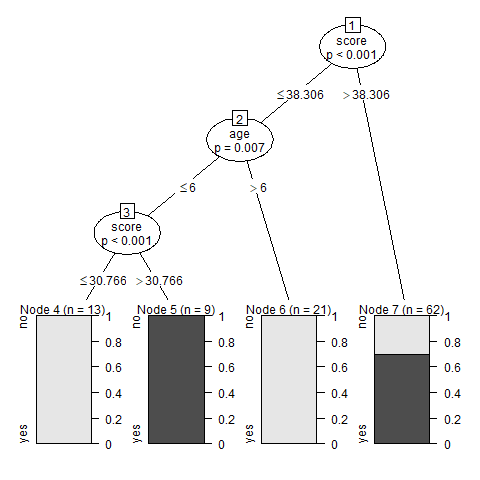
input.dat<-readingSkills[c(1:105),]

png(file="suraj.png")

output.tree<-ctree(nativeSpeaker~age+shoeSize+score,data=input.dat)

plot(output.tree)

dev.off()

****

1. **What-If analysis scenario (Student marks analysis).**

**Ans:**

Refer Q1