Assignment: Bitcoin Data Analysis Using Python

Objective:
You are required to demonstrate your ability to manipulate data using Python, including filtering, looping, and conditional statements. The task involves creating a data frame, adding the data, performing exploratory data analysis (EDA), and applying additional operations.

Don't use ChatGPT or any other AI tools to complete this assignment. If found then your candidature will be eliminated.

1. Create a DataFrame:

Create a Pandas DataFrame with the following columns:

- `Date`: Date of Bitcoin prices
- `Open`: Opening price of Bitcoin on that day
- `High`: Highest price of Bitcoin on that day
- `Low`: Lowest price of Bitcoin on that day
- `Close`: Closing price of Bitcoin on that day
- `Volume`: Trading volume of Bitcoin on that day
- `Market Cap`: Market capitalization of Bitcoin on that day

2. Add Data:

- Add least 30 days of historical Bitcoin dummy data in the DataFrame.

3. Perform Exploratory Data Analysis (EDA):

- Display the descriptive statistics of the dataset (mean, median, standard deviation, etc.).
- Plot a time series of the `Close` price of Bitcoin.
- Plot the distribution of the `Volume`.
- Create a pair plot to show the relationship between `Open`, `Close`, `High`, and `Low`.
- Calculate the correlation between `Open`, `High`, `Low`, `Close`, and `Volume`.

4. Apply Filtering Conditions:

- Filter the DataFrame to display only the rows where:
- The `Close` price is greater than the `Open` price (indicating a positive gain).
- The trading `Volume` was higher than the average trading volume for that period.

5. For Loop and Conditional Statements:

- Use a for loop to iterate over the rows of the DataFrame and calculate the daily price change (difference between `Close` and `Open`), adding this as a new column `Price Change`.
- Apply an if-else condition to create another column `Price Trend`, where:
    - If the `Close` price is higher than the `Open` price, set the value to "Up."
    - Otherwise, set the value to "Down."

Deliverables:

- A Python script or Jupyter Notebook that includes: