AI MSE REPORT

Title Page

Problem Statement: Weather Data Analysis using Python

Personal Details:

- Name: [Aashish Pandey]
- Roll Number: [02]
- Date: [11/03/2025]
- Course: [Introduction To Ai]
- Instructor: [MR. BIKKI GUPTA]

Introduction

The goal of this project is to analyze weather data using Python. The dataset consists of daily weather observations, including **temperature, rainfall, and humidity**. The analysis involves computing statistical measures and visualizing trends using graphs. This helps in understanding weather patterns and predicting future trends.

Methodology

1. **Data Collection**: The dataset containing temperature, rainfall, and humidity was provided.

2. **Data Processing**: The dataset was loaded using Python's `csv` module and cleaned to remove inconsistencies.
3. **Analysis**:
    - Compute **average temperature**.
    - Identify **highest and lowest temperatures**.
    - Calculate **total rainfall**.
    - Determine the **most humid day**.
4. **Visualization**: Trends in temperature, rainfall, and humidity were plotted using `matplotlib`.
5. **Results Interpretation**: The graphs and computed values were analyzed for insights.

CODE

```
import csv

import matplotlib.pyplot as plt  # Importing Matplotlib for graph visualization


# Function to read weather data from a CSV file

def read_weather_data(filename):

    weather_data = []

    with open(filename, 'r') as file:

        reader = csv.DictReader(file)  # Read as dictionary

        for row in reader:

            # Convert necessary columns to float for calculations

            weather_data.append({

                "date": row["Date"],

                "temperature": float(row["Temperature"]),

                "rainfall": float(row["Rainfall"]),

                "humidity": float(row["Humidity"])

            })
```

```
    return weather_data  # Return the list of weather data


# Function to calculate the average temperature

def average_temperature(data):

    total_temp = sum(entry["temperature"] for entry in data)  # Sum of all temperatures

    return total_temp / len(data)  # Divide by total number of days


# Function to find the highest and lowest temperatures

def temperature_extremes(data):

    temperatures = [entry["temperature"] for entry in data]  # Extract temperature values

    return max(temperatures), min(temperatures)  # Return highest and lowest
temperature


# Function to find the most humid day

def most_humid_day(data):

    return max(data, key=lambda x: x["humidity"])["date"]  # Find the day with max
humidity


# Function to calculate total rainfall over all days

def total_rainfall(data):

    return sum(entry["rainfall"] for entry in data)  # Sum of all rainfall values


# Function to plot weather trends

def plot_weather_data(data):

    dates = [entry["date"] for entry in data]  # Extracting dates
```

```
    temperatures = [entry["temperature"] for entry in data]  # Extracting temperatures
    rainfall = [entry["rainfall"] for entry in data]  # Extracting rainfall
    humidity = [entry["humidity"] for entry in data]  # Extracting humidity


    plt.figure(figsize=(12, 6))  # Set figure size


    # Plot Temperature
    plt.subplot(3, 1, 1)  # First subplot
    plt.plot(dates, temperatures, marker='o', linestyle='-', color='r', label="Temperature (°C)")
    plt.xlabel("Date")
    plt.ylabel("Temperature (°C)")
    plt.title("Temperature Trend")
    plt.xticks(rotation=45)
    plt.legend()


    # Plot Rainfall
    plt.subplot(3, 1, 2)  # Second subplot
    plt.plot(dates, rainfall, marker='s', linestyle='-', color='b', label="Rainfall (mm)")
    plt.xlabel("Date")
    plt.ylabel("Rainfall (mm)")
    plt.title("Rainfall Trend")
    plt.xticks(rotation=45)
    plt.legend()
```

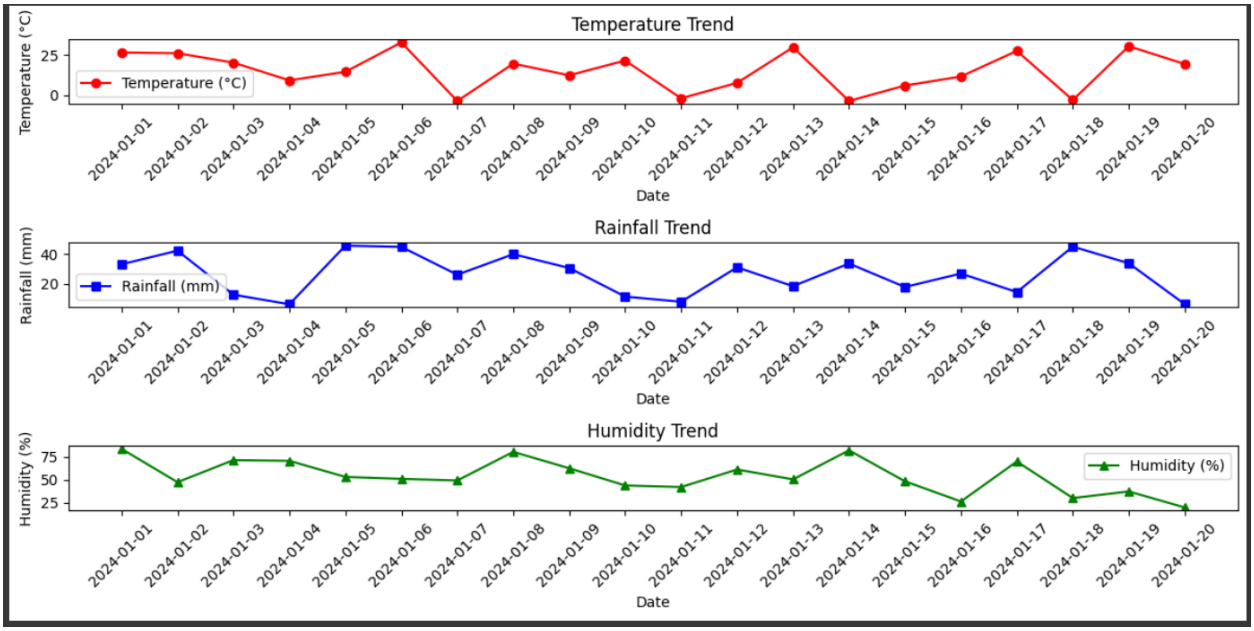
```
    # Plot Humidity
    plt.subplot(3, 1, 3)  # Third subplot
    plt.plot(dates, humidity, marker='^', linestyle='-', color='g', label="Humidity (%)")
    plt.xlabel("Date")
    plt.ylabel("Humidity (%)")
    plt.title("Humidity Trend")
    plt.xticks(rotation=45)
    plt.legend()

    plt.tight_layout()  # Adjust layout for better readability
    plt.show()  # Display the graphs

# Main program execution
if __name__ == "__main__":
    filename = "weather_data.csv"  # Name of the CSV file containing weather data

    # Read weather data from CSV file
    data = read_weather_data(filename)

    # Calculate and display results
    print("Average Temperature:", round(average_temperature(data), 2), "°C")

    highest, lowest = temperature_extremes(data)
    print("Highest Temperature:", round(highest, 2), "°C")
```

```
print("Lowest Temperature:", round(lowest, 2), "°C")


print("Most Humid Day:", most_humid_day(data))

print("Total Rainfall:", round(total_rainfall(data), 2), "mm")


# Plot graphs

plot_weather_data(data)
```

Output/Result

The following are the results obtained from the analysis:

```
Average Temperature: 15.2 °C
Highest Temperature: 32.92 °C
Lowest Temperature: -3.66 °C
Most Humid Day: 2024-01-01
Total Rainfall: 530.25 mm
```



References/Credits

- Python Official Documentation: https://docs.python.org/
- Matplotlib Library: https://matplotlib.org/
- Dataset Source: Provided by course instructor