**19.Key functions for descriptive statistics**

import pandas as pd

import numpy as np

# Sample dataset

data = pd.Series([12, 15, 14, 10, 18, 19, 25, 14, 17, 21, 13, 15, 19, 18])

# Descriptive statistics

print("Describe:\n", data.describe())

# Mean

print("Mean:", data.mean())

# Median

print("Median:", data.median())

# Mode

print("Mode:", data.mode()[0])  # Pandas handles multiple modes

# Minimum

print("Min:", data.min())

# Maximum

print("Max:", data.max())

# Standard deviation

print("Standard Deviation:", data.std())

# Variance

print("Variance:", data.var())

# Count

print("Count:", data.count())

# Skewness (using Pandas)

print("Skewness:", data.skew())

# Kurtosis (using Pandas)

print("Kurtosis:", data.kurt())

**Output:** Describe:

count    14.000000

mean     16.428571

std       3.936306

min      10.000000

25%      14.000000

50%      16.000000

75%      18.750000

max      25.000000

Mean: 16.428571428571427

Median: 16.0

Mode: 14

Min: 10

Max: 25

Standard Deviation: 3.9363060722593075

Variance: 15.494505494505496

Count: 14

Skewness: 0.5001629167294701

Kurtosis: 0.36076930463530843

**19.implement the web API’s using post,get and update methods**

import json

# Sample data store

data\_store = {

    1: {"name": "Alice", "age": 27},

    2: {"name": "Bob", "age": 35}}

# GET method - Retrieve all records

def get\_users():

 return json.dumps(data\_store, indent=4)

# POST method - Add a new record

def add\_user(user\_data):

  new\_id = max(data\_store.keys()) + 1

  data\_store[new\_id] = user\_data

   return json.dumps({"message": "User added", "user": data\_store[new\_id]}, indent=4)

# PUT method - Update an existing record

def update\_user(user\_id, updated\_data):

    if user\_id in data\_store:

        data\_store[user\_id].update(updated\_data)

        return json.dumps({"message": "User updated", "user": data\_store[user\_id]}, indent=4)

    return json.dumps({"error": "User not found"}, indent=4)

# Example usage

print("GET users:\n", get\_users())

print("\nPOST new user:\n", add\_user({"name": "Charlie", "age": 22}))

print("\nPUT update user:\n", update\_user(1, {"age": 26}))

**Output:**

GET users:

 {

    "1": {

        "name": "Alice",

        "age": 27  },

    "2": {

        "name": "Bob",

        "age": 35}}

POST new user:

 { "message": "User added",

    "user": {

        "name": "Charlie",

        "age": 22 }}

  PUT update user:

 {  "message": "User updated",

    "user": {

        "name": "Alice",

        "age": 26 }}