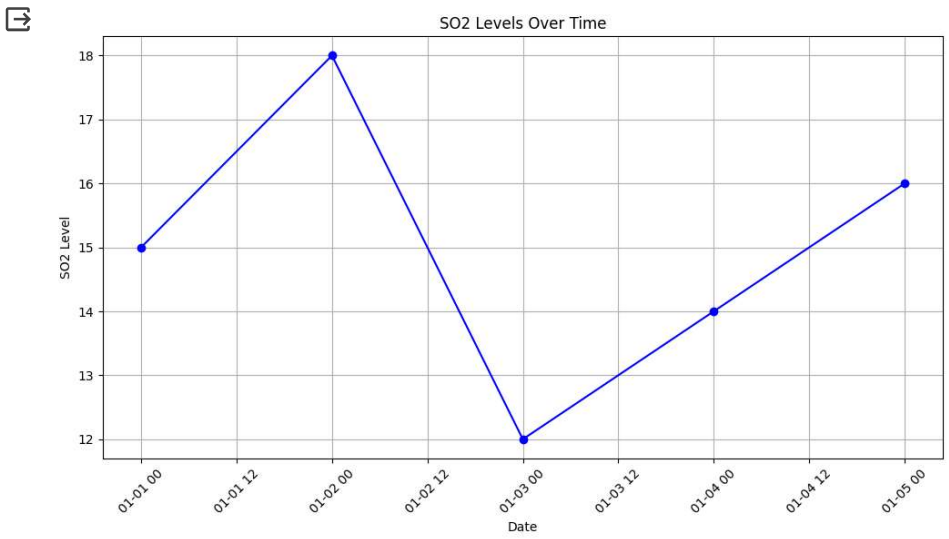
STEP 1: create visualization for SO2

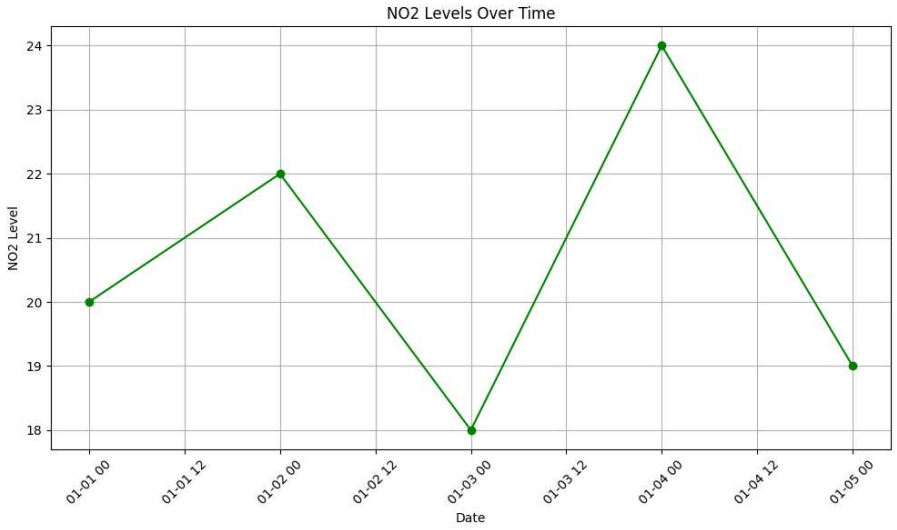
```
import matplotlib.pyplot as plt
import pandas as pd

# Sample data (replace this with your actual SO2 dataset)
data = {
    'Date': ['2023-01-01', '2023-01-02', '2023-01-03', '2023-01-04', '2023-01-05'],
    'SO2_Level': [15, 18, 12, 14, 16]  # Replace with your SO2 data
}

# Create a DataFrame from the data
df = pd.DataFrame(data)
df['Date'] = pd.to_datetime(df['Date'])  # Convert Date column to datetime format

# Plot the SO2 levels
plt.figure(figsize=(10, 6))
plt.plot(df['Date'], df['SO2_Level'], marker='o', linestyle='-', color='b')
plt.title('SO2 Levels Over Time')
plt.xlabel('Date')
plt.ylabel('SO2 Level')
plt.grid(True)

# Format the x-axis to display dates nicely
plt.xticks(rotation=45)

# Show the plot
plt.tight_layout()
plt.show()
```



STEP 2: create visualization for NO2

```
import matplotlib.pyplot as plt
import pandas as pd

# Sample data (replace this with your actual NO2 dataset)
data = {
    'Date': ['2023-01-01', '2023-01-02', '2023-01-03', '2023-01-04', '2023-01-05'],
    'NO2_Level': [20, 22, 18, 24, 19]  # Replace with your NO2 data
}
```

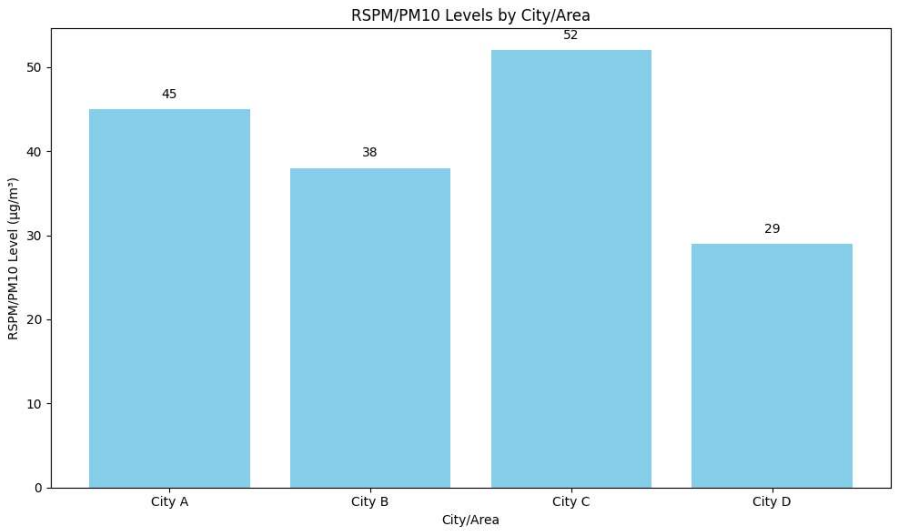
```
# Create a DataFrame from the data
df = pd.DataFrame(data)
df['Date'] = pd.to_datetime(df['Date'])  # Convert Date column to datetime format

# Plot the NO2 levels
plt.figure(figsize=(10, 6))
plt.plot(df['Date'], df['NO2_Level'], marker='o', linestyle='-', color='g')
plt.title('NO2 Levels Over Time')
plt.xlabel('Date')
plt.ylabel('NO2 Level')
plt.grid(True)

# Format the x-axis to display dates nicely
plt.xticks(rotation=45)

# Show the plot
plt.tight_layout()
plt.show()
```



STEP 3: create visualization for RSPM/PM10

```
import matplotlib.pyplot as plt

# Sample data (replace this with your actual RSPM/PM10 dataset)
categories = ['City A', 'City B', 'City C', 'City D']
rspm_pm10_levels = [45, 38, 52, 29]  # Replace with your RSPM/PM10 data

# Create a bar chart to visualize RSPM/PM10 levels
plt.figure(figsize=(10, 6))
plt.bar(categories, rspm_pm10_levels, color='skyblue')
plt.title('RSPM/PM10 Levels by City/Area')
plt.xlabel('City/Area')
plt.ylabel('RSPM/PM10 Level (µg/m³)')

# Add data labels above each bar
for i, level in enumerate(rspm_pm10_levels):
    plt.text(i, level + 1, str(level), ha='center', va='bottom')

# Show the plot
plt.tight_layout()
```

```
plt.show()
```




```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

STEP 4: grouping data

```
import pandas as pd

# Load your data into a Pandas DataFrame (replace 'data.csv' with your data file)
df = pd.read_csv('/content/drive/MyDrive/Excel 4.csv')

# Group the data by the desired column (e.g., 'City' or 'Monitoring Station')
grouped_data = df.groupby('CITIES')
```

STEP 5: calculating average

```
# Calculate the average SO2, NO2, and RSPM/PM10 levels for each group
averages = grouped_data[['SO2', 'NO2', 'RSPM/PM10']].mean()

# Display the calculated averages
print(averages)
```

```
                     SO2        NO2   RSPM/PM10
CITIES
Chennai         9.433333  23.036667  145.640000
Dindigul       22.880000  23.753333  129.780000
Erode          18.900000  27.065000  165.580000
Kanniyakumari  27.790000  34.330000  148.680000
Kodaikanal     12.130000  38.050000  171.910000
Madurai        13.897500  26.727500  156.572500
Salem          11.100000  30.060000  160.976667
Thanjavur       3.650000  30.790000  145.180000
Vellore        16.726667  23.116667  141.240000
madurai        10.220000  30.910000  114.270000
salem          30.990000  29.050000  128.150000
```

```
# Calculate the average SO2, NO2, and RSPM/PM10 levels for each group
averages = grouped_data[[ 'RSPM/PM10']].mean()

# Display the calculated averages
print(averages)
```

```
                    RSPM/PM10
CITIES
Chennai        145.640000
Dindigul       129.780000
Erode          165.580000
Kanniyakumari  148.680000
Kodaikanal     171.910000
Madurai        156.572500
Salem          160.976667
Thanjavur      145.180000
Vellore        141.240000
madurai        114.270000
salem          128.150000
```