**1) Consider following observations/data. And apply simple linear regression and find out estimated coefficients b0 and b1.( use numpy package) x= [0, 1, 2, 3, 4, 5, 6, 7, 8, 9,11,13] y = ([1, 3, 2, 5, 7, 8, 8, 9, 10, 12,16, 18]**

import numpy as np

x = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 13]) y = np.array([1, 3, 2, 5, 7, 8, 8, 9, 10, 12, 16, 18])

# Calculate the means of x and y x\_mean = np.mean(x) y\_mean = np.mean(y)

# Calculate b1 (slope) and b0 (intercept) b1 = np.sum((x - x\_mean) \* (y - y\_mean)) / np.sum((x - x\_mean) \*\* 2) b0 = y\_mean - b1 \* x\_mean print("Intercept (b0):", b0) print("Slope (b1):", b1)

output:

Intercept (b0): 0.838709677419355

Slope (b1): 1.2889200561009817

**2) Consider following observations/data. And apply simple linear regression and find out estimated coefficients b1 and b1 Also analyse the performance of the model (Use sklearn package) x = np.array([1,2,3,4,5,6,7,8]) y = np.array([7,14,15,18,19,21,26,23])**

import numpy as np from sklearn.linear\_model import LinearRegression from sklearn.metrics import mean\_squared\_error, r2\_score

# Given data x = np.array([1, 2, 3, 4, 5, 6, 7, 8]).reshape(-1, 1) y = np.array([7, 14, 15, 18, 19, 21, 26, 23])

# Create and fit the model model = LinearRegression().fit(x, y)

# Get the intercept and slope b0= model.intercept\_

b1 = model.coef\_[0]

# Make predictions y\_pred = model.predict(x)

# Calculate performance metrics mse = mean\_squared\_error(y, y\_pred) r2 = r2\_score(y, y\_pred)

print("Intercept (b0):", b0) print("Slope (b1):", b1) print("Mean Squared Error (MSE):", mse) print("R-squared (R2):", r2) output:

Intercept (b0): 7.642857142857139

Slope (b1): 2.2738095238095246

Mean Squared Error (MSE): 3.4657738095238106

R-squared (R2): 0.886774107294781

**SET B** <https://www.w3schools.com/python/python_ml_multiple_regression.asp>

import pandas

from sklearn import linear\_model

df = pandas.read\_csv("data.csv")

X = df[['Weight', 'Volume']] y = df['CO2']

regr = linear\_model.LinearRegression() regr.fit(X, y)

#predict the CO2 emission of a car where the weight is 2300kg, and the volume is 1300cm3:

predictedCO2 = regr.predict([[2300, 1300]])

print(predictedCO2) output [107.2087328] **SET C** import numpy as np

# Generate 200 samples from a normal distribution centered around 100 with standard deviation 5