Practice: NumPy and Pandas for Beginners

Part 1: NumPy Practice

1. Student Marks Analyzer

You are helping a teacher check exam results.
Create a NumPy array with 10 random integers between 0 and 100 (student scores).

Find:
- the highest score
- the lowest score
- the average score

Hint: use np.random.randint(), arr.max(), arr.min(), arr.mean()

2. Warehouse Weights

A warehouse stores item weights (in kg):
arr = np.array([45, 60, 55, 30, 90, 75, 40, 50])

Find and print:
- all items heavier than 50 kg
- the sorted weights
- the index of the heaviest item

Hint: use arr[arr > 50], np.sort(arr), arr.argmax()

3. Temperature Report

A weather station collects temperatures (°C) from 4 zones.
Generate random integers between 20 and 40 for a 4×4 grid.

Then find:
- the average temperature of each zone (row)
- the highest temperature in the grid
- replace all temperatures above 35 with 35

4. Traffic Data

A city records the number of cars passing 5 intersections in 3 time slots (morning, afternoon, night).
Create a 3×5 array of random integers between 0 and 50.

Then find:
- total cars per intersection (sum by column)
- total cars per time slot (sum by row)

5. Fruit Store Stock

You sell fruits in your store. Stock levels (kg) for 5 days are:
stock = np.array([12, 15, 9, 20, 18])

A new delivery adds 5 kg to each stock.
Then 10% of all fruits get spoiled.

Find the final stock levels.

Hint: use broadcasting and simple math.

Part 2: Pandas Practice

1. Student Report Analyzer

You are given this data:

```
import pandas as pd

data = {
        'Name': ['Rahim', 'Karim', 'Selina', 'Babul', 'Mitu'],
        'Math': [88, 75, 95, 60, 80],
        'English': [78, 85, 90, 70, 88],
        'Science': [92, 70, 85, 65, 90]
}
df = pd.DataFrame(data)
```

Tasks:
1. Display only the first 3 rows.
2. Print the column names and index values.

2. Shop Sales Data

You run a small electronics shop. The data is stored like this:

```
data = {
        'Product': ['Phone', 'Laptop', 'Tablet', 'Monitor', 'Mouse'],
        'Price': [700, 1200, 300, 250, 50],
        'Units_Sold': [10, 5, 8, 6, 20]
}
df = pd.DataFrame(data)
```

Tasks:
1. Display only the 'Product' and 'Price' columns.
2. Access the 3rd row using both iloc and loc.
3. Show the top 3 products with the highest revenue.

3. Attendance Tracker

Here's an attendance sheet:

```
data = {
        'Student': ['Amin', 'Tina', 'Rafi', 'Nadia', 'Sabbir'],
        'Days_Present': [20, 18, 22, 15, 25],
        'Days_Absent': [2, 4, 1, 7, 0]
}
df = pd.DataFrame(data)
```

Tasks:
1. Make 'Student' the index column.
2. Drop the 'Days_Absent' column.

4. Filtering Employee Data

Employee record:

```
data = {
        'Name': ['Rafiq', 'Nadia', 'Tuhin', 'Rima', 'Faruk'],
        'Department': ['HR', 'IT', 'IT', 'Sales', 'Sales'],
        'Salary': [40000, 55000, 52000, 45000, 47000],
        'Experience': [2, 5, 4, 3, 6]
}
df = pd.DataFrame(data)
```

Tasks:
1. Show employees with Salary > 45000.
2. Show employees from the IT department only.

3. Show employees with Salary > 45000 and Experience > 3.
4. Sort the employees by Experience (ascending).

5. City Weather Dashboard

```
data = {
        'City': ['Dhaka', 'Sylhet', 'Chittagong', 'Rajshahi', 'Khulna'],
        'Temp': [32, 28, 30, 35, 33],
        'Humidity': [60, 75, 70, 55, 65]
}
df = pd.DataFrame(data)
```

Tasks:
1. Show only the cities where FeelsLike > 33.
2. Sort the DataFrame by FeelsLike (descending).
3. Find which city has the highest humidity.