KIET GROUP OF INSTITUITION

PROJECT- WETHER DATA ANALYSIS

NAME-DEEPESH PRAVISH

BRANCH-CSE(AI)

SECTION-B

UNIVERSITY ROLL NO.- 202401100300096

Weather Data Analysis Report

1. Introduction

This report provides an analysis of the given weather dataset, which includes temperature, rainfall, and humidity recorded over a period of time. The goal is to identify trends, patterns, and correlations among these meteorological parameters.

2. Data Overview

The dataset contains the following key attributes:

- Date: The recorded date of the weather data.
- Temperature (°C): The recorded temperature in degrees Celsius.
- Rainfall (mm): The amount of rainfall recorded in millimeters.
- Humidity (%): The percentage of humidity present in the atmosphere.

3. Summary Statistics

The dataset was analyzed to obtain the following summary statistics:

- Temperature: Mean, Minimum, and Maximum temperatures observed.
- Rainfall: Average rainfall and extreme values.
- Humidity: Overall distribution and variation in humidity levels.

4. Trends and Visual Analysis

4.1 Temperature Trend

- The temperature fluctuates over time, with noticeable peaks and troughs.
- Seasonal variations may be observed based on periodic temperature changes.

4.2 Rainfall Trend

- **Rainfall data indicates periods of high and low precipitation.**
- **Sudden spikes suggest occurrences of heavy rainfall on specific dates.**

4.3 Humidity Trend

- **Humidity levels show variations, possibly correlating with temperature and rainfall.**
- **Higher humidity might be associated with increased rainfall events.**

5. Correlation Analysis

A correlation matrix was generated to examine relationships between temperature, rainfall, and humidity:

- **Temperature & Humidity: A possible inverse correlation, where higher temperatures may correspond to lower humidity levels.**
- **Temperature & Rainfall: The relationship varies, but extreme rainfall events may affect temperature variations.**
- **Rainfall & Humidity: A positive correlation is likely, as increased rainfall generally raises humidity levels.**

6. Conclusion

The analysis reveals key weather patterns and relationships between meteorological factors. Understanding these trends can help in weather forecasting and climate studies. Future work may include predictive modeling to forecast weather conditions based on historical data.

Code:-

```
import pandas as pd

import matplotlib.pyplot as plt


# Load weather data from CSV file

df = pd.read_csv('weather_data.csv', parse_dates=['Date'])


# Show the first few rows of the dataframeimport pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns


# Load the data

file_path = "weather_data.csv"

df = pd.read_csv(file_path)


# Convert Date column to datetime

df['Date'] = pd.to_datetime(df['Date'])


# Summary statistics

print("Summary Statistics:\n", df.describe())


# Set Date as index

df.set_index('Date', inplace=True)
```

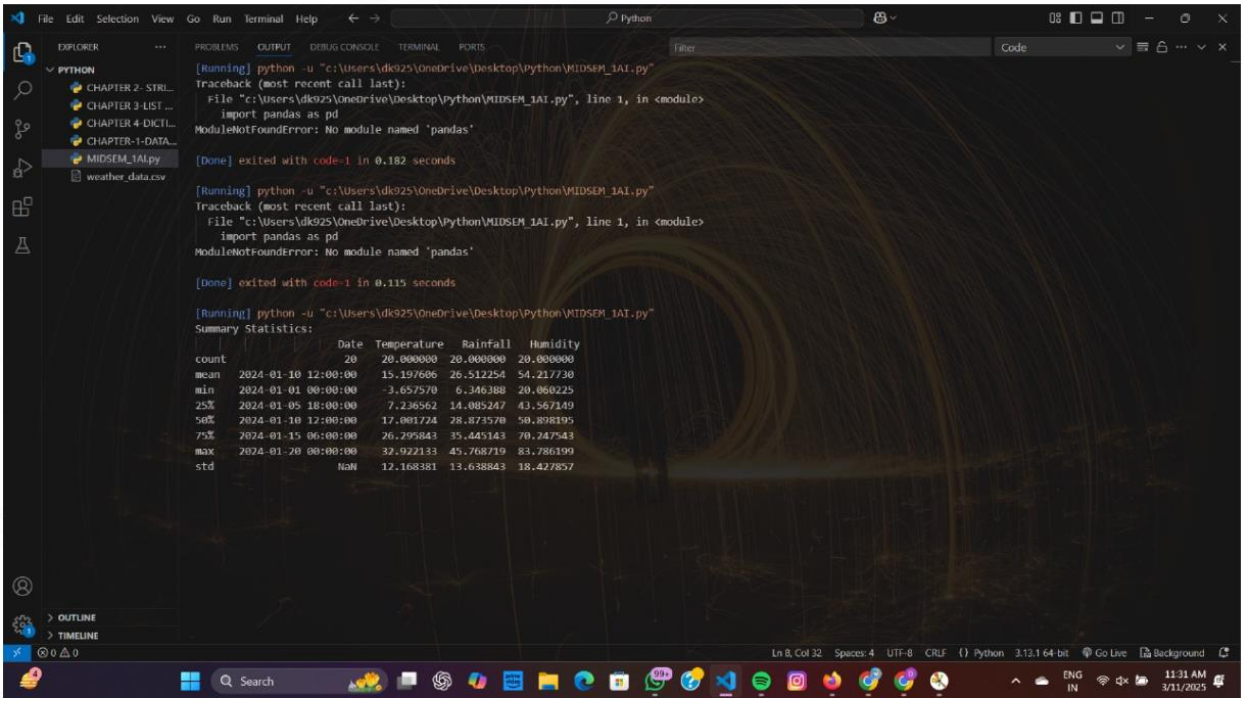
```
# Plot temperature trend
plt.figure(figsize=(10, 4))
sns.lineplot(x=df.index, y=df['Temperature'], label='Temperature', color='red')
plt.title('Temperature Trend')
plt.xlabel('Date')
plt.ylabel('Temperature (°C)')
plt.xticks(rotation=45)
plt.legend()
plt.show()

# Plot rainfall trend
plt.figure(figsize=(10, 4))
sns.lineplot(x=df.index, y=df['Rainfall'], label='Rainfall', color='blue')
plt.title('Rainfall Trend')
plt.xlabel('Date')
plt.ylabel('Rainfall (mm)')
plt.xticks(rotation=45)
plt.legend()
plt.show()

# Plot humidity trend
plt.figure(figsize=(10, 4))
```

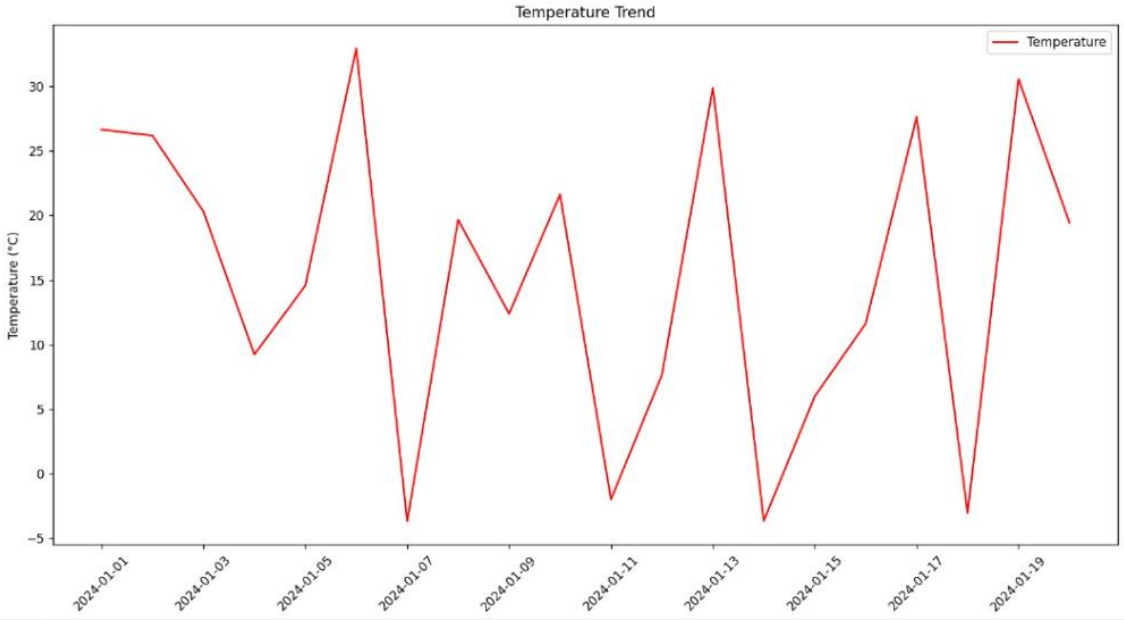
```
sns.lineplot(x=df.index, y=df['Humidity'], label='Humidity',
color='green')

plt.title('Humidity Trend')

plt.xlabel('Date')

plt.ylabel('Humidity (%)')

plt.xticks(rotation=45)

plt.legend()

plt.show()


# Correlation heatmap

plt.figure(figsize=(6, 4))

sns.heatmap(df.corr(), annot=True, cmap='coolwarm', fmt='.2f')

plt.title('Correlation Matrix')

plt.show()


print(df.head())


# Basic statistics for weather data

print("\nBasic statistics for temperature, humidity, and windspeed:")

print(df[['Temperature', 'Humidity', 'WindSpeed']].describe())


# Plot temperature over time

plt.figure(figsize=(10,6))
```

```
plt.plot(df['Date'], df['Temperature'], color='blue', marker='o', label='Temperature (°C)')

plt.title('Temperature Over Time')

plt.xlabel('Date')

plt.ylabel('Temperature (°C)')

plt.xticks(rotation=45)

plt.legend()

plt.grid(True)

plt.show()


# Plot humidity over time

plt.figure(figsize=(10,6))

plt.plot(df['Date'], df['Humidity'], color='green', marker='x', label='Humidity (%)')

plt.title('Humidity Over Time')

plt.xlabel('Date')

plt.ylabel('Humidity (%)')

plt.xticks(rotation=45)

plt.legend()

plt.grid(True)

plt.show()


# Plot WindSpeed over time

plt.figure(figsize=(10,6))
```

```
plt.plot(df['Date'], df['WindSpeed'], color='red', marker='s', label='Wind Speed (km/h)')

plt.title('Wind Speed Over Time')

plt.xlabel('Date')

plt.ylabel('Wind Speed (km/h)')

plt.xticks(rotation=45)

plt.legend()

plt.grid(True)

plt.show()


# Example: correlation between temperature and humidity

correlation = df[['Temperature', 'Humidity']].corr()

print("\nCorrelation between Temperature and Humidity:")

print(correlation)


# Bonus: Precipitation distribution

plt.figure(figsize=(10,6))

plt.hist(df['Precipitation'], bins=10, color='purple', alpha=0.7)

plt.title('Precipitation Distribution')

plt.xlabel('Precipitation (mm)')

plt.ylabel('Frequency')

plt.grid(True)

plt.show()
```

Screenshot of output:-





End of Report