Assignment 1:
Getting Started with Numpy and Pandas

Instructions

- Complete all tasks in a Jupyter Notebook (.ipynb).
- Add comments and explanations for each task.
- Submit your completed notebook file with all outputs visible.

Tasks

Task 1: Array Basics (Numpy)

1. Create Arrays

- Generate the following numpy arrays:
    1. A 1D array of integers from 0 to 9.
    2. A 3x3 matrix of random integers between 1 and 100.

2. Reshape Arrays

- Reshape the 1D array from (1.1) into a 2D array with 2 rows and 5 columns.

3. Array Attributes

- Display the following attributes of any one array:
    - Shape
    - Size
    - Data type

Task 2: Data Manipulation (Pandas)

1. DataFrame Creation

- Create a pandas DataFrame with random data (5 rows and 3 columns).
- Label the columns as "A," "B," and "C."

2. Basic Operations

- Display the first two rows of the DataFrame.
- Calculate the mean of each column.
- Add a new column "D" which is the sum of columns "A" and "B."

3. Filtering and Sorting

- Filter rows where column "A" values are greater than 50.
- Sort the DataFrame by column "C" in descending order.

Task 3: Numerical Computations (Numpy)

1. Element-Wise Operations

- Create two 1D arrays of the same size with random integers.
- Perform addition, subtraction, and multiplication on the arrays.

2. Aggregate Functions

- Calculate the mean, sum, and standard deviation of a 1D array.

3. Matrix Multiplication

- Create a 2x3 matrix and a 3x2 matrix.
- Perform matrix multiplication using `np.dot()`.

Task 4:

1. Temperature Analysis

- Create a 1D array of 7 temperatures (one for each day of the week).
- Find:
    - The average temperature.
    - Days with temperatures above the weekly average.

2. DataFrame Operations

- Using pandas, create a DataFrame with two columns:
    - "Day" (Monday to Sunday)
    - "Temperature" (use the numpy array from Task 4.1).
- Calculate the day with the highest temperature.

3. Stock Price simulation

- Create a pandas DataFrame with random stock prices for 5 days.
- Add a column to calculate the percentage change in price day-to-day.

Bonus Question:

Multi-Dimensional Data Analysis

- Create a 3D array representing sales data for 3 products over 4 weeks (rows) and 7 days per week (columns).
- Calculate the total sales for each product across all weeks and days.
- Determine the day with the highest sales for each product in the dataset.
- Identify the week with the lowest total sales overall.

Submission Requirements

1. Submit a Jupyter Notebook file (.ipynb) containing:
    - Completed code for all tasks.
    - Explanations and comments.
2. Ensure all outputs are visible.

Good luck!