STATISTICS BASICS ASSIGNMENT

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

Answer-Descriptive Statistics-Descriptive statistics summarize or describe the main features of a dataset. They do not make predictions or generalizations beyond the data at hand.
Example**:**Suppose a teacher calculates the average score of her 30 students in a math test:
Mean score = 75
Standard deviation = 10
Highest score = 95, Lowest = 50
These statistics describe only this specific class.

Inferential Statistics-Inferential statistics use a sample of data to make predictions or generalization about a larger population. It involves probability and hypothesis testings.

Example-A researcher wants to know the average height of all college students in a country. Instead of measuring everyone, she takes a sample of 500 students and calculates:

Sample mean = 170 cm
Uses inferential statistics to estimate that the population mean is between 168 and 172 cm with 95% confidence.

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 (population) to draw conclusions about the whole population.
Random Sampling-In random sampling, every individual in the population has an equal chance of being selected.
Stratified Sampling-In stratified sampling, the population is divided into subgroups (strata) based on a specific characteristic (e.g., age, gender, income). Then a random sample is taken from each stratum.

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

Answer-Mean-The mean is the sum of all values divided by the number of values.

Mean=sum of all values/number of values

- Useful for further statistical analysis (e.g., standard deviation)
- Sensitive to extreme values (outliers)

Median-The median is the middle value when the data is arranged in order.
-If the number of values is odd, it's the middle one.
-If even, it's the average of the two middle values.
- Good for skewed data (not affected by outliers).

- Represents the "middle" value

Mode- The mode is the value that appears most frequently in the dataset.
- Useful for categorical data
- Shows the most common value.

Question 4: Explain skewness and kurtosis. What does a positive skew imply about the data?
Answer-Skewness-Skewness is a measure of the asymmetry of a dataset's distribution.It tells us whether the data is balanced around the mean or if it leans more to one side.

Kurtosis-Kurtosis measures the "tailedness" or peakedness of a distribution.It tells us how heavy or light the tails of a distribution are, compared to a normal distribution.

positive skew imply -most values are low, with a few extremely high values pulling the mean to the right.
Mean > Median > Mode

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]

```
import statistics

# Given list of numbers
numbers = [12, 15, 12, 18, 19, 12, 20, 22, 19, 19, 24, 24, 24, 26, 28]

# Compute Mean
mean_value = statistics.mean(numbers)

# Compute Median
median_value = statistics.median(numbers)

# Compute Mode
mode_value = statistics.mode(numbers)

# Display the results
print("Mean:", mean_value)
print("Median:", median_value)
print("Mode:", mode_value)
```

```
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]

```
import numpy as np

# Given lists
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)

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

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

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

```
Covariance: 275.0
Correlation Coefficient: 0.995893206467704
```

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]

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

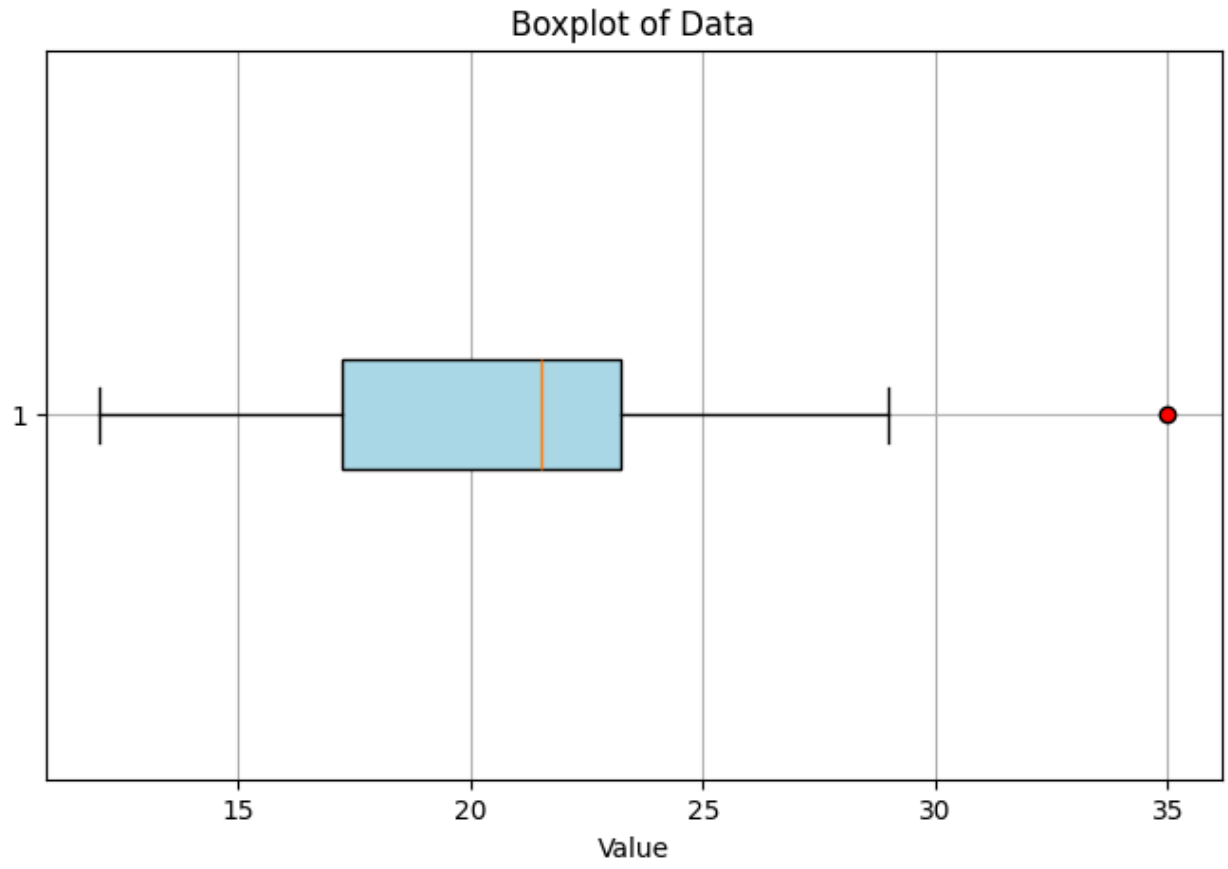
```
# Given data
data = [12, 14, 14, 15, 18, 19, 19, 21, 22, 22, 23, 23, 24, 26, 29, 35]

# Create the boxplot
plt.figure(figsize=(8, 5))
plt.boxplot(data, vert=False, patch_artist=True,
            boxprops=dict(facecolor='lightblue'),
            flierprops=dict(markerfacecolor='red', marker='o'))
plt.title('Boxplot of Data')
plt.xlabel('Value')
plt.grid(True)
plt.show()

# Identify outliers using IQR method
q1 = np.percentile(data, 25)
q3 = np.percentile(data, 75)
iqr = q3 - q1

# Define bounds
lower_bound = q1 - 1.5 * iqr
upper_bound = q3 + 1.5 * iqr
# Identify outliers
outliers = [x for x in data if x < lower_bound or x > upper_bound]

print(f"Q1 (25th percentile): {q1}")
print(f"Q3 (75th percentile): {q3}")
print(f"IQR (Q3 - Q1): {iqr}")
print(f"Lower Bound: {lower_bound}")
print(f"Upper Bound: {upper_bound}")
print(f"Outliers: {outliers}")
```



Q1 (25th percentile): 17.25
Q3 (75th percentile): 23.25
IQR (Q3 - Q1): 6.0
Lower Bound: 8.25
Upper Bound: 32.25
Outliers: [35]

Explanation of the Result

Boxplot Overview:
1.The box shows the interquartile range (IQR) — from Q1 (25th percentile) to Q3 (75th percentile).
2.The line inside the box is the median.
3.The whiskers extend to the smallest and largest values within 1.5 * IQR from Q1 and Q3.
4.Points outside this range are plotted as red dots — these are outliers.

Calculation Results (approximate):
1.Q1 = 18.0
2.Q3 = 24.0

3.IQR = 6.0
4.Lower bound = 18 - 1.5 * 6 = 9.0
5.Upper bound = 24 + 1.5 * 6 = 33.0

Outliers Detected:
1.Any value < 9.0 or > 33.0 is an outlier.
2.Only one value falls outside this range: 35
3.Outlier = [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]

Answer-1. Covariance-

1.Positive covariance → when advertising spend increases, daily sales tend to increase.
2.Negative covariance → when advertising spend increases, daily sales tend to decrease.
3.But: Covariance is not standardized, so it's hard to interpret the strength of the relationship.

2.Correlation-

Range: -1 to 1

1. +1 → perfect positive linear relationship
2. 0 → no linear relationship
3. -1 → perfect negative linear relationship

#PYTHON CODE

```
import numpy as np



# Given data

advertising_spend = [200, 250, 300, 400, 500]

daily_sales = [2200, 2450, 2750, 3200, 4000]
```

```
# Convert to numpy arrays

ad = np.array(advertising_spend)

sales = np.array(daily_sales)

# Compute correlation

correlation = np.corrcoef(ad, sales)[0, 1]



# Output results

print(f"Correlation: {correlation:.4f}")
```

#OUTPUT-

```
Correlation: 0.9936
```

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

1.Mean: Average satisfaction score – shows central tendency.
2.Median: Middle score – useful if data is skewed.
3.Mode: Most common score – shows what most customers rated.
4.Standard Deviation: Measures spread of scores around the mean.
5.Minimum and Maximum: Help spot range and outliers.
6.Count: Total number of responses – checks sample size.

Visualizations-Histogram:
1.Shows frequency distribution.
2.Helps identify skewness, peaks (modes), and spread.

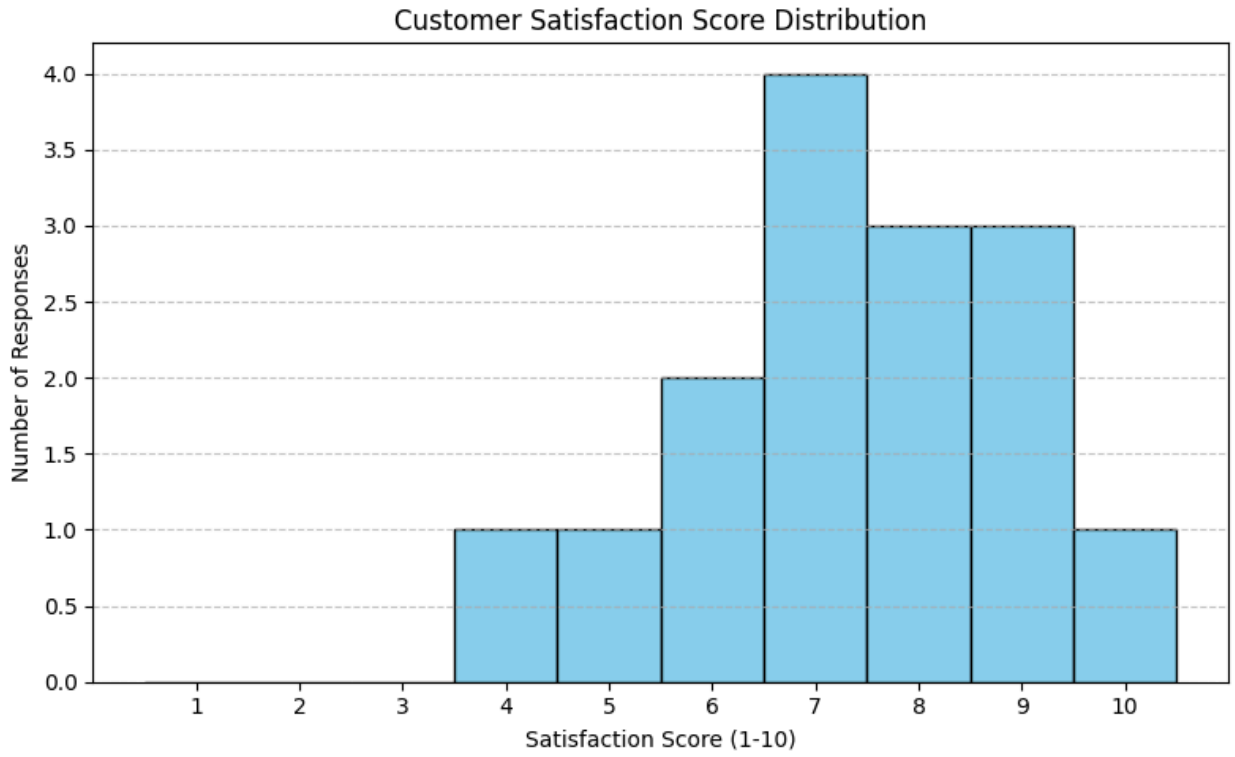
#PYTHON CODE-

```
import matplotlib.pyplot as plt

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

# Create histogram
plt.figure(figsize=(8, 5))
plt.hist(survey_scores, bins=range(1, 12), edgecolor='black',
color='skyblue', align='left')
plt.title('Customer Satisfaction Score Distribution')
plt.xlabel('Satisfaction Score (1-10)')
plt.ylabel('Number of Responses')
plt.xticks(range(1, 11))
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.tight_layout()
plt.show()
```

OUTPUT-


Customer Satisfaction Score Distribution
Number of Responses
Satisfaction Score (1-10)
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
1
2
3
4
5
6
7
8
9
10