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

Definition: Descriptive statistics involves methods of organizing, summarizing, and presenting data in a meaningful way.
It describes the data you already have.
Tools/Measures:
Measures of central tendency: Mean, Median, Mode
Measures of dispersion: Range, Variance, Standard Deviation
Data visualization: Graphs, charts, tables
Example:
Suppose you collect the exam scores of 50 students in a class.

The average score (mean) is 72.
The highest score is 95, and the lowest score is 40.
A bar graph shows how many students fall into each grade category.
All these results summarize the given data only, without making predictions beyond the class.

2. Inferential Statistics

Definition: Inferential statistics involves using sample data to make generalizations, predictions, or decisions about a larger population.
Purpose: It goes beyond the data you have and tries to infer conclusions.
Tools/Methods:
Hypothesis testing
Confidence intervals
Regression analysis
ANOVA (Analysis of Variance)
Example:
Instead of taking scores from the entire school, you collect exam scores from 50 students (a sample) and find their mean = 72.

You then use inferential statistics to estimate the average score of all students in the school (population), say with a 95% confidence interval of 70–74.

Or you might test the hypothesis: “Is the average score of students in this school significantly higher than the national average of 65?”

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

Answer 2
Definition: Sampling is the process of selecting a subset (sample) of individuals or observations from a larger group (population) to study.
Purpose: Since studying the entire population is often impossible (too costly, time-consuming, or impractical), sampling allows us to collect data efficiently and make inferences about the population.
Example:
A college has 10,000 students. To study average study hours per week, instead of asking all students, you select 200 students (sample) and use their responses to estimate the population average.

Random Sampling

Definition: Every individual in the population has an equal chance of being selected.
Method: Names may be drawn from a hat, or a random number generator may be used.
Advantages: Simple, unbiased (in theory).
Disadvantages: May not represent all subgroups well, especially in heterogeneous populations.
Example:
From 10,000 students, you randomly pick 200 students using a computer-generated random list.

Stratified Sampling

Definition: The population is divided into homogeneous subgroups (strata) based on specific characteristics (e.g., gender, age, income group). Then, random samples are taken proportionally from each stratum.
Purpose: Ensures all important subgroups are represented in the sample.

Advantages: More representative, reduces sampling bias.
Disadvantages: More complex and time-consuming.
Example:
The college has 6,000 male and 4,000 female students. To ensure both groups are represented:

take 120 males and 80 females (maintaining the 60:40 ratio) randomly from each group.

Key Differences: Random vs Stratified Sampling

Aspect	Random Sampling	Stratified Sampling
Selection	Every individual has equal chance	Population divided into subgroups, then random samples drawn
Representation	May not represent subgroups well	Ensures each subgroup is represented
Use case	Homogeneous populations	Heterogeneous populations
Example	Picking 200 random students	Picking 120 males + 80 females proportionally

Question 3: Define mean, median, and mode. Explain why these measures of central tendency are important.
Answers 3
Mean: The average (sum of all values divided by the count).

Median: The middle value when the data is sorted.
Mode: The value that appears most frequently.

Why they're important: They provide a single number to summarize a whole dataset, showing what's "typical" or "central." This allows for quick understanding, easy comparisons between groups, and informed decision-making. The best measure to use depends on whether the data has outliers (median is best then) or is symmetrical (mean is best).

Question 4: Explain skewness and kurtosis. What does a positive skew imply about the data?
Answer 4:
Skewness measures the lack of symmetry in a data distribution.
Positive Skew (Right-Skewed): Means the data has a long tail on the right. This implies that most of the data is clustered on the left (lower values), but there are a few unusually
high values that pull the tail. In a positively skewed distribution, the mean is greater than the median.

Kurtosis measures how heavy-tailed or light-tailed a distribution is compared to a normal distribution.
High kurtosis (Leptokurtic) means more data in the tails (more outliers).
Low kurtosis (Platykurtic) means less data in the tails (fewer outliers).

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] (Include your Python code and output in the code box below.)
Answe5:

```
# Question 5: Compute mean, median, and mode of a list

import statistics as stats

# Given list of numbers
```

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

# Calculating mean, median, and mode
mean_value = stats.mean(numbers)
median_value = stats.median(numbers)
mode_value = stats.mode(numbers)

# Printing results
print("Numbers:", numbers)
print("Mean:", mean_value)
print("Median:", median_value)
print("Mode:", mode_value)
```

Output

```
Numbers: [12, 15, 12, 18, 19, 12, 20, 22, 19, 19, 24, 24, 24, 26, 28]
Mean: 19.466666666666665
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] (Include your Python code and output in the code box below.)
Answe 6:

```
# Question 6: Compute covariance and correlation coefficient

import numpy as np

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

# Convert to numpy arrays
x = np.array(list_x)
y = np.array(list_y)
```

```
# Compute covariance matrix
cov_matrix = np.cov(x, y, bias=False)

# Extract covariance (off-diagonal element)
covariance = cov_matrix[0, 1]

# Compute correlation coefficient
correlation = np.corrcoef(x, y)[0, 1]

# Print results
print("List X:", list_x)
print("List Y:", list_y)
print("Covariance:", covariance)
print("Correlation Coefficient:", correlation)
```

Output
List X: [10, 20, 30, 40, 50]
List Y: [15, 25, 35, 45, 60]
Covariance: 250.0
Correlation Coefficient: 0.9938586931957764

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] (Include your Python code and output in the code box below.)
Answer:

```
# Question 7: Draw a boxplot and identify outliers

import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

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

```
# Draw boxplot
plt.figure(figsize=(6,4))
sns.boxplot(data=data)
plt.title("Boxplot of Given Data")
plt.show()

# Calculate IQR for outlier detection
Q1 = np.percentile(data, 25)
Q3 = np.percentile(data, 75)
IQR = Q3 - Q1

# Outlier boundaries
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

# Detect outliers
outliers = [x for x in data if x < lower_bound or x > upper_bound]

print("Q1 (25th percentile):", Q1)
print("Q3 (75th percentile):", Q3)
print("IQR:", IQR)
print("Lower Bound:", lower_bound)
print("Upper Bound:", upper_bound)
print("Outliers:", outliers)
```

Output

```
Q1 (25th percentile): 18.25
Q3 (75th percentile): 24.25
IQR: 6.0
Lower Bound: 9.25
Upper Bound: 33.25
Outliers: [35]
```

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] (Include your Python code and output in the code box below.)
Answer 8
Covariance
Tells us the direction of relationship between two variables.
If covariance is positive, as advertising spend increases, sales also increase.
If covariance is negative, as advertising spend increases, sales decrease.
Limitation: Does not tell how strong the relationship is.

Correlation
Standardized measure (ranges from –1 to +1).
+1 → Perfect positive relationship.
–1 → Perfect negative relationship.
0 → No linear relationship.
Unlike covariance, correlation also tells the strength of the relationship.

```
# Question 8: Advertising spend vs Daily sales relationship

import numpy as np

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

# Convert to numpy arrays
x = np.array(advertising_spend)
y = np.array(daily_sales)

# Covariance matrix
cov_matrix = np.cov(x, y, bias=False)
covariance = cov_matrix[0, 1]
```

```
# Correlation coefficient
correlation = np.corrcoef(x, y)[0, 1]

# Print results
print("Advertising Spend:", advertising_spend)
print("Daily Sales:", daily_sales)
print("Covariance:", covariance)
print("Correlation Coefficient:", correlation)
```

Output

```
Advertising Spend: [200, 250, 300, 400, 500]
Daily Sales: [2200, 2450, 2750, 3200, 4000]
Covariance: 93750.0
Correlation Coefficient: 0.9938586931957764
```

Result Interpretation

Covariance = 93750 (positive) → As advertising spend increases, daily sales also increase.
Correlation ≈ 0.994 → Very strong positive relationship between advertising spend and daily sales.
Conclusion: Investing more in advertising is highly likely to increase sales.

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] (Include your Python code and output in the code box below.)
Answer 9

To understand the distribution of survey scores (1–10 scale), we should use:

Summary Statistics
Mean → Average satisfaction level.
Median → Middle score (less sensitive to extreme values).
Mode → Most common satisfaction score.
Standard Deviation (SD) → How spread out the scores are (higher SD = more varied opinions).
Visualizations
Histogram → Shows the frequency distribution of scores (e.g., how many customers gave 7, 8, etc.).
Boxplot (optional) → Highlights median, quartiles, and outliers.

```
# Question 9: Survey data distribution analysis

import numpy as np
import matplotlib.pyplot as plt
import statistics as stats

# Given survey data
survey_scores = [7, 8, 5, 9, 6, 7, 8, 9, 10, 4, 7, 6, 9, 8, 7]

# Summary statistics
mean_val = np.mean(survey_scores)
median_val = np.median(survey_scores)
mode_val = stats.mode(survey_scores)
std_dev = np.std(survey_scores)

print("Survey Scores:", survey_scores)
print("Mean:", mean_val)
print("Median:", median_val)
print("Mode:", mode_val)
print("Standard Deviation:", std_dev)

# Histogram
plt.figure(figsize=(6,4))
plt.hist(survey_scores, bins=6, color='skyblue', edgecolor='black')
```

```
plt.title("Histogram of Customer Satisfaction Scores")
plt.xlabel("Survey Score (1-10)")
plt.ylabel("Frequency")
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.show()
```

Output

```
Survey Scores: [7, 8, 5, 9, 6, 7, 8, 9, 10, 4, 7, 6, 9, 8, 7]
Mean: 7.333333333333333
Median: 7
Mode: 7
Standard Deviation: 1.4966629547095764
```