Naan Mudhalvan Project

Air Quality Analysis in Tamil Nadu

Phase 4

TEAM MEMBERS:

Hematharshini E – 2021115041
Email - hemae2512@gmail.com
2. Sabitha S – 2021115087
Email - sabitha.suresh15@gmail.com
3. Sandhya Shankar – 2021115090
Email - sandhya.shankar.2002@gmail.com
4. Sanmitha V.S – 2021115092
Email - sanmithasadhishkumar@gmail.com
5. Akash P – 2021115314
Email - akashpanneer2004@gmail.com

Phase objective:

In this phase of our air quality analysis project, we continue to explore and visualize the air quality data. The dataset is loaded from the file "modified_transportation_data.csv," and we focus on understanding the average levels of SO_2, NO_2, and RSPM/PM10 across monitoring stations and city/town/village/area. Additionally, we create visualizations, time-series plots, and correlation matrices to gain insights into air quality trends and relationships.

Data Loading and Preparation:

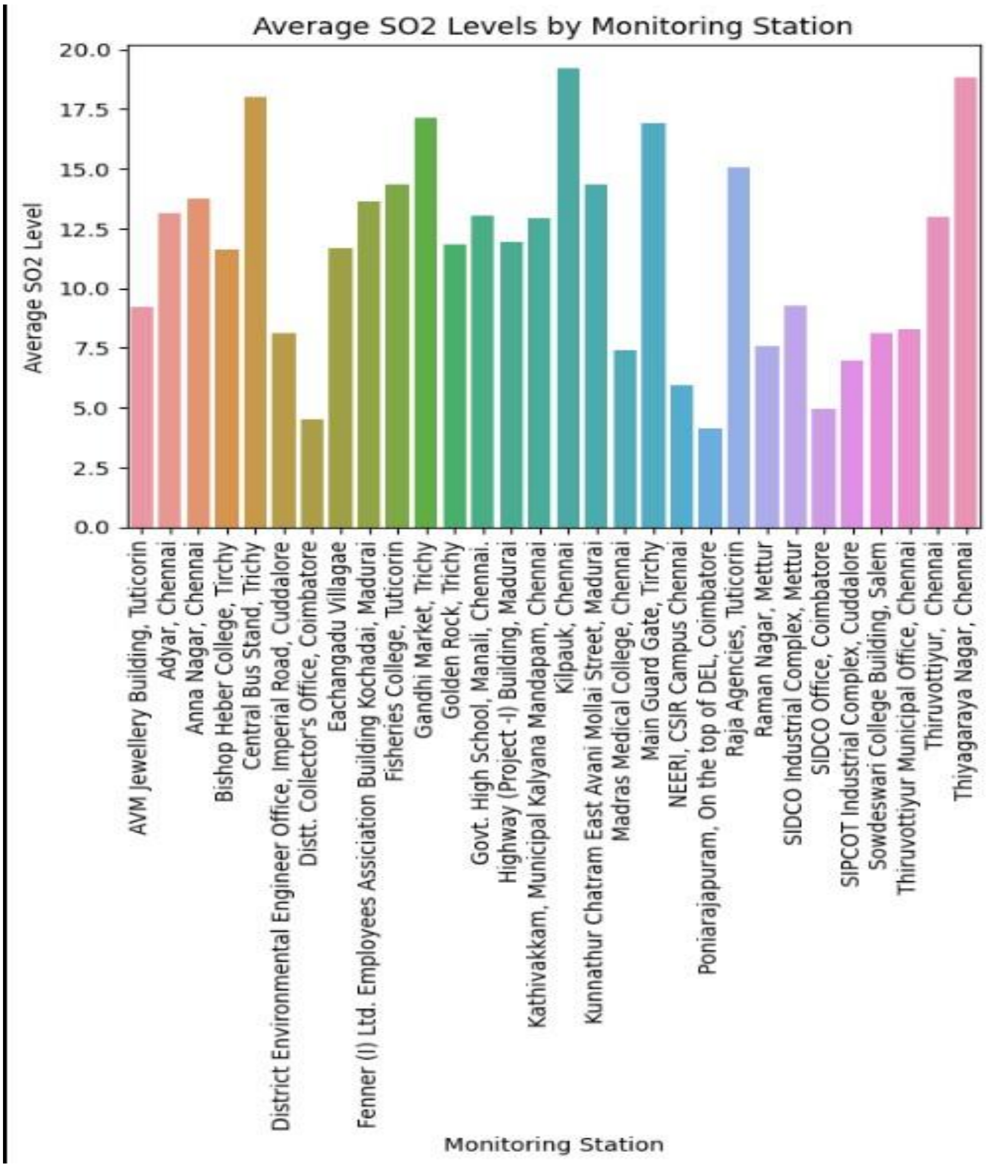
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

data = pd.read_csv("/content/modified_transportation_data.csv")
```

Average Pollution Levels by Monitoring Stations:

We calculated and visualized the average SO2, NO2, and RSPM/PM10 levels across different monitoring stations. The bar plots provide a clear overview of pollution levels by station.

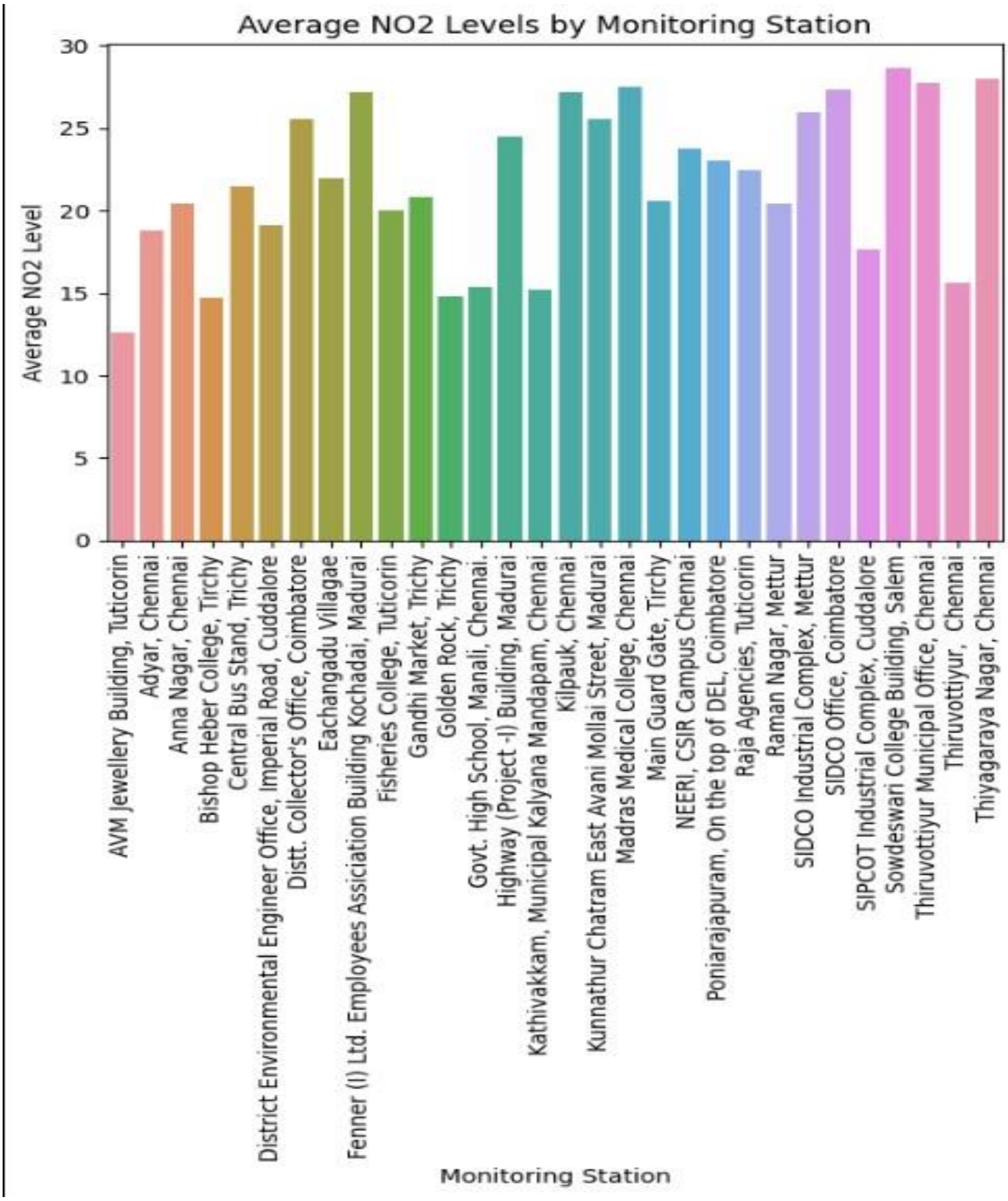
Average SO2 Levels:

```
# Create a bar plot to visualize average SO2 levels by monitoring station
sns.barplot(x=average_levels.index, y=average_levels['SO2'])
plt.xlabel('Monitoring Station')
plt.ylabel('Average SO2 Level')
plt.title('Average SO2 Levels by Monitoring Station')
plt.xticks(rotation=90)
plt.show()
```



Average NO2 Levels:

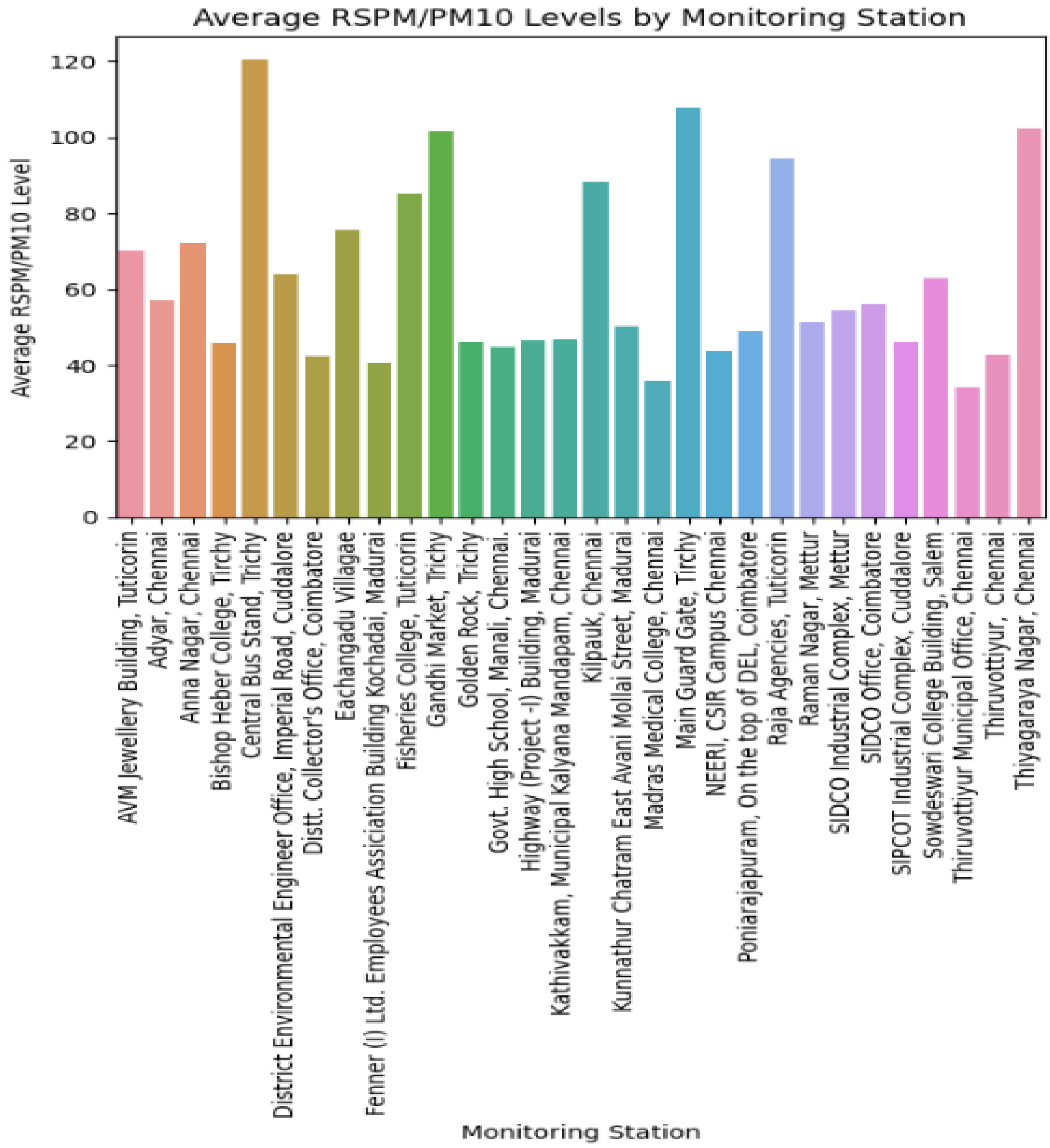
```
# Create a bar plot to visualize average NO2 levels by monitoring station
sns.barplot(x=average_levels.index, y=average_levels['NO2'])
plt.xlabel('Monitoring Station')
plt.ylabel('Average NO2 Level')
plt.title('Average NO2 Levels by Monitoring Station')
plt.xticks(rotation=90)
plt.show()
```



Average RSPM/PM10 Levels:

```
# Create a bar plot to visualize average RSPM/PM10 levels by monitoring station
sns.barplot(x=average_levels.index, y=average_levels['RSPM/PM10'])
plt.xlabel('Monitoring Station')
plt.ylabel('Average RSPM/PM10 Level')
plt.title('Average RSPM/PM10 Levels by Monitoring Station')
```

```
plt.xticks(rotation=90)
plt.show()
```

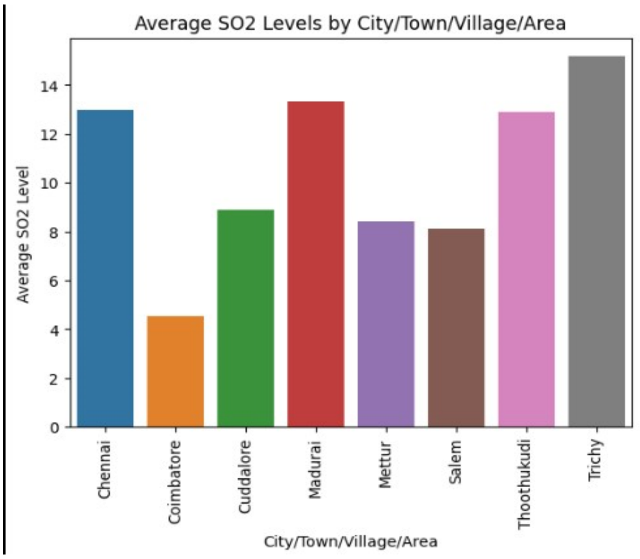


Average Pollution Levels by City/Town/Village/Area:

We calculated and visualized the average SO2, NO2, and RSPM/PM10 levels by city/town/village/area, providing insights into air quality on a larger scale.

Average SO2 Levels:

```
# Create a bar plot to visualize average SO2 levels by city/town/village/area
sns.barplot(x=average_city_levels.index, y=average_city_levels['SO2'])
plt.xlabel('City/Town/Village/Area')
plt.ylabel('Average SO2 Level')
plt.title('Average SO2 Levels by City/Town/Village/Area')
plt.xticks(rotation=90)
plt.show()
```

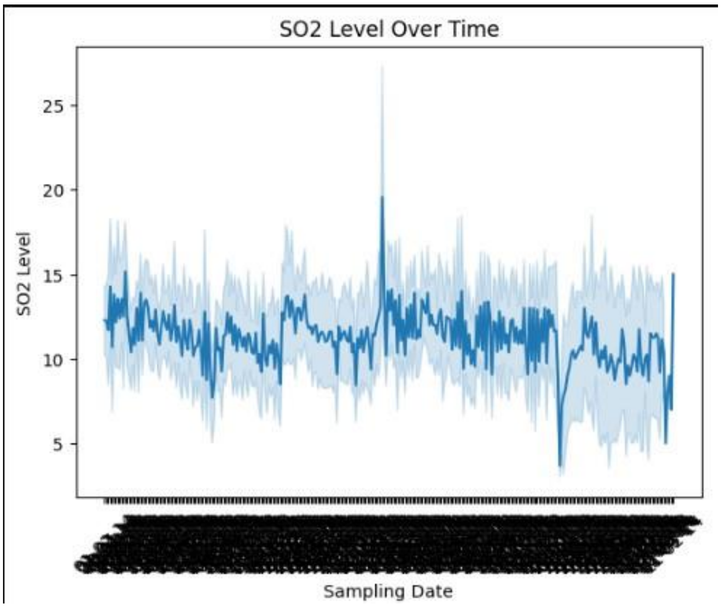


Time-Series Plots:

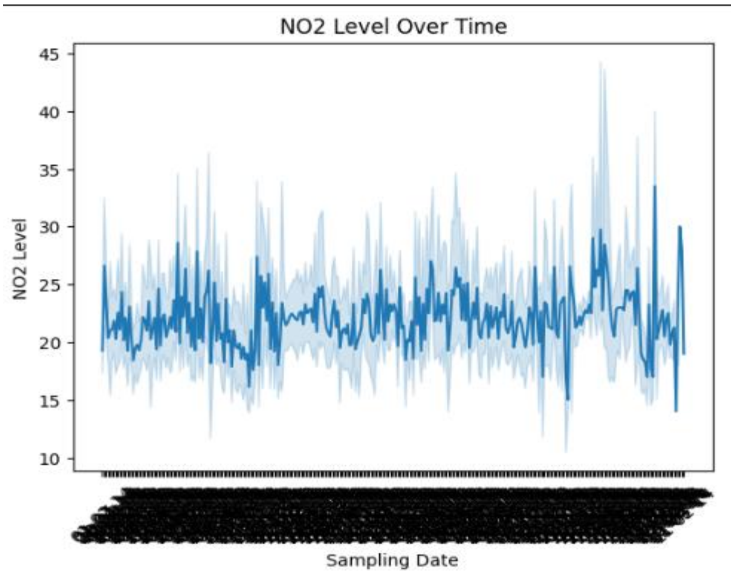
We created time-series plots to visualize the changes in pollutants levels over time.

SO2:

```
# Time-series plot for SO2 levels
sns.lineplot(x="Sampling Date", y="SO2", data=data)
plt.xlabel('Sampling Date')
plt.ylabel('SO2 Level')
plt.title('SO2 Level Over Time')
plt.xticks(rotation=45)
plt.show()
```



NO2:



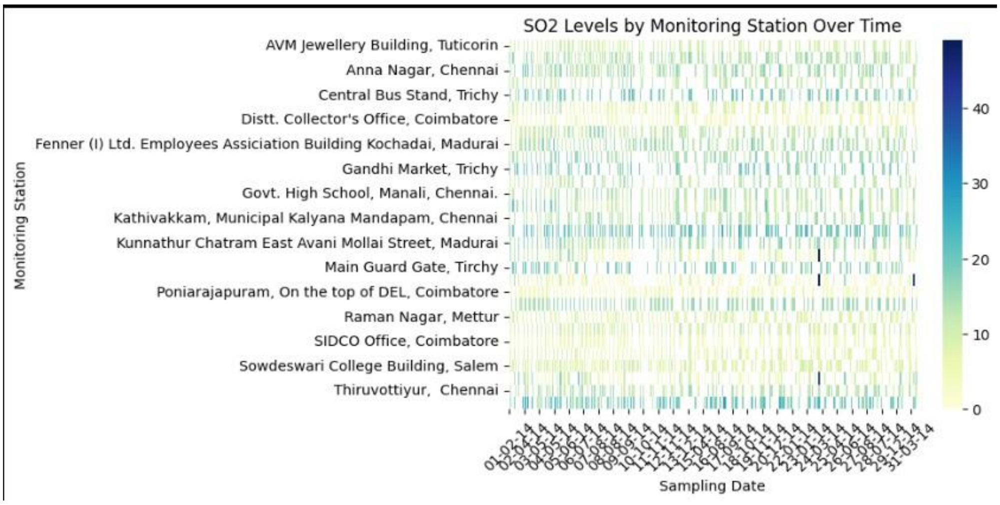
Heatmaps:

We generated heatmaps to observe variations in pollutants levels by monitoring station over time.

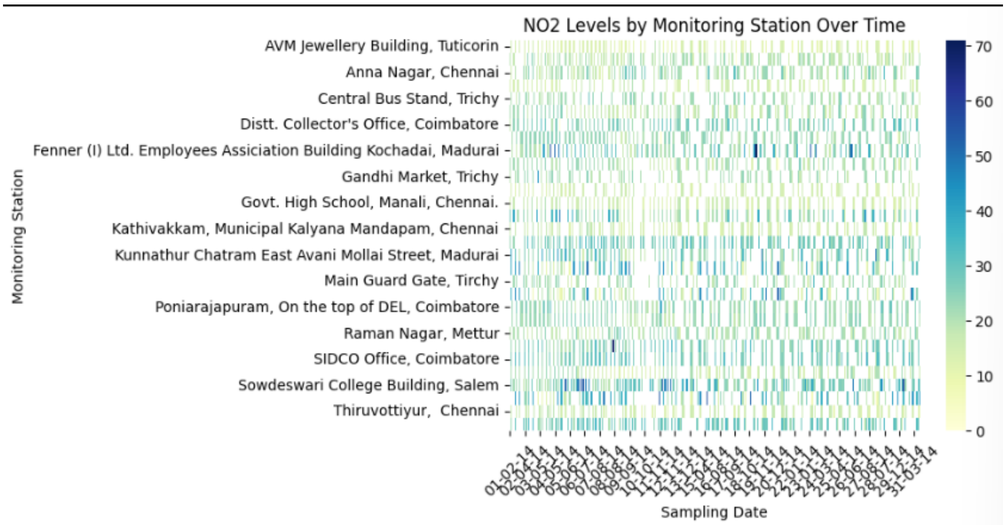
SO2:

```
# Heatmap for SO2 levels by monitoring station
sns.heatmap(data.pivot_table(values='SO2', index='Location of Monitoring Station', columns='Sampling Date'), cmap='YlGnBu')
```

```python
plt.xlabel('Sampling Date')
plt.ylabel('Monitoring Station')
plt.title('SO2 Levels by Monitoring Station Over Time')
plt.xticks(rotation=45)
plt.show()
```



NO2:



Areas with Highest Pollution Levels:

We sorted and identified areas with the highest average SO2, NO2, and RSPM/PM10 levels.

SO2:

```python
# Sort by average SO2 levels
sorted_city_so2 = average_city_levels.sort_values(by='SO2', ascending=False)
```

```
print("Areas with highest average SO2 levels:")
print(sorted_city_so2.head(10))
```

```
Areas with highest average SO2 levels:
                            SO2        NO2  RSPM/PM10
City/Town/Village/Area
Trichy                15.168937  18.542234  85.054496
Madurai               13.319728  25.768707  45.724490
Chennai               12.975000  21.978000  58.998000
Thoothukudi           12.901024  18.385666  83.458904
Cuddalore              8.905405  19.577703  61.881757
Mettur                 8.429268  23.185366  52.721951
Salem                  8.114504  28.664122  62.954198
Coimbatore             4.525597  25.238908  49.217241
```

NO2:

```
# Sort by average NO2 levels
sorted_city_no2 = average_city_levels.sort_values(by='NO2', ascending=False)
print("Areas with highest average NO2 levels:")
print(sorted_city_no2.head(10))
```

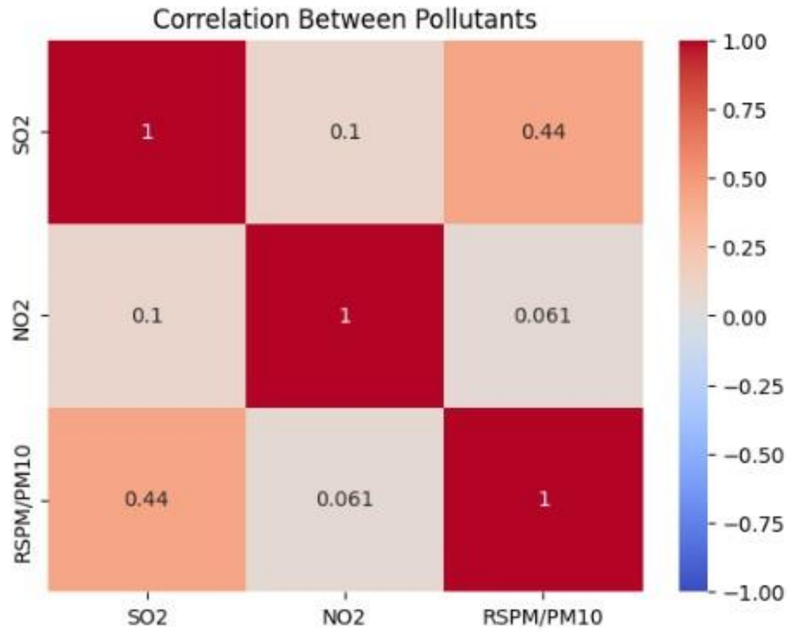
```
Areas with highest average NO2 levels:
                            SO2        NO2  RSPM/PM10
City/Town/Village/Area
Salem                  8.114504  28.664122  62.954198
Madurai               13.319728  25.768707  45.724490
Coimbatore             4.525597  25.238908  49.217241
Mettur                 8.429268  23.185366  52.721951
Chennai               12.975000  21.978000  58.998000
Cuddalore              8.905405  19.577703  61.881757
Trichy                15.168937  18.542234  85.054496
Thoothukudi           12.901024  18.385666  83.458904
```

Correlation Analysis:

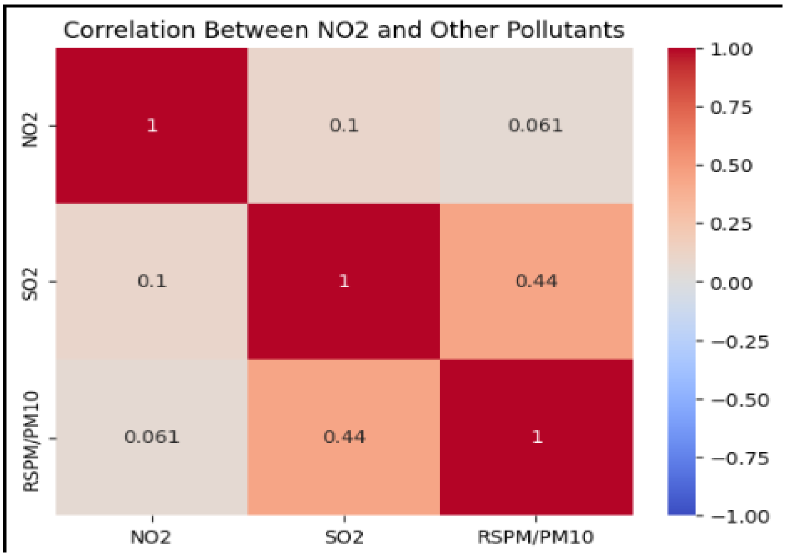
We calculated the correlation between pollutants and visualized the results using a heatmap.

SO2:

```
# Create similar analyses for NO2 and RSPM/PM10.
correlation_matrix = data[['SO2', 'NO2', 'RSPM/PM10']].corr()
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', vmin=-1, vmax=1)
plt.title('Correlation Between Pollutants')
plt.show()
```



NO2:



Conclusion:

In this phase of the air quality analysis project, we have conducted extensive data analysis, visualization, and correlation studies. These insights into pollution levels across monitoring stations, city/town/village/areas, and their correlations provide a solid foundation for understanding air quality trends and patterns. Future work may involve predictive modeling and more advanced analytics to address air quality challenges comprehensively.