21.Scenario:

you are a scientist conducting research on rare elements found in a specific region. Your goal is to

estimate the average concentration of a rare element in the region using a random sample of

measurements. You will use the NumPy library to perform point estimation and calculate

confidence intervals for the population mean.The rare element concentration data is stored in a CSV

file named "rare\_elements.csv," where each row contains a single measurement of the

concentration.

Question:

write a Python program that allows the user to input the sample size, confidence level, and desired

level of precision.

Code:

import numpy as np

import pandas as pd

import scipy.stats as stats

def calculate\_confidence\_interval(sample\_data, sample\_size, confidence\_level, precision):

# Calculate sample mean and standard deviation

sample\_mean = np.mean(sample\_data)

sample\_std = np.std(sample\_data, ddof=1) # Sample standard deviation

# Determine the Z-score or t-score based on the confidence level

alpha = 1 - confidence\_level

if sample\_size > 30: # Use Z-score if sample size > 30

z\_score = stats.norm.ppf(1 - alpha/2)

else: # Use t-score if sample size <= 30

t\_score = stats.t.ppf(1 - alpha/2, df=sample\_size - 1)

# Margin of Error Calculation

margin\_of\_error = z\_score \* (sample\_std / np.sqrt(sample\_size)) if sample\_size > 30 else t\_score \* (sample\_std / np.sqrt(sample\_size))

# Calculate Confidence Interval

lower\_bound = sample\_mean - margin\_of\_error

upper\_bound = sample\_mean + margin\_of\_error

# Calculate desired precision

while (upper\_bound - lower\_bound) > precision:

sample\_size += 1

margin\_of\_error = z\_score \* (sample\_std / np.sqrt(sample\_size)) if sample\_size > 30 else t\_score \* (sample\_std / np.sqrt(sample\_size))

lower\_bound = sample\_mean - margin\_of\_error

upper\_bound = sample\_mean + margin\_of\_error

return sample\_mean, lower\_bound, upper\_bound

def main():

# Input the Excel file

file\_path = r"C:\Users\hares\Downloads\q21\_05.xlsx" # Change this to your actual Excel file path

# Load data from Excel (assuming data is in the first sheet)

df = pd.read\_excel(file\_path, engine='openpyxl')

# Assuming the data is in the first column named 'Concentration'

data = df['Concentration'].dropna() # Dropping any missing values

# Input parameters from the user

sample\_size = int(input("Enter the sample size: "))

confidence\_level = float(input("Enter the confidence level (e.g., 0.95 for 95% confidence): "))

precision = float(input("Enter the desired precision (e.g., 0.01): "))

# Calculate Confidence Interval and Point Estimate

mean, lower\_bound, upper\_bound = calculate\_confidence\_interval(data, sample\_size, confidence\_level, precision)

# Output results

print(f"Sample Mean: {mean:.4f}")

print(f"Confidence Interval: ({lower\_bound:.4f}, {upper\_bound:.4f})")

print(f"Sample Size: {sample\_size}")

if \_\_name\_\_ == "\_\_main\_\_":

main()

output :’

Enter the sample size: 20

Enter the confidence level (e.g., 0.95 for 95% confidence): 58

Enter the desired precision (e.g., 0.01): 54

Sample Mean: 0.1410

Confidence Interval: (nan, nan)

Sample Size: 20

Dataset :

Enter the sample size: 20

Enter the confidence level (e.g., 0.95 for 95% confidence): 58

Enter the desired precision (e.g., 0.01): 54

Sample Mean: 0.1410

Confidence Interval: (nan, nan)

Sample Size: 20