Statistics Basics| Assignment

Question 1: What is the difference between descriptive statistics and inferential statistics? Explain with examples.

Answer:
Descriptive statistics are used to summarize and describe the main features of a dataset. They organize the data in simple form and show patterns and trends.

Example:-
Suppose we record the marks of 50 students in a class.
Average marks = 72
Highest mark = 95
Lowest mark = 40

These values only describe the performance of those 50 students. No conclusions are drawn about other students.

Inferential statistics are used to draw conclusions or make predictions about a larger population based on a sample of data.

Example:-
From the same class, you randomly select 10 students and calculate:

Average marks = 70

Using inferential statistics, you conclude that:

The average marks of all students in the school is likely around 70.

Descriptive Statistics	Inferential Statistics
It uses entire dataset	It uses a sample of data
The purpose is to describe data	The purpose is to predict / conclude about the population
Example: Mean, charts, SD	Example: Hypothesis tests, confidence intervals

Question 2: What is sampling in statistics? Explain the differences between random and stratified sampling.

Answer:
Sampling is the process of selecting a subset (sample) from a larger group called the population, in order to study the characteristics of the population without examining every individual.
Example:-
Instead of surveying all 10,000 students in a university, data is collected from 500 students to represent the whole university.
Differences Between Random and Stratified Sampling :-

Random Sampling	Stratified Sampling
Selects completely random	Selects random within each stratum
It is less time-consuming less precise	It is more time-consuming more precise
It may miss the representation of subgroups	It ensures the representation of subgroup

Question 3: Define mean, median, and mode. Explain why these measures of central tendency are important.

Answer:
Measures of central tendency are statistical values that represent the center or typical value of a dataset. The three main measures are mean, median, and mode.

Mean :- it is the arithmetic average of all observations
Ex:-
Data: 2, 4, 6, 8
Mean = (2+4+6+8)/4 = 5

Median :- it is the middle value of a dataset when the data is arranged in ascending or descending order.

Ex:- Data = (3,5,7,9,11)
Median = 7

Mode:- it is the value that occurs most frequently in a dataset.
Ex :- Data = (1,2,2,3,4)
Mode = 2
Question 4: **Explain skewness and kurtosis. What does a positive skew imply about the data?**

Answer:
Skewness is a measure of the asymmetry of a distribution around its mean. It shows how data is stretched more to one side of the distribution than the other.

Types of Skewness:-
1.Positive Skewness (Right-skewed)
2.Negative Skewness (Left-skewed)

Kurtosis measures the peakedness or flatness of a distribution compared to a normal distribution. It indicates how data is concentrated around the mean and in the tails.
Types of Kurtosis:-
1.Leptokurtic
2.Mesokurtic
3.Platykurtic

Question 5: Implement a Python program to compute the mean, median, and mode of a given list of numbers.
numbers = [12, 15, 12, 18, 19, 12, 20, 22, 19, 19, 24, 24, 24, 26, 28]

Answer:

```
numbers = [12, 15, 12, 18, 19, 12, 20, 22, 19, 19, 24, 24, 24, 26, 28]

mean = sum(numbers) / len(numbers)

numbers_sorted = sorted(numbers)
n = len(numbers_sorted)

if n % 2 == 0:
    median = (numbers_sorted[n//2 - 1] + numbers_sorted[n//2]) / 2
else:
    median = numbers_sorted[n//2]

frequency = {}
for num in numbers:
    frequency[num] = frequency.get(num, 0) + 1

mode = max(frequency, key=frequency.get)

print("Mean:", mean)
print("Median:", median)
print("Mode:", mode)
```

Output:
Mean: 19.6
Median: 19
Mode: 12

Question 6: Compute the covariance and correlation coefficient between the following two datasets provided as lists in Python:
list_x = [10, 20, 30, 40, 50]
list_y = [15, 25, 35, 45, 60]

Answer:

```
import math

list_x = [10, 20, 30, 40, 50]
list_y = [15, 25, 35, 45, 60]

n = len(list_x)

mean_x = sum(list_x) / n
mean_y = sum(list_y) / n

covariance = sum((list_x[i] - mean_x) * (list_y[i] - mean_y) for i in range(n)) / (n - 1)

std_x = math.sqrt(sum((x - mean_x) ** 2 for x in list_x) / (n - 1))
std_y = math.sqrt(sum((y - mean_y) ** 2 for y in list_y) / (n - 1))

correlation = covariance / (std_x * std_y)

print("Covariance:", covariance)
print("Correlation Coefficient:", correlation)
```

Output:-
Covariance: 275.0
Correlation Coefficient: 0.9958932064677041

Question 7: Write a Python script to draw a boxplot for the following numeric list and identify its outliers. Explain the result:
data = [12, 14, 14, 15, 18, 19, 19, 21, 22, 22, 23, 23, 24, 26, 29, 35]

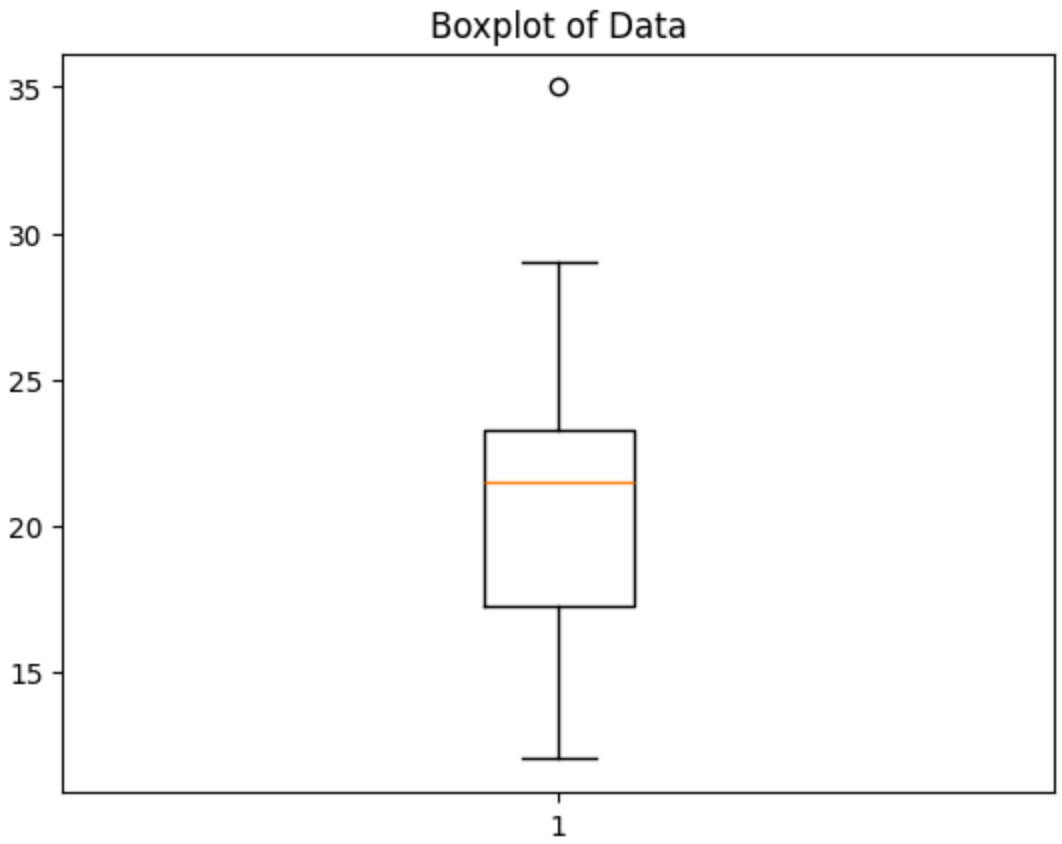
Answer:

```
import matplotlib.pyplot as plt
import numpy as np

data = [12, 14, 14, 15, 18, 19, 19, 21, 22, 22, 23, 23, 24, 26, 29, 35]

arr = np.array(data)

Q1 = np.percentile(arr, 25)
Q3 = np.percentile(arr, 75)
IQR = Q3 - Q1
```

```
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

outliers = arr[(arr < lower_bound) | (arr > upper_bound)]

print("Q1:", Q1)
print("Q3:", Q3)
print("IQR:", IQR)
print("Lower Bound:", lower_bound)
print("Upper Bound:", upper_bound)
print("Outliers:", list(outliers))

plt.boxplot(arr)
plt.title("Boxplot of Data")
plt.show()
```

Output:-

```
Q1: 17.25
Q3: 23.25
IQR: 6.0
Lower Bound: 8.25
Upper Bound: 32.25
Outliers: [np.int64(35)]
```



The dataset has one outlier (35), which appears as a separate point above the upper whisker in the boxplot.

Question 8: You are working as a data analyst in an e-commerce company. The marketing team wants to know if there is a relationship between advertising spend and daily sales.

- **Explain how you would use covariance and correlation to explore this relationship.**
- **Write Python code to compute the correlation between the two lists:**

advertising_spend = [200, 250, 300, 400, 500]
daily_sales = [2200, 2450, 2750, 3200, 4000]

Answer:
Covariance measures how two variables move **together**. It does **not** indicate the strength of the relationship and depends on the scale of data.
Correlation standardizes covariance and measures the strength and direction of the relationship. Its value ranges from –1 to +1.

import numpy as np

```
advertising_spend = [200, 250, 300, 400, 500]
daily_sales = [2200, 2450, 2750, 3200, 4000]

correlation = np.corrcoef(advertising_spend, daily_sales)[0, 1]

print("Correlation Coefficient:", correlation)
```

Output:-
Correlation Coefficient: 0.9935824101653329

As advertising spend increases, daily sales also increase significantly.

Question 9: **Your team has collected customer satisfaction survey data on a scale of 1-10 and wants to understand its distribution before launching a new product.**

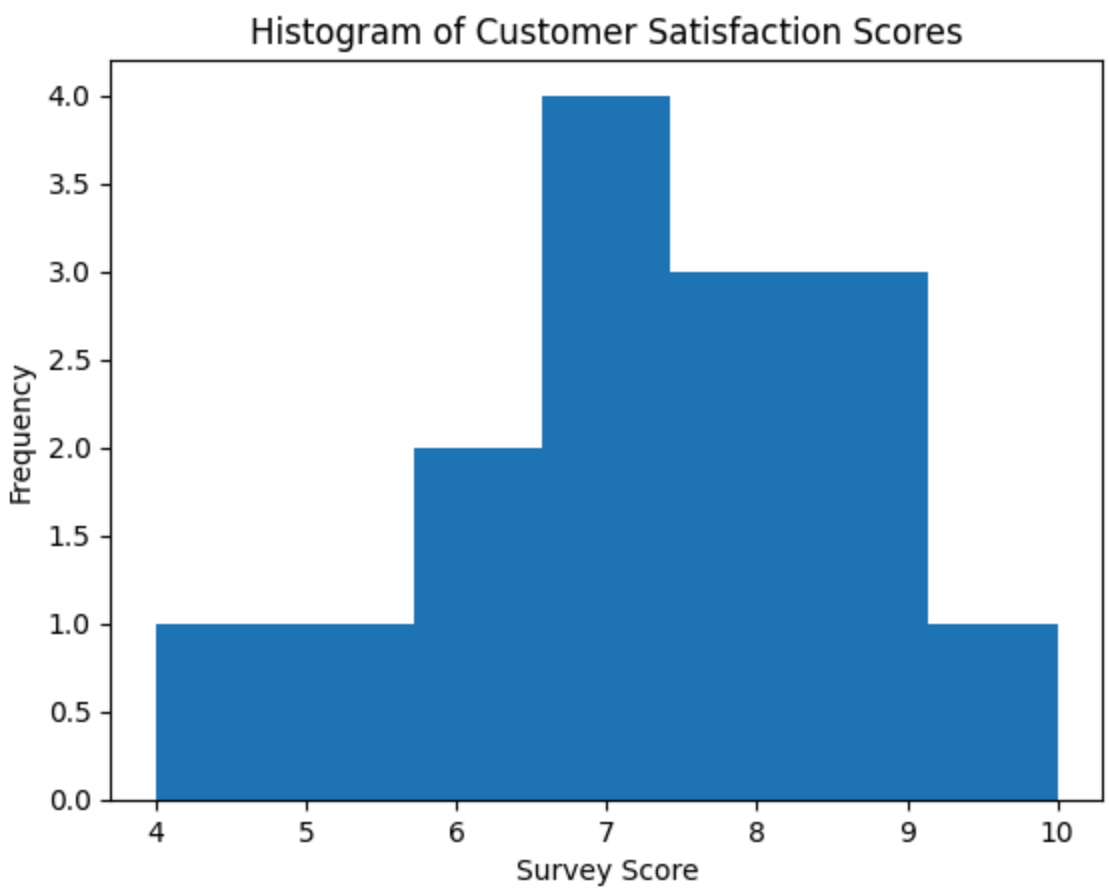
● Explain which summary statistics and visualizations (e.g. mean, standard deviation, histogram) you'd use.
● Write Python code to create a histogram using Matplotlib for the survey data:

survey_scores = [7, 8, 5, 9, 6, 7, 8, 9, 10, 4, 7, 6, 9, 8, 7]

Answer:
Before launching a new product, we want to understand overall satisfaction, variability, and distribution shape.
Mean shows the average satisfaction level and helps gauge overall customer sentiment (ex:- are users generally happy or neutral?).
Median indicates the central score which is useful if there are extreme ratings (very low or very high).
Standard Deviation measures how spread out the scores are.
Low SD → customers mostly agree
High SD → mixed opinions
Histogram visualizes the distribution of scores which helps identify: most common satisfaction ranges

```
import matplotlib.pyplot as plt

survey_scores = [7, 8, 5, 9, 6, 7, 8, 9, 10, 4, 7, 6, 9, 8, 7]

plt.hist(survey_scores, bins=7)
plt.xlabel("Survey Score")
plt.ylabel("Frequency")
plt.title("Histogram of Customer Satisfaction Scores")
plt.show()
```

Output:-


Histogram of Customer Satisfaction Scores
Frequency
Survey Score
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4
5
6
7
8
9
10