Project Semester January–April 2025

DATA SCIENCE MINOR PROJECT REPORT

ON

Exploratory Data Analysis and Visualization of Climate Change

DATA SCIENCE TOOLBOX: PYTHON PROGRAMMING

COURSE CODE: INT375

B. TECH COMPUTER SCIENCE AND ENGINEERING

LOVELY PROFESSIONAL UNIVERSITY

PHAGWARA, PUNJAB

PROJECT SUBMITTED BY:

Yogita Yadav (12304886)

Section: K23RT

Roll No.: 48

PROJECT SUBMITTED TO:

Ms. Maneet Kaur (15709)

DECLARATION

I, **Yogita Yadav**, student of B.Tech – Computer Science and Engineering (Section K23RT) at Lovely Professional University, Punjab, hereby declare that all the information furnished in this project report titled:

“Exploratory data analysis and visualisation of climate change”

is based on my own intensive work and is genuine. The content of this report has not been submitted to any other university or institution for the award of any degree or diploma.

Date: 12-04-2025
Registration No.: 12304886
Name: Yogita Yadav

CERTIFICATE

This is to certify that **Ms. Yogita Yadav,** bearing Registration No. **12304886**, has successfully completed the **INT375** – Python Programming project titled:

“Exploratory data analysis and visualisation of climate change”

under my guidance and supervision. To the best of my knowledge, the present work is the result of her original development, effort, and study. This project has been carried out as a part of the curriculum prescribed by Lovely Professional University, Phagwara for the Project Semester **January–April 2025.**

SignatureandNameoftheSupervisor
Ms.Maneet Kaur

ACKNOWLEDGEMENT

I sincerely thank Ms. Maneet Kaur , for her guidance and support throughout this project. I also thank the faculty of the CSE Department at Lovely Professional University for providing the necessary resources and assistance.

Tanvi
Reg. No.: 12304886

1. **INTRODUCTION:**
   Climate change has emerged as one of the most pressing global challenges, with far-reaching impacts on ecosystems, economies, and public health. Understanding historical climate patterns is essential for predicting future trends and developing effective mitigation strategies. This project focuses on analyzing a cleaned dataset containing historical climate data up to the year 2000. The dataset includes key indicators such as temperature variations, CO_2 levels, and other environmental metrics across different regions and time periods. Using data science techniques such as exploratory data analysis (EDA), statistical testing, and visualization, the objective is to uncover trends, assess the extent of climate change over time, and identify potential correlations between various environmental factors. The goal of the project is not only to conduct thorough technical analysis but also to present the findings in a visually interpretable manner, enhancing public understanding and supporting data-driven environmental policy decisions.

2. **SOURCE OF DATASET:**

https://www.data.gov.in/

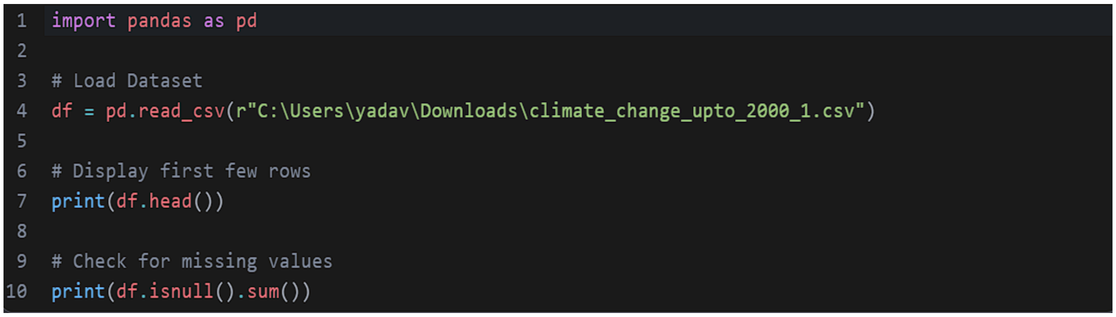
3. EDA PROCESS

Exploratory Data Analysis (EDA) is a fundamental step in any data science project. In this project, EDA was performed to understand the structure, patterns, trends, and anomalies within the Climate Change dataset.

The key steps in the EDA process were as follows:

3.1 Data Loading and Inspection

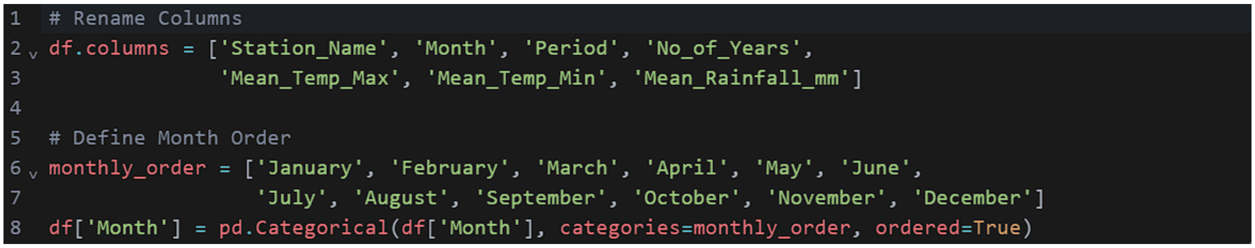
The first step in any data analysis project is loading and inspecting the dataset. Using Python's **pandas** library, we load the CSV file and examine its structure.



```python
import pandas as pd

# Load Dataset
df = pd.read_csv(r"C:\Users\yadav\Downloads\climate_change_upto_2000_1.csv")

# Display first few rows
print(df.head())

# Check for missing values
print(df.isnull().sum())
```

3.2. Data Cleaning and Month Ordering

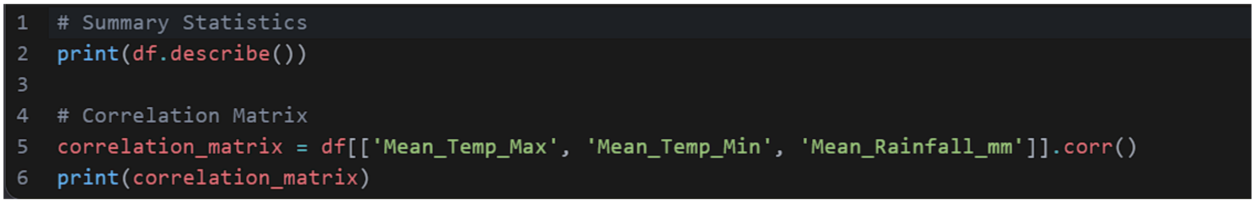
To ensure consistency, we renamed the columns for easier access and converted the "Month" column into a categorical variable with a predefined order (January to December).



```python
# Rename Columns
df.columns = ['Station_Name', 'Month', 'Period', 'No_of_Years',
              'Mean_Temp_Max', 'Mean_Temp_Min', 'Mean_Rainfall_mm']

# Define Month Order
monthly_order = ['January', 'February', 'March', 'April', 'May', 'June',
                 'July', 'August', 'September', 'October', 'November', 'December']
df['Month'] = pd.Categorical(df['Month'], categories=monthly_order, ordered=True)
```

3.3. Basic Statistical Exploration

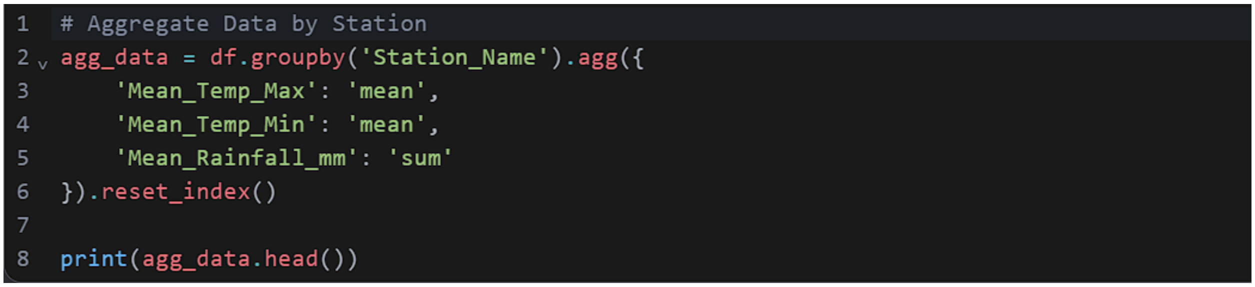
We performed basic statistical analysis to understand the distribution of temperature and rainfall data.



```python
# Summary Statistics
print(df.describe())

# Correlation Matrix
correlation_matrix = df[['Mean_Temp_Max', 'Mean_Temp_Min', 'Mean_Rainfall_mm']].corr()
print(correlation_matrix)
```

3.4. Grouping and Aggregation by Station

Data was grouped by a station to calculate aggregate statistics such as average maximum and minimum temperatures and total rainfall.



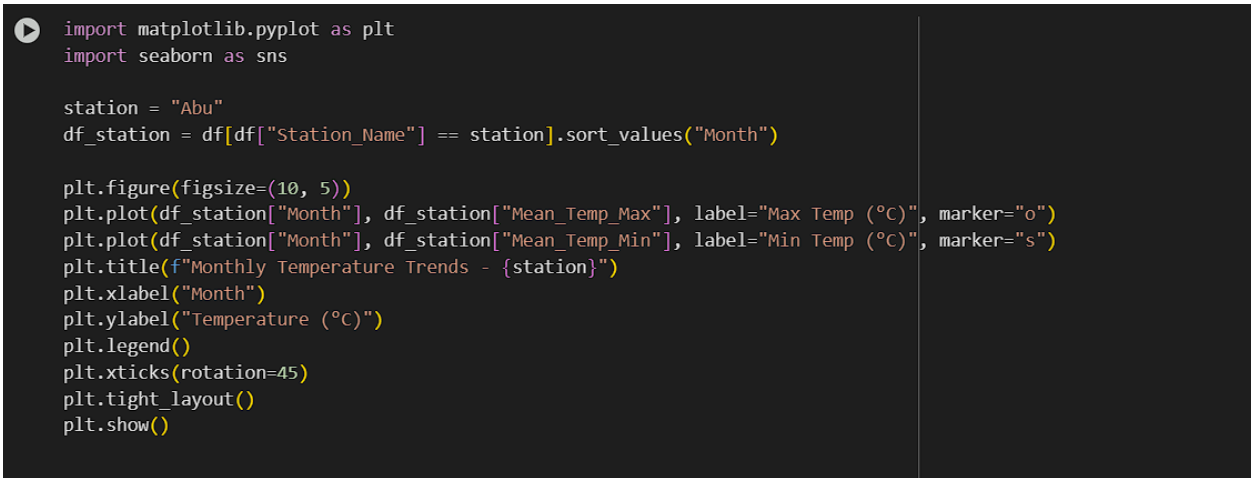
```python
# Aggregate Data by Station
agg_data = df.groupby('Station_Name').agg({
    'Mean_Temp_Max': 'mean',
    'Mean_Temp_Min': 'mean',
    'Mean_Rainfall_mm': 'sum'
}).reset_index()

print(agg_data.head())
```

3.5. Visual Exploration Strategy

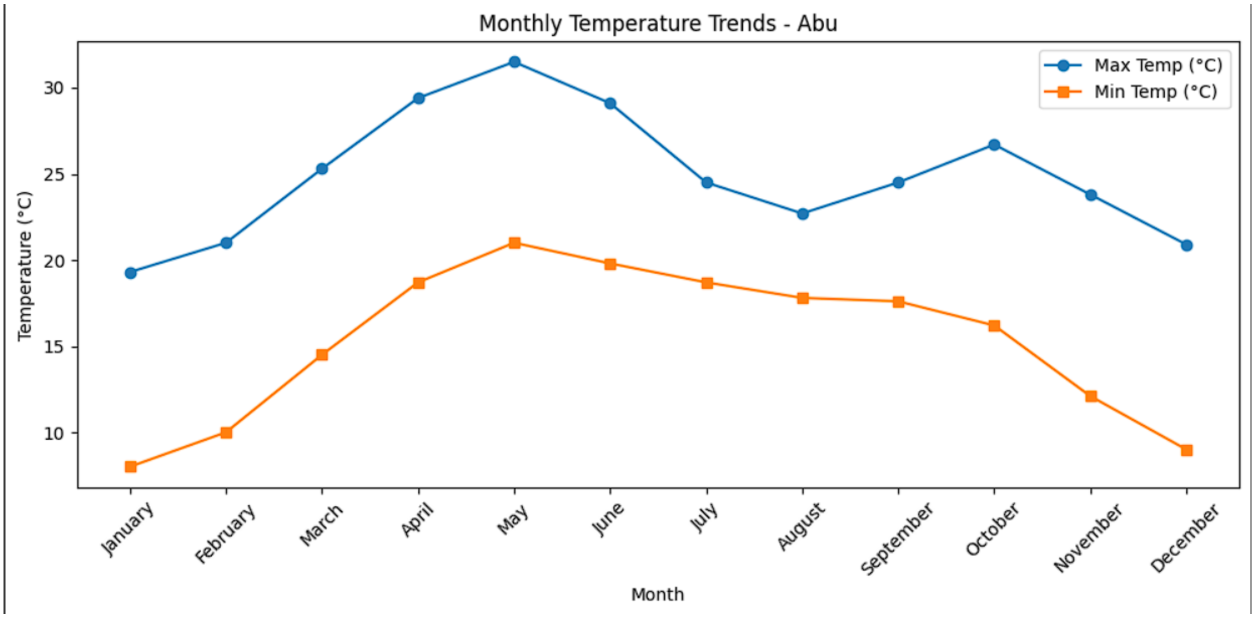
Visualization plays a crucial role in understanding trends and patterns. We employed various plots such as line charts, bar charts, heatmaps, boxplots, and scatter plots to explore relationships between variables.

4.1 Temperature Trends by Station (Line Plot)

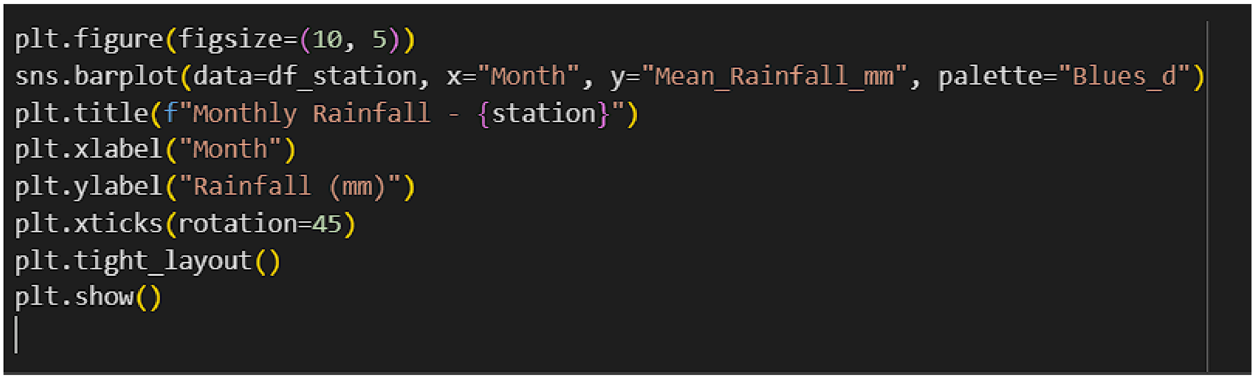
A line plot was created to visualize monthly temperature trends for a specific station (e.g., Abu).



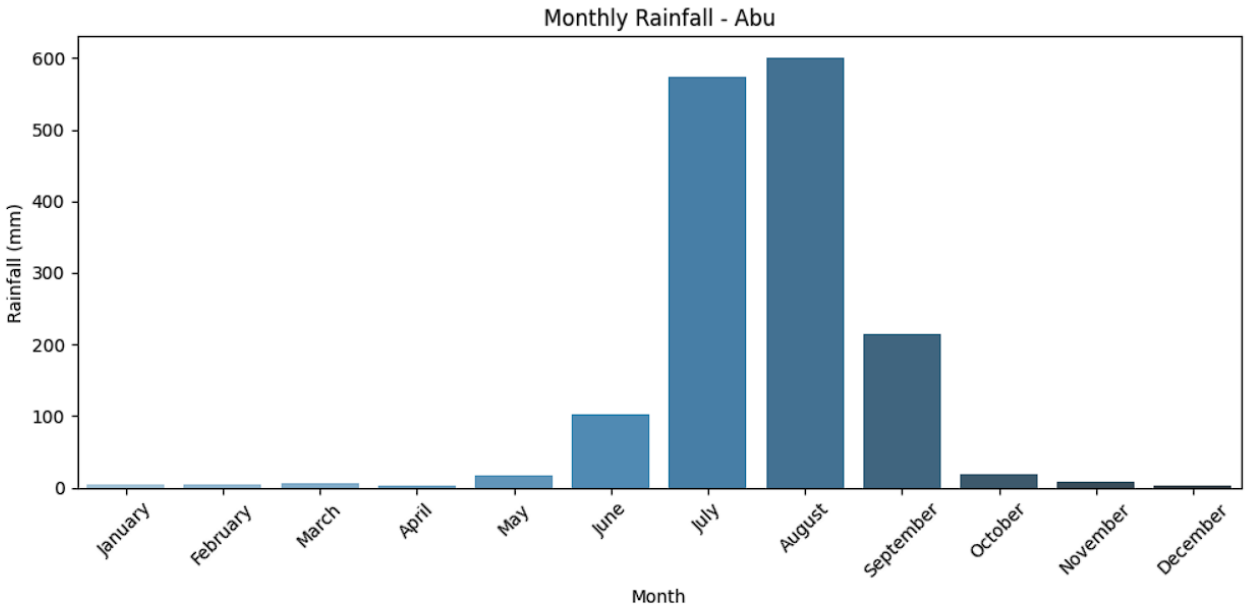
```python
import matplotlib.pyplot as plt
import seaborn as sns

station = "Abu"
df_station = df[df["Station_Name"] == station].sort_values("Month")

plt.figure(figsize=(10, 5))
plt.plot(df_station["Month"], df_station["Mean_Temp_Max"], label="Max Temp (°C)", marker="o")
plt.plot(df_station["Month"], df_station["Mean_Temp_Min"], label="Min Temp (°C)", marker="s")
plt.title(f"Monthly Temperature Trends - {station}")
plt.xlabel("Month")
plt.ylabel("Temperature (°C)")
plt.legend()
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



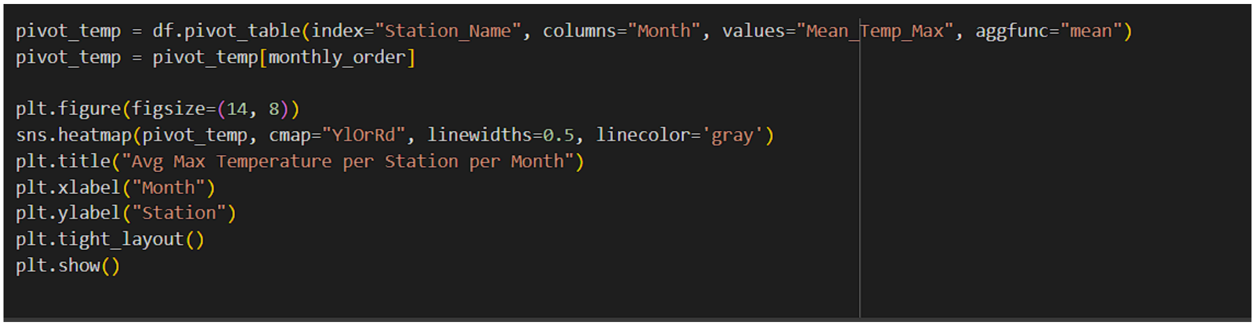
4.2 Bar plot of mean rainfall for a station



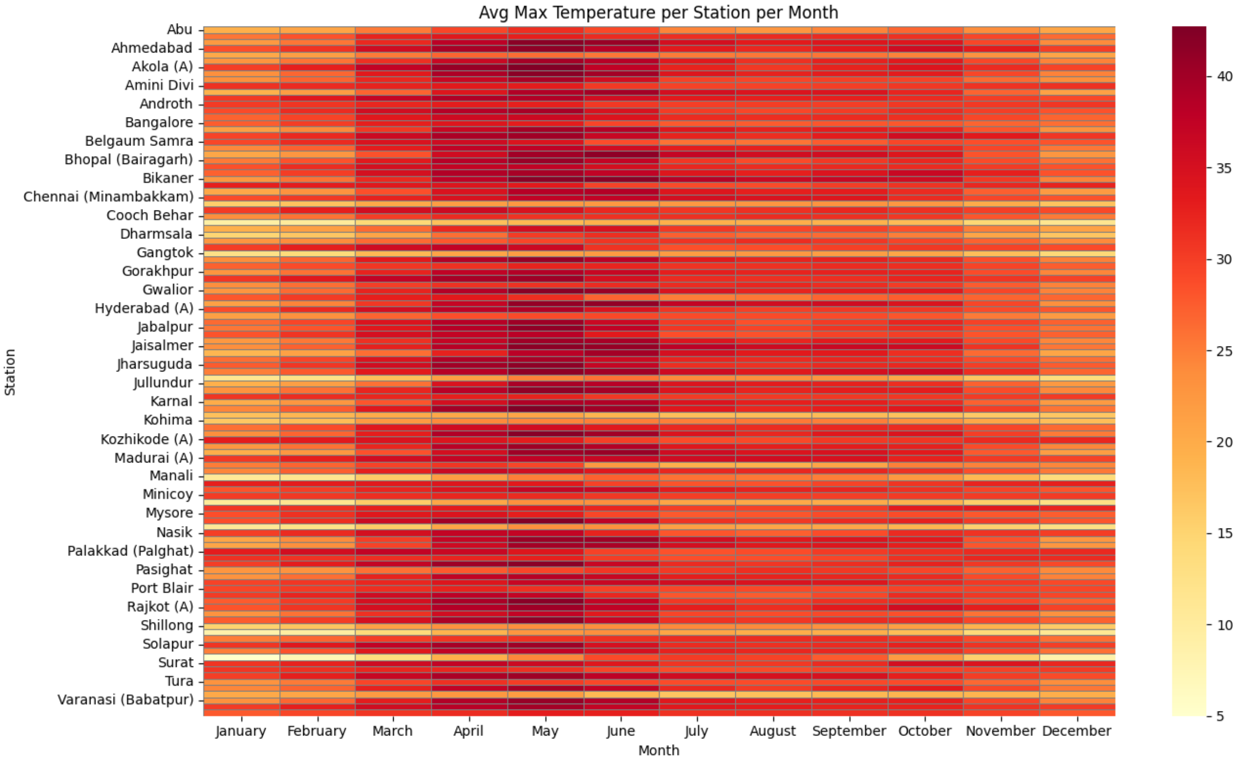
```
plt.figure(figsize=(10, 5))
sns.barplot(data=df_station, x="Month", y="Mean_Rainfall_mm", palette="Blues_d")
plt.title(f"Monthly Rainfall - {station}")
plt.xlabel("Month")
plt.ylabel("Rainfall (mm)")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



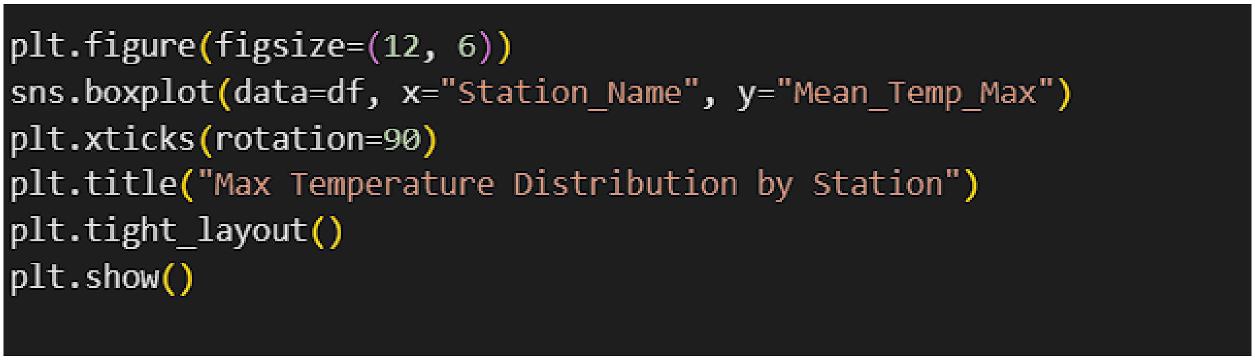
4.3 Heatmap of average max temp per station per month



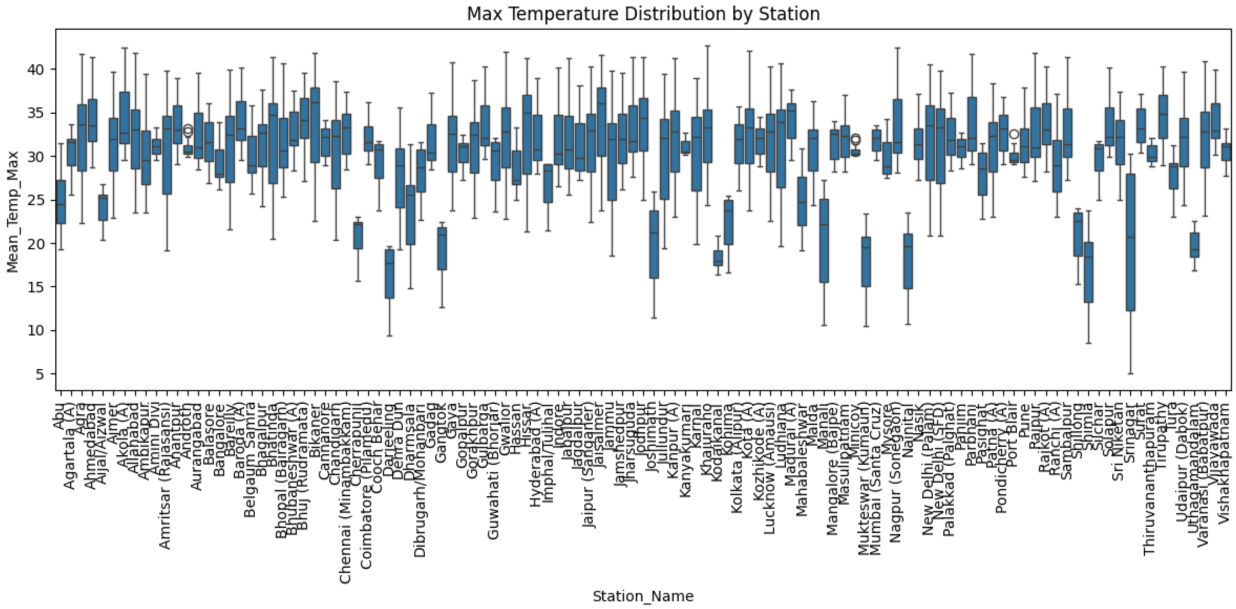
```
pivot_temp = df.pivot_table(index="Station_Name", columns="Month", values="Mean_Temp_Max", aggfunc="mean")
pivot_temp = pivot_temp[monthly_order]

plt.figure(figsize=(14, 8))
sns.heatmap(pivot_temp, cmap="YlOrRd", linewidths=0.5, linecolor='gray')
plt.title("Avg Max Temperature per Station per Month")
plt.xlabel("Month")
plt.ylabel("Station")
plt.tight_layout()
plt.show()
```



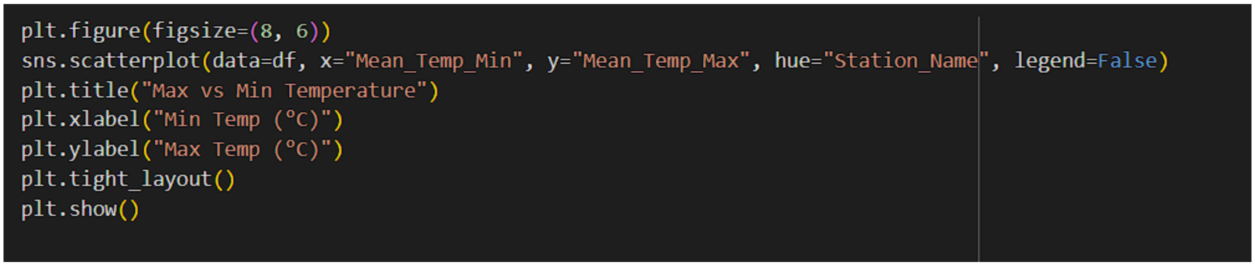
4. Boxplot of max temperatures across all stations

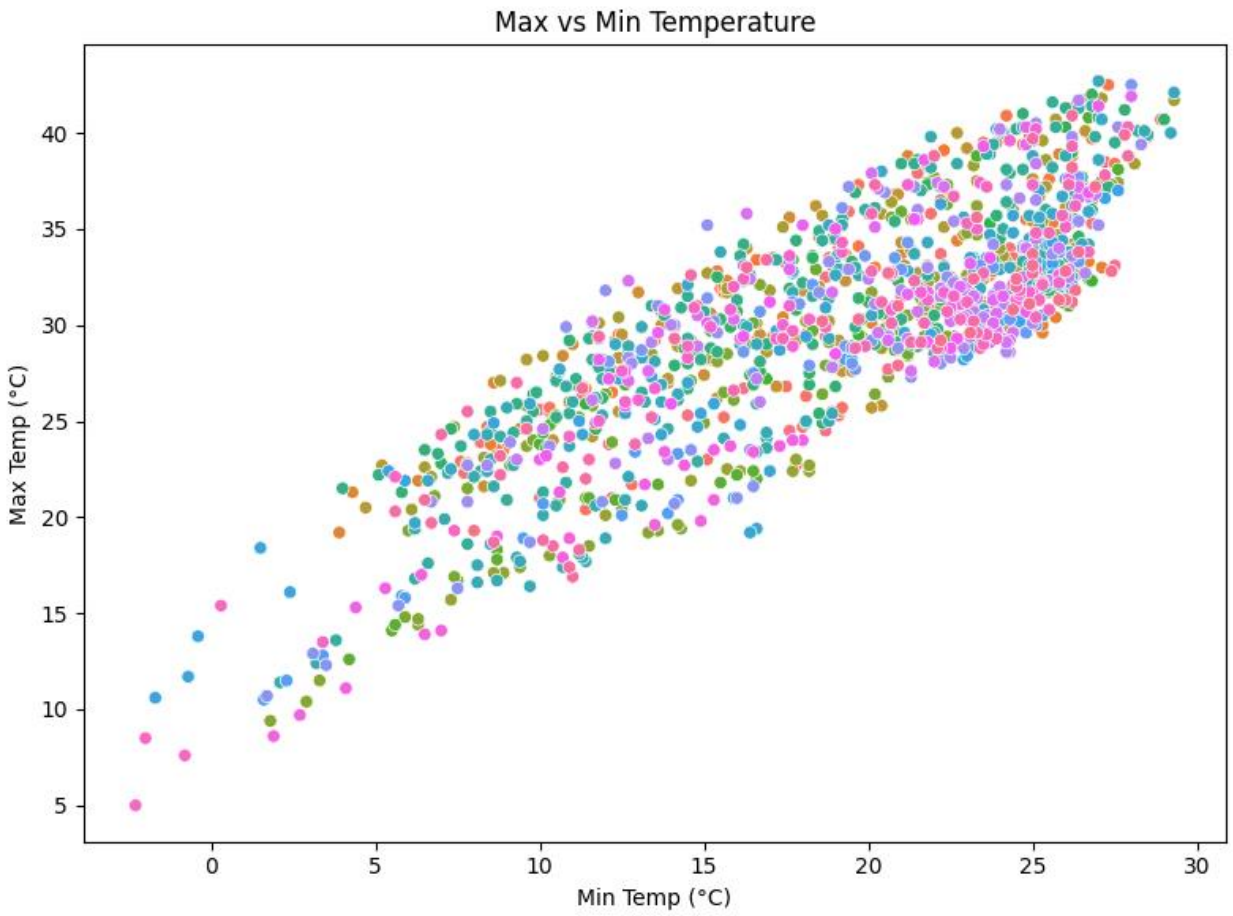


```
plt.figure(figsize=(12, 6))
sns.boxplot(data=df, x="Station_Name", y="Mean_Temp_Max")
plt.xticks(rotation=90)
plt.title("Max Temperature Distribution by Station")
plt.tight_layout()
plt.show()
```

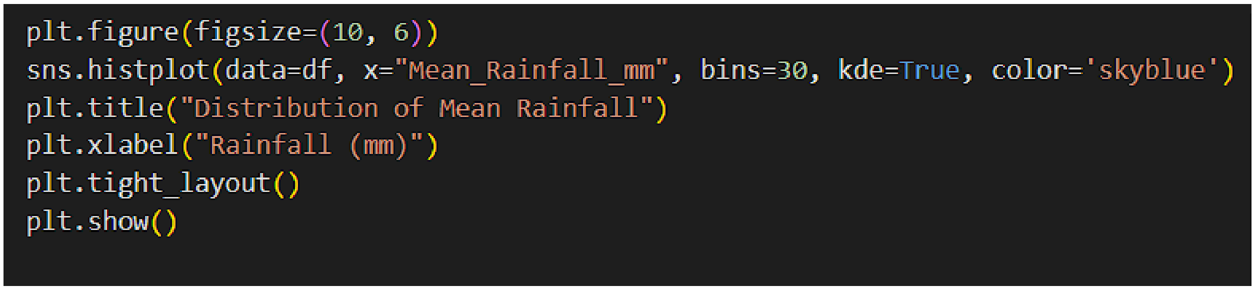


5. Scatter plot between max and min temp

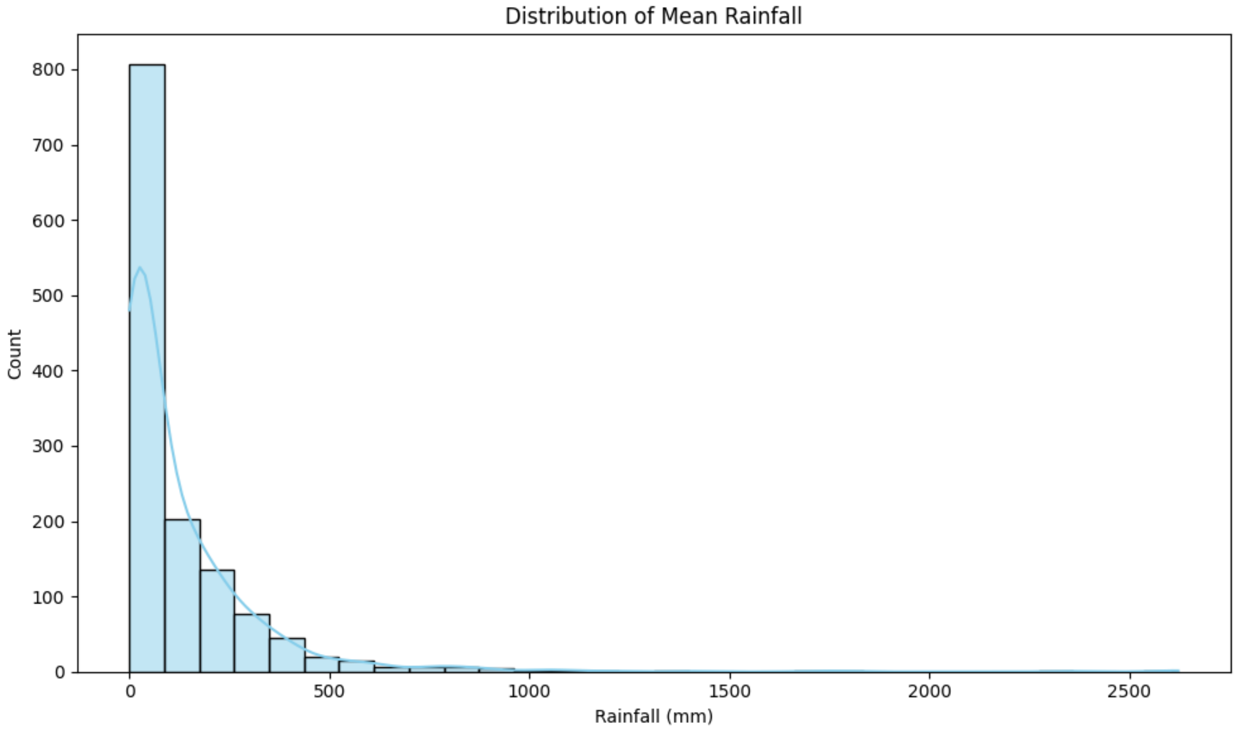

```
plt.figure(figsize=(8, 6))
sns.scatterplot(data=df, x="Mean_Temp_Min", y="Mean_Temp_Max", hue="Station_Name", legend=False)
plt.title("Max vs Min Temperature")
plt.xlabel("Min Temp (°C)")
plt.ylabel("Max Temp (°C)")
plt.tight_layout()
plt.show()
```



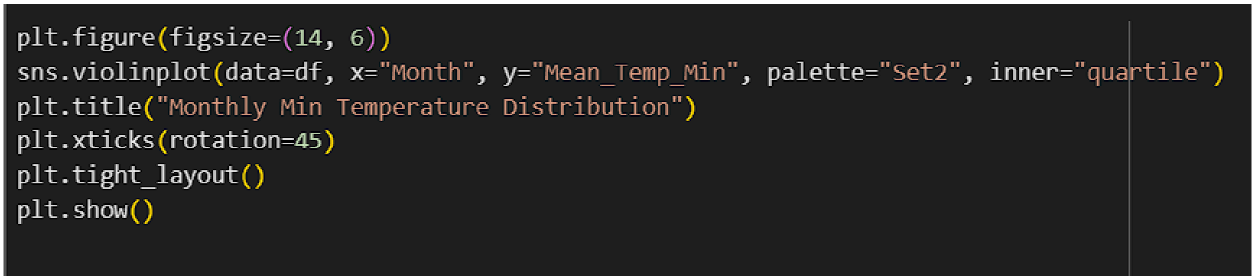
Histogram of rainfall values



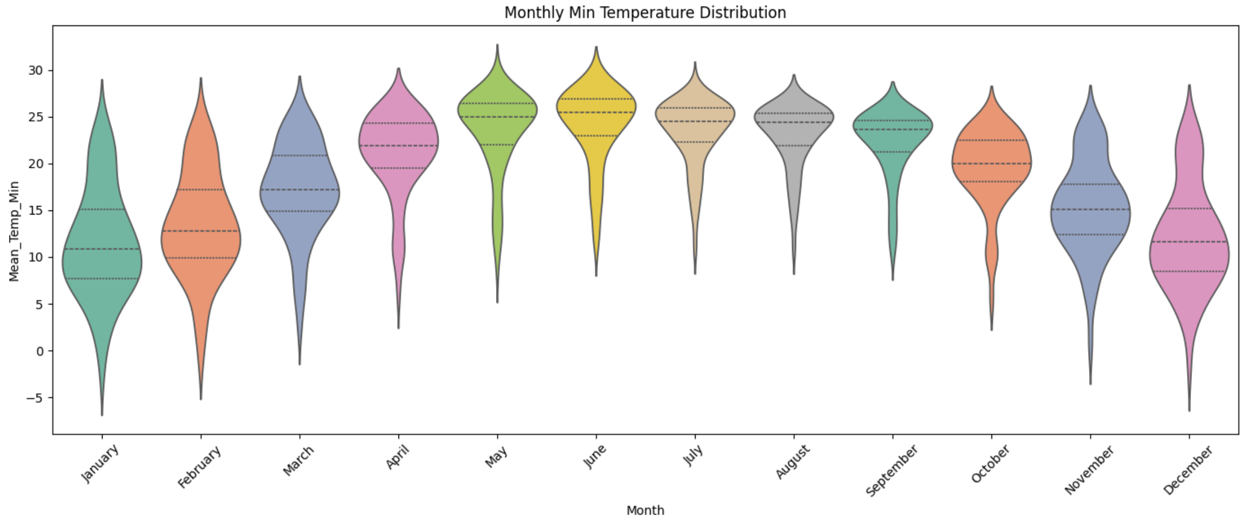
```
plt.figure(figsize=(10, 6))
sns.histplot(data=df, x="Mean_Rainfall_mm", bins=30, kde=True, color='skyblue')
plt.title("Distribution of Mean Rainfall")
plt.xlabel("Rainfall (mm)")
plt.tight_layout()
plt.show()
```



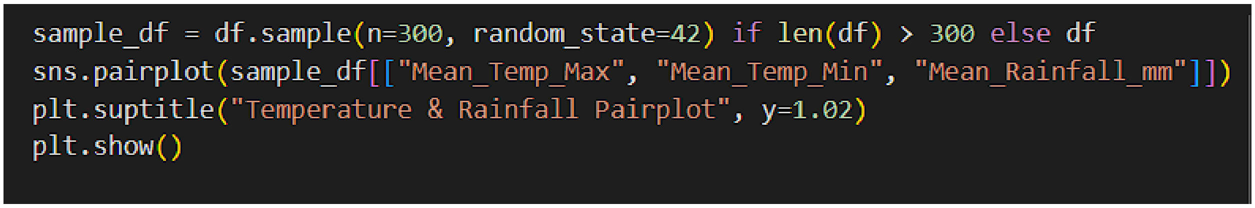
Violin plot of min temps by month



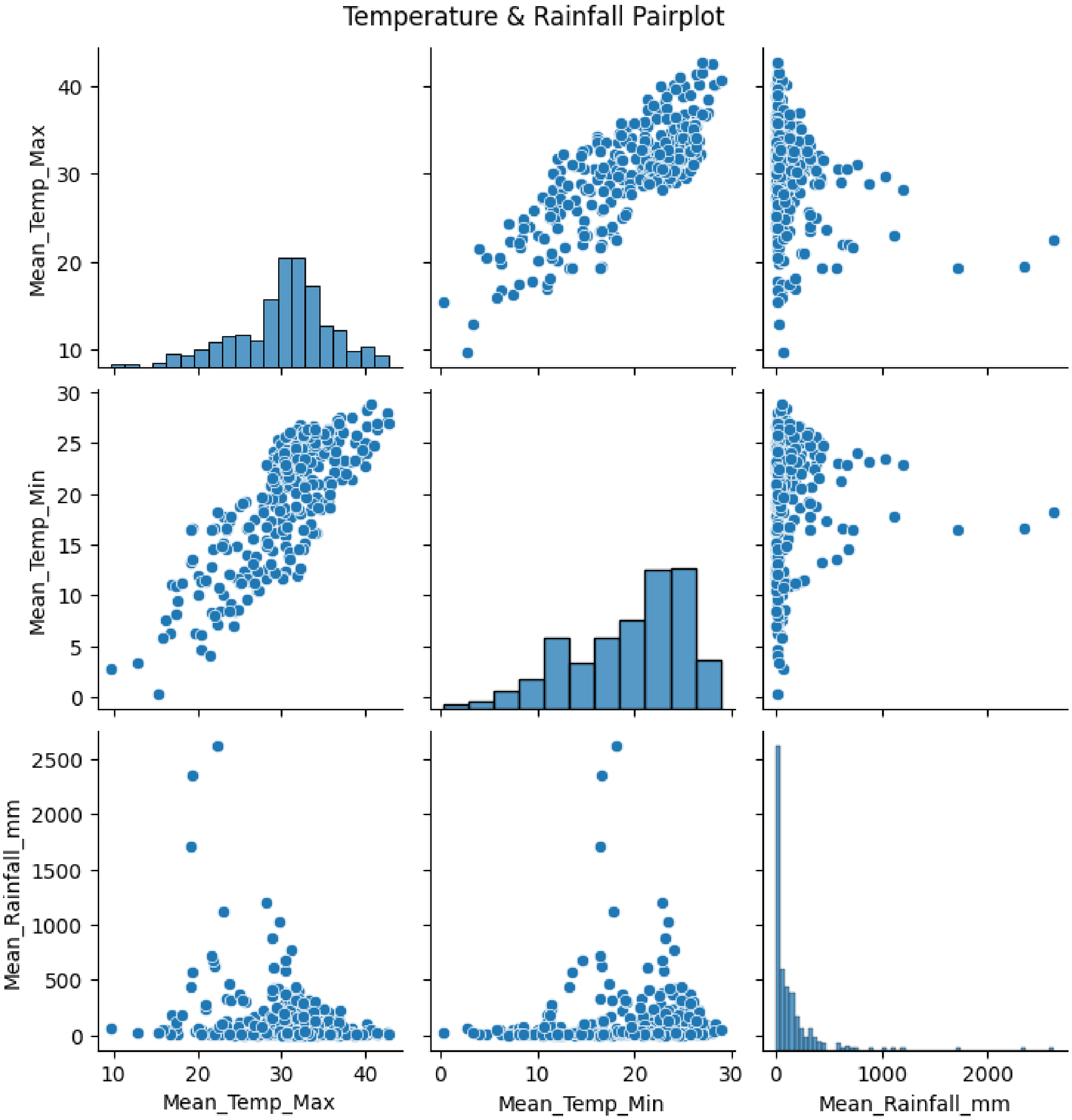
```
plt.figure(figsize=(14, 6))
sns.violinplot(data=df, x="Month", y="Mean_Temp_Min", palette="Set2", inner="quartile")
plt.title("Monthly Min Temperature Distribution")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



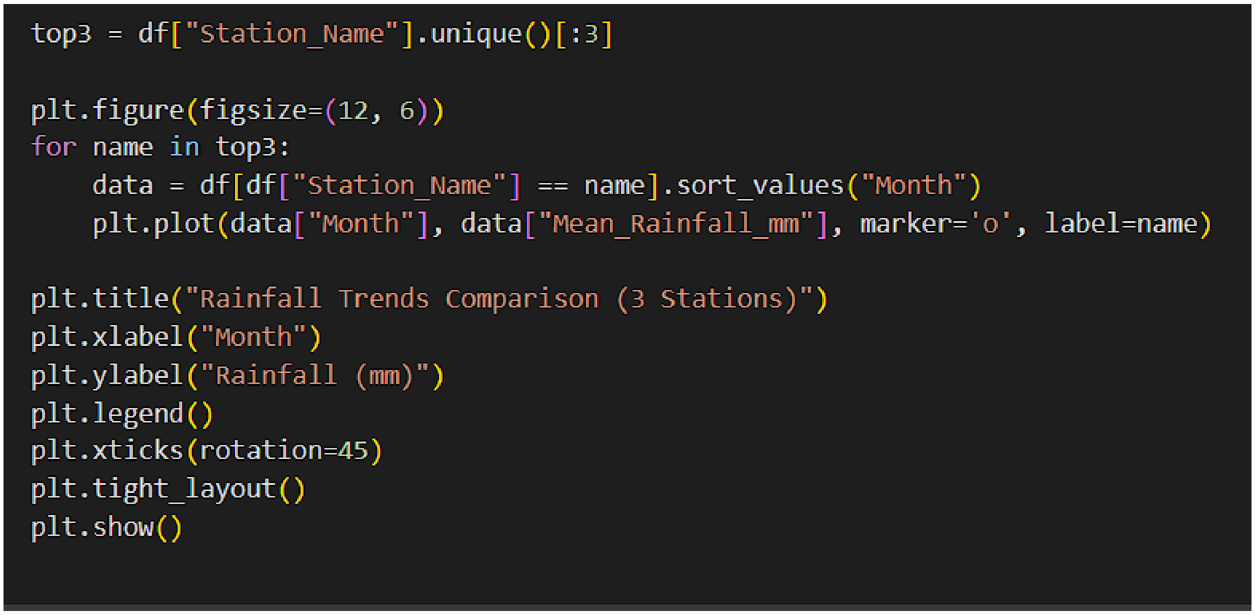
Pairplot of temperature and rainfall



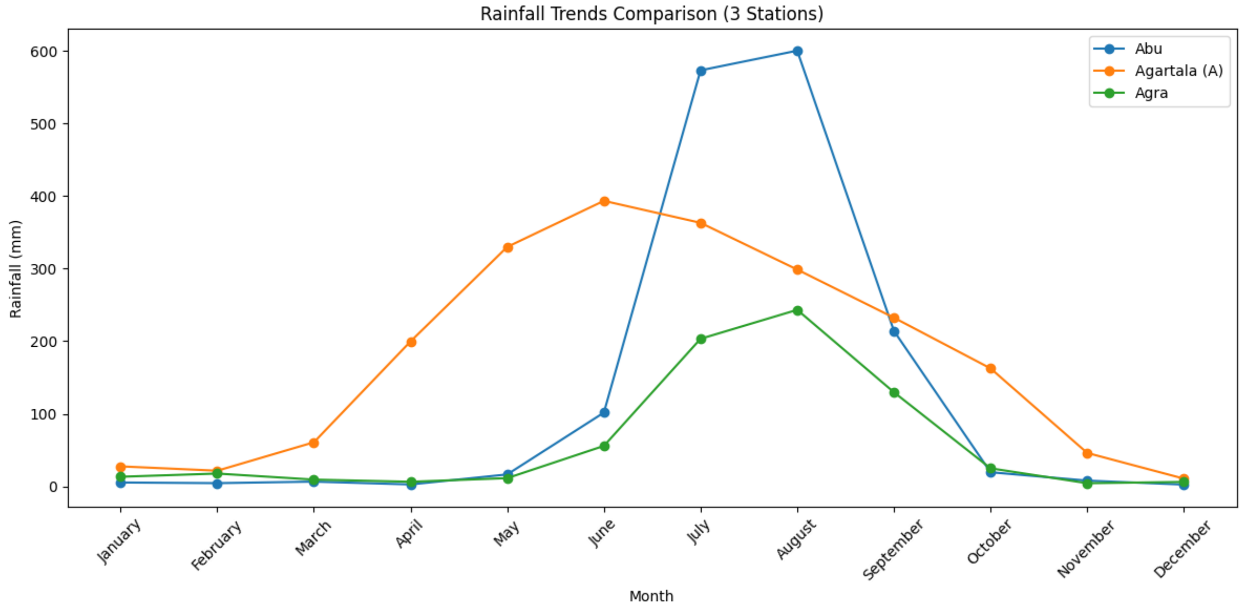
```
sample_df = df.sample(n=300, random_state=42) if len(df) > 300 else df
sns.pairplot(sample_df[["Mean_Temp_Max", "Mean_Temp_Min", "Mean_Rainfall_mm"]])
plt.suptitle("Temperature & Rainfall Pairplot", y=1.02)
plt.show()
```



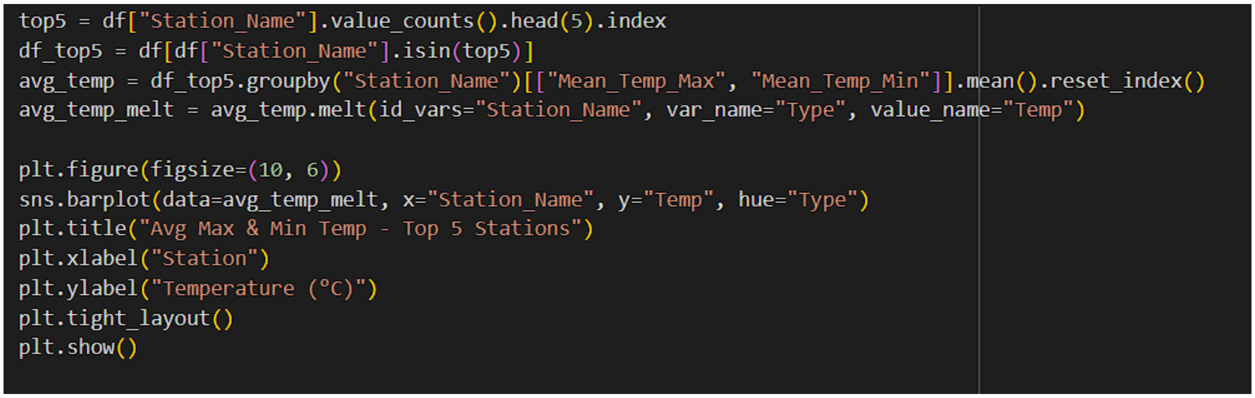
Line plot comparing rainfall across 3 stations



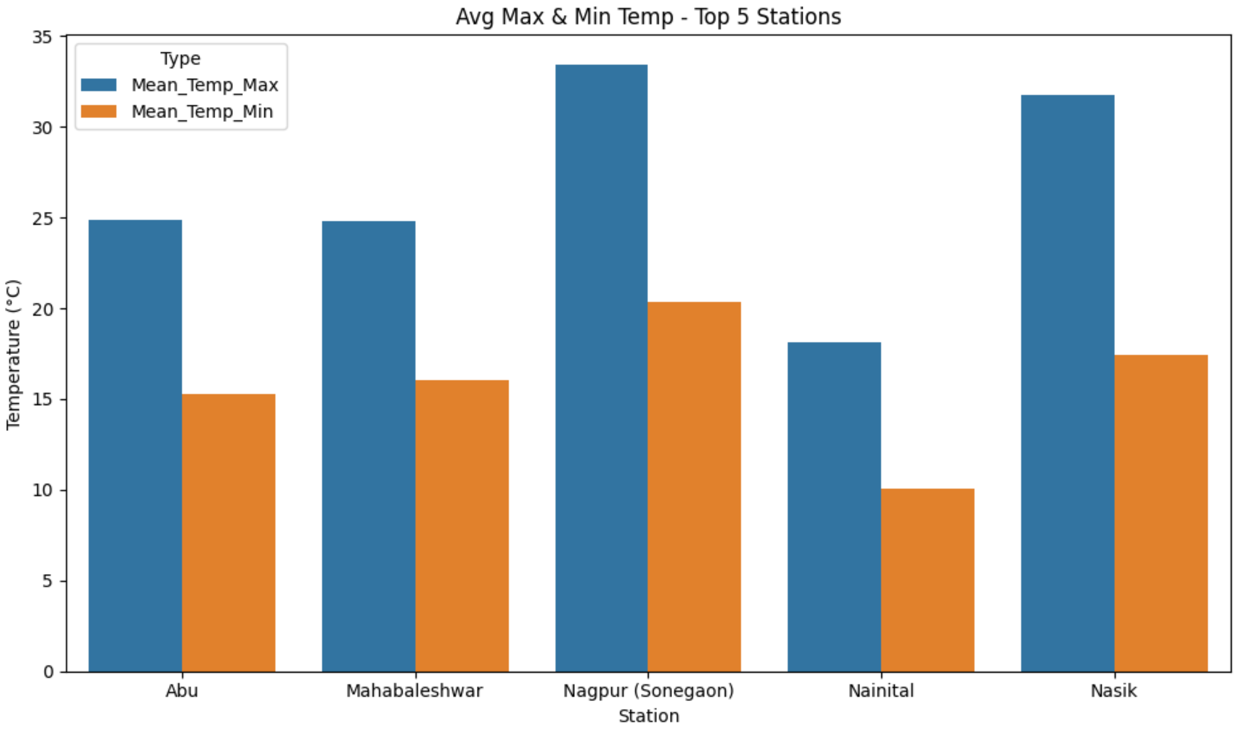
```
top3 = df["Station_Name"].unique()[:3]

plt.figure(figsize=(12, 6))
for name in top3:
    data = df[df["Station_Name"] == name].sort_values("Month")
    plt.plot(data["Month"], data["Mean_Rainfall_mm"], marker='o', label=name)

plt.title("Rainfall Trends Comparison (3 Stations)")
plt.xlabel("Month")
plt.ylabel("Rainfall (mm)")
plt.legend()
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



Grouped bar chart of avg max & min temp for top 5 stations



```
top5 = df["Station_Name"].value_counts().head(5).index
df_top5 = df[df["Station_Name"].isin(top5)]
avg_temp = df_top5.groupby("Station_Name")[["Mean_Temp_Max", "Mean_Temp_Min"]].mean().reset_index()
avg_temp_melt = avg_temp.melt(id_vars="Station_Name", var_name="Type", value_name="Temp")

plt.figure(figsize=(10, 6))
sns.barplot(data=avg_temp_melt, x="Station_Name", y="Temp", hue="Type")
plt.title("Avg Max & Min Temp - Top 5 Stations")
plt.xlabel("Station")
plt.ylabel("Temperature (°C)")
plt.tight_layout()
plt.show()
```



Conclusion

Through this analysis, we observed significant variations in temperature and rainfall patterns across different stations. Key insights include seasonal trends, extreme weather events, and regional disparities. These findings can inform policymakers and researchers working on climate resilience strategies.

Future Scope

Future work could involve incorporating additional datasets, such as greenhouse gas emissions or land-use changes, to better understand drivers of climate variability. Advanced machine learning models could also predict future climate scenarios

7. References

- **Source Dataset**: **climate_change_upto_2000_1.csv**
- **Python Documentation - Pandas Library. Retrieved from: https://pandas.pydata.org**
- **Seaborn Visualization Library, Official Documentation. Accessed from: https://seaborn.pydata.org**
- **Course Lecture Notes** – Data Science & Python, INT375,LPU.

8.**Linkedin:**

https://www.linkedin.com/posts/yogita-yadav-7199b2353_climatechange-datascience-python-activity-7316829272894033920-LnDK?utm_source=share&utm_medium=member_android&rcm=ACoAAFg_KIUBMv-7NQsj8h5tJ0zbf7E_EM550og

9.Github:

https://github.com/yogita-0204/int375