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

Answer: **Descriptive Statistics** aim to describe, show, or summarize data in a meaningful way. They do not allow us to make conclusions beyond the data we have analyzed.

- **Example:** Calculating the average test score of a specific class of 30 students.

Inferential Statistics use a random sample of data taken from a population to describe and make inferences about the population. It allows you to "infer" trends from a smaller group to a larger one.

- **Example:** Surveying 1,000 voters to predict the winner of a national election.

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

Answer: **Sampling** is the process of selecting a subset of individuals from a statistical population to estimate characteristics of the whole population.

- **Random Sampling:** Every member of the population has an equal chance of being selected. It reduces bias and is simple to implement.
- **Stratified Sampling:** The population is divided into subgroups (strata) based on shared characteristics (e.g., age, gender). A random sample is then taken from each stratum to ensure the sample is representative of the entire population structure.

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

These are **Measures of Central Tendency**, used to find the "center" of a data distribution.

- **Mean:** The arithmetic average (sum of values divided by the count). It is sensitive to outliers.
- **Median:** The middle value when data is sorted. It is "robust," meaning it isn't heavily affected by extreme outliers.
- **Mode:** The most frequently occurring value in the dataset.

Importance: They help simplify large datasets into a single representative value, allowing for quick comparisons between different groups of data.

Question 4: Explain skewness and kurtosis. What does a positive skew imply about the data?
Answer: **Skewness** measures the asymmetry of the probability distribution.

- **Positive Skew (Right-Skewed):** The tail on the right side is longer. This implies that most data points are clustered on the left, but there are some exceptionally high values pulling the mean to the right (Mean > Median).

Kurtosis measures the "tailedness" or the sharpness of the peak of the distribution. It tells us how much of the data is in the tails versus the center.

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: SQLAlchemy is an **Object-Relational Mapper (ORM)**. It acts as a bridge between your Python code and your SQL database (MySQL, PostgreSQL, SQLite).

- **Role:** Instead of writing `SELECT * FROM users;`, you write `User.query.all()`.
- **Abstraction:** It allows you to define your database schema as Python classes (Models), making the code cleaner and more maintainable.

```
numbers = [12, 15, 12, 18, 19, 12, 20, 22, 19, 19, 24, 24, 24, 26, 28]
mean_q5 = statistics.mean(numbers)
median_q5 = statistics.median(numbers)
try:
    mode_q5 = statistics.mode(numbers)
except statistics.StatisticsError:
    mode_q5 = "Multimodal or no unique mode"
```

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: Covariance measures the directional relationship between two variables, while the correlation coefficient measures the strength and direction of the linear relationship, normalized between -1 and 1.

The covariance is

275.0275.0

275.0

.
The correlation coefficient is

0.9958932064677040.995893206467704

0.995893206467704

.

```
list_x = [10, 20, 30, 40, 50]
list_y = [15, 25, 35, 45, 60]
covariance_q6 = np.cov(list_x, list_y)[0, 1]
correlation_q6 = np.corrcoef(list_x, list_y)[0, 1]
```

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: A boxplot helps visualize the distribution and identify outliers, which are data points that fall below the lower bound (

8.258.25
8.25
) or above the upper bound (

32.2532.25
32.25
).
The outlier identified is

3535
35
.

```
data = [12, 14, 14, 15, 18, 19, 19, 21, 22, 22, 23, 23, 24, 26, 29, 35]
Q1 = np.percentile(data, 25)
Q3 = np.percentile(data, 75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR
outliers = [x for x in data if x < lower_bound or x > upper_bound]
```

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: Step 1: Explain the Use of Covariance and Correlation

Covariance indicates whether sales increase as spending increases (positive covariance) or decrease (negative covariance). A positive result here means a potential positive relationship. Correlation quantifies the strength of this linear relationship. A correlation coefficient close to

11
1
would suggest a strong positive linear relationship, indicating that higher advertising spend is strongly associated with higher sales.
Step 2: Compute Correlation
The calculated correlation coefficient between advertising spend and daily sales is

0.99358241016533290.9935824101653329

0.9935824101653329
. This suggests a very strong positive linear relationship between the two variables.

```
advertising_spend = [200, 250, 300, 400, 500]
daily_sales = [2200, 2450, 2750, 3200, 4000]
correlation_q8 = np.corrcoef(advertising_spend, daily_sales)[0, 1]
```

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: • **Summary Statistics:** I would use the **Mean** to find the average satisfaction, **Standard Deviation** to see how much the scores vary (consistency), and **Median** to ensure the mean isn't skewed by a few extremely unhappy or happy customers.

- **Visualizations:** A **Histogram** is best for seeing the frequency of scores and identifying the shape of the distribution.

```
survey_scores = [7, 8, 5, 9, 6, 7, 8, 9, 10, 4, 7, 6, 9, 8, 7]
plt.figure(figsize=(8, 6))
plt.hist(survey_scores, bins=len(set(survey_scores)), edgecolor='black',
alpha=0.7)
plt.title('Customer Satisfaction Survey Scores Distribution')
plt.xlabel('Score (1-10)')
plt.ylabel('Frequency')
plt.grid(axis='y', alpha=0.75)
# In a standard execution environment, the plot would be displayed.
# We'll print the computed values for verification as a proxy for the
'output' within the box requirement.

print(f"Q5 Mean: {mean_q5}, Median: {median_q5}, Mode: {mode_q5}")
print(f"Q6 Covariance: {covariance_q6}, Correlation: {correlation_q6}")
print(f"Q7 Outliers: {outliers}, Q1: {Q1}, Q3: {Q3}, IQR: {IQR}, Lower:
{lower_bound}, Upper: {upper_bound}")
print(f"Q8 Correlation: {correlation_q8}")
print(f"Q9 Survey Scores Mean: {statistics.mean(survey_scores)}, Std Dev:
{statistics.stdev(survey_scores)}")
```