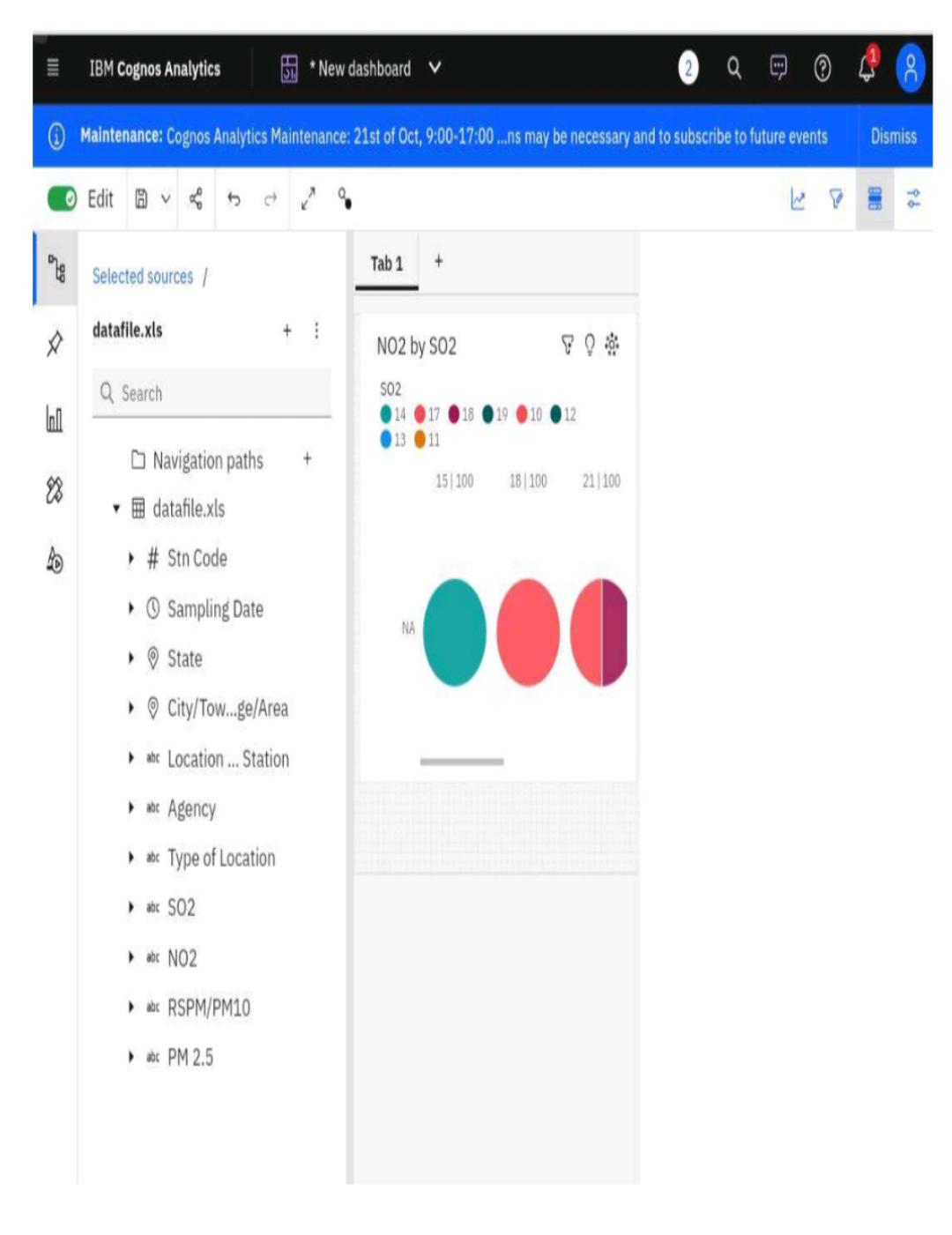
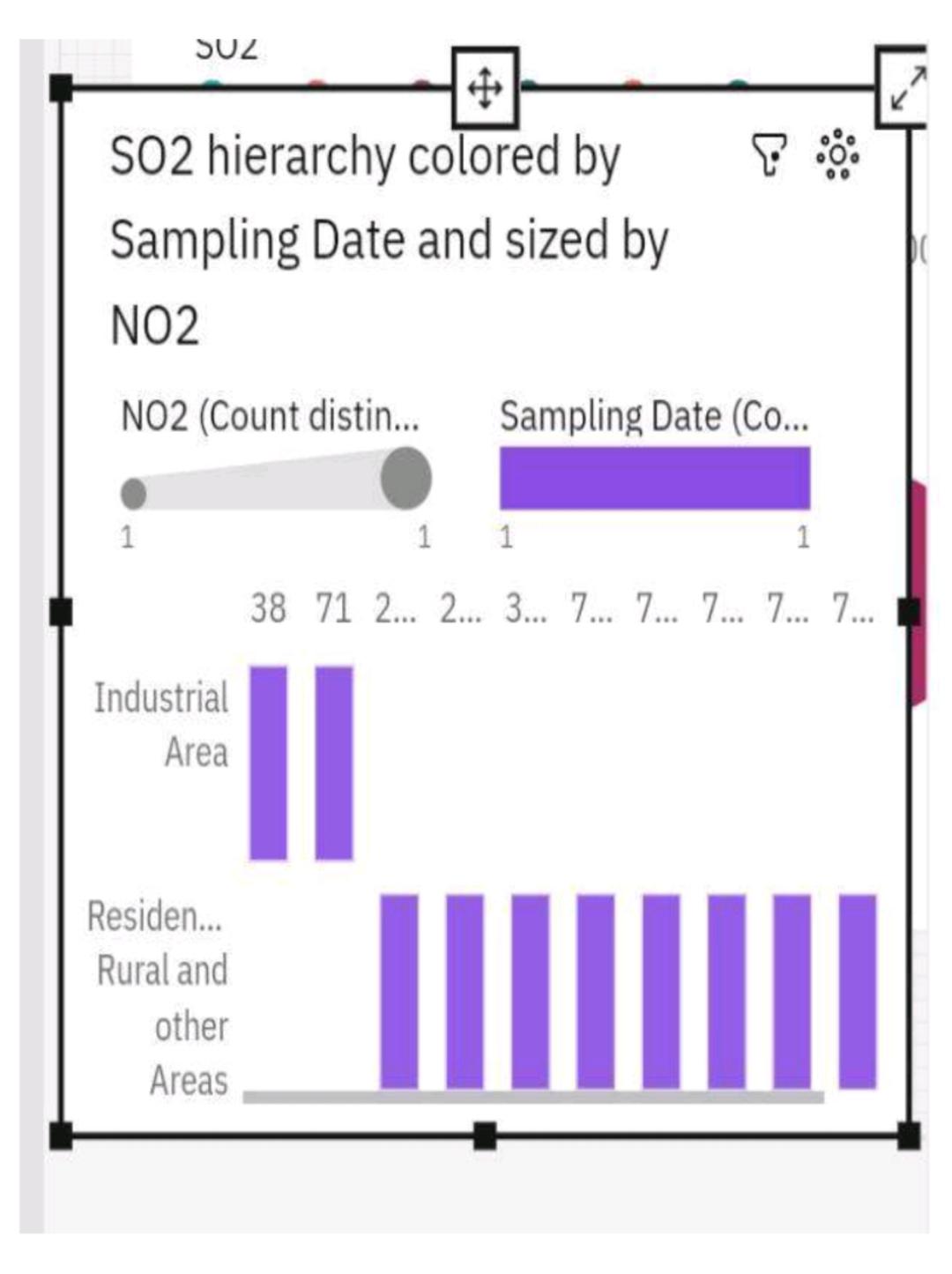
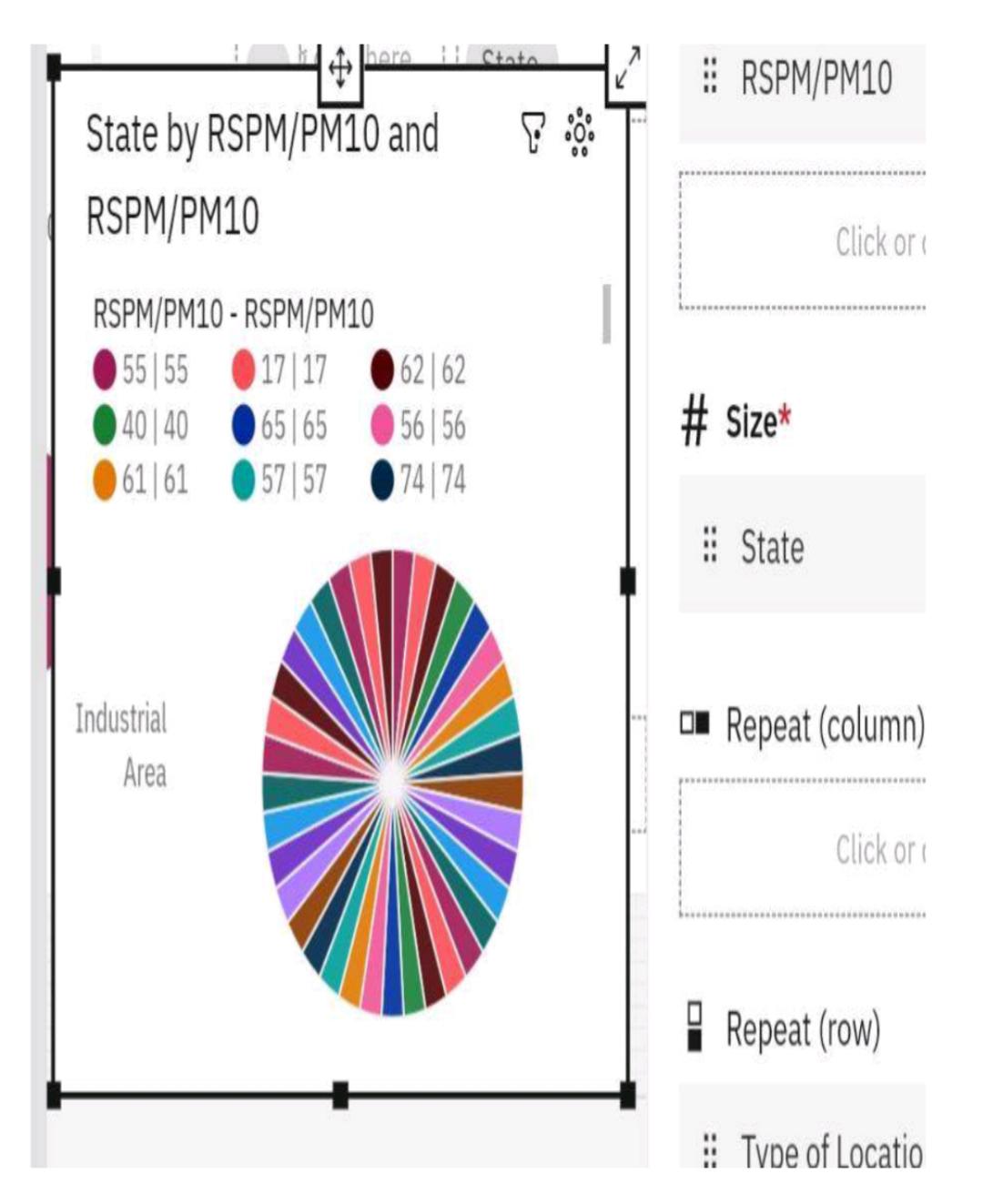
**Air quality analysis in tamilnadu**

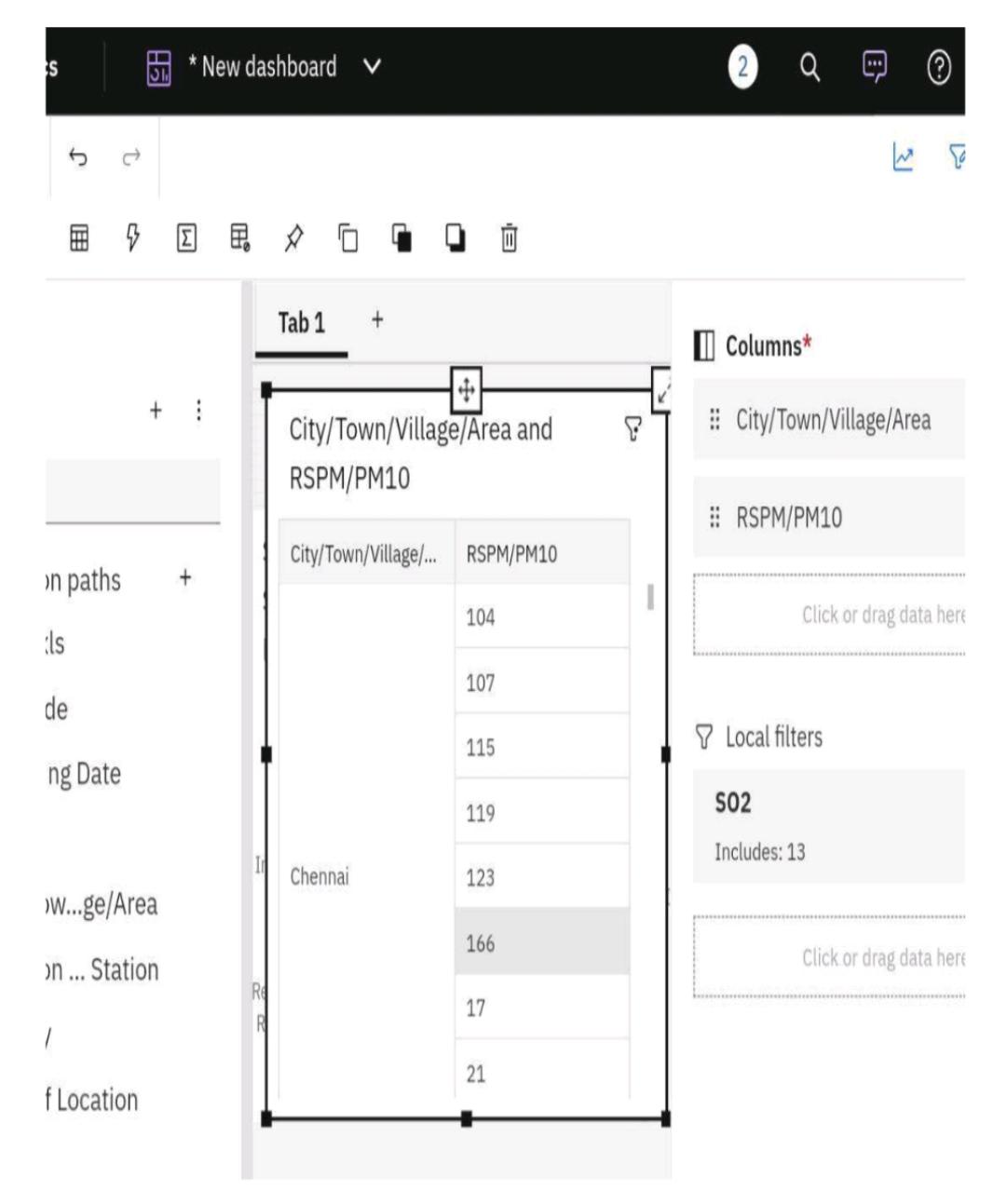
To perform the air quality analysis and to create visualizations

The MATLAB code helps preprocess the sensor data, provides functions to classify the data, and provides functions for visualizing the processed air quality data.

Calculated average SO2,NO2 and RSPM/PM10 levels across different monitoring stations,cities and areas are as follows..







Creating visualizations using visualization libraries

(example matplotlib and seaborn)

To perform air quality analysis and create visualizations using data visualization libraries such as Matplotlib and Seaborn, you will need to follow these steps:

1. Import the necessary libraries:

Python

Import pandas as pd

Import matplotlib.pyplot as plt

Import seaborn as sns

1. Load the air quality data into a Pandas DataFrame:

Python

Data = pd.read\_csv(‘air\_quality\_data.csv’)

1. Explore the data to understand its structure and available columns:

Python

Print(data.head()) # Display the first few rows of the DataFrame

Print(data.info()) # Get information about the DataFrame

1. Clean and preprocess the data if necessary (e.g., handle missing values, convert data types).
2. Perform analysis on the air quality data using various statistical methods and calculations.
3. Create visualizations using Matplotlib and Seaborn to gain insights from the data. Here are a few examples:
   * Line Plot: Visualize the trend of a specific air quality parameter over time.

Python

Plt.plot(data[‘Date’], data[‘PM2.5’])

Plt.xlabel(‘Date’)

Plt.ylabel(‘PM2.5’)

Plt.title(‘PM2.5 Trend Over Time’)

Plt.show()

* + Histogram: Display the distribution of a particular air quality parameter.

Python

Sns.histplot(data[‘O3’], bins=20)

Plt.xlabel(‘Ozone Level’)

Plt.ylabel(‘Frequency’)

Plt.title(‘Distribution of Ozone Levels’)

Plt.show()

* + Box Plot: Compare the air quality parameters across different categories.

Python

Sns.boxplot(x=’Location’, y=’PM10’, data=data)

Plt.xlabel(‘Location’)

Plt.ylabel(‘PM10’)

Plt.title(‘Comparison of PM10 Levels across Locations’)

Plt.show()

* + Heatmap: Visualize the correlation between different air quality parameters.

Python

Corr\_matrix = data[[‘PM2.5’, ‘PM10’, ‘O3’, ‘CO’, ‘SO2’, ‘NO2’]].corr()

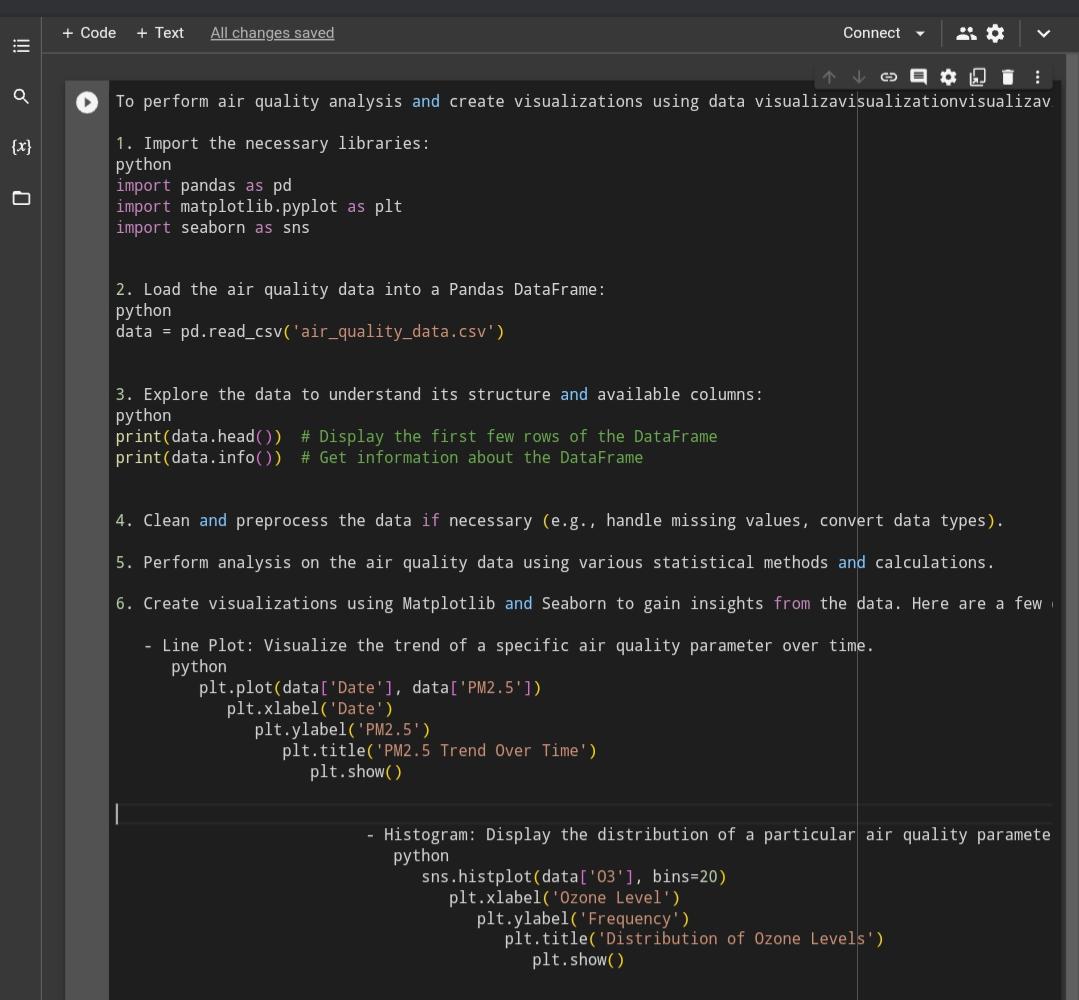
Sns.heatmap(corr\_matrix, annot=True, cmap=’coolwarm’)

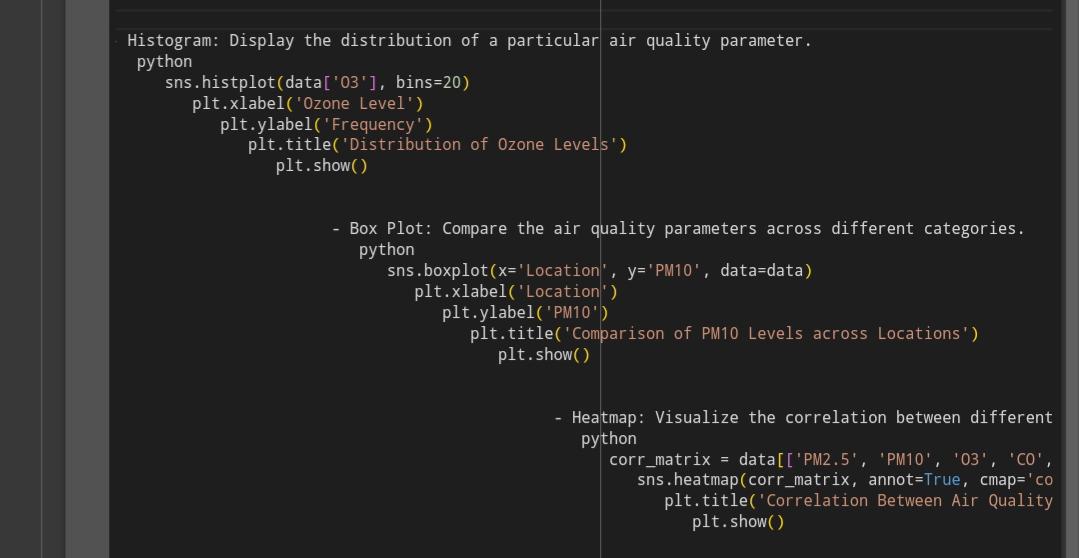
Plt.title(‘Correlation Between Air Quality Parameters’)

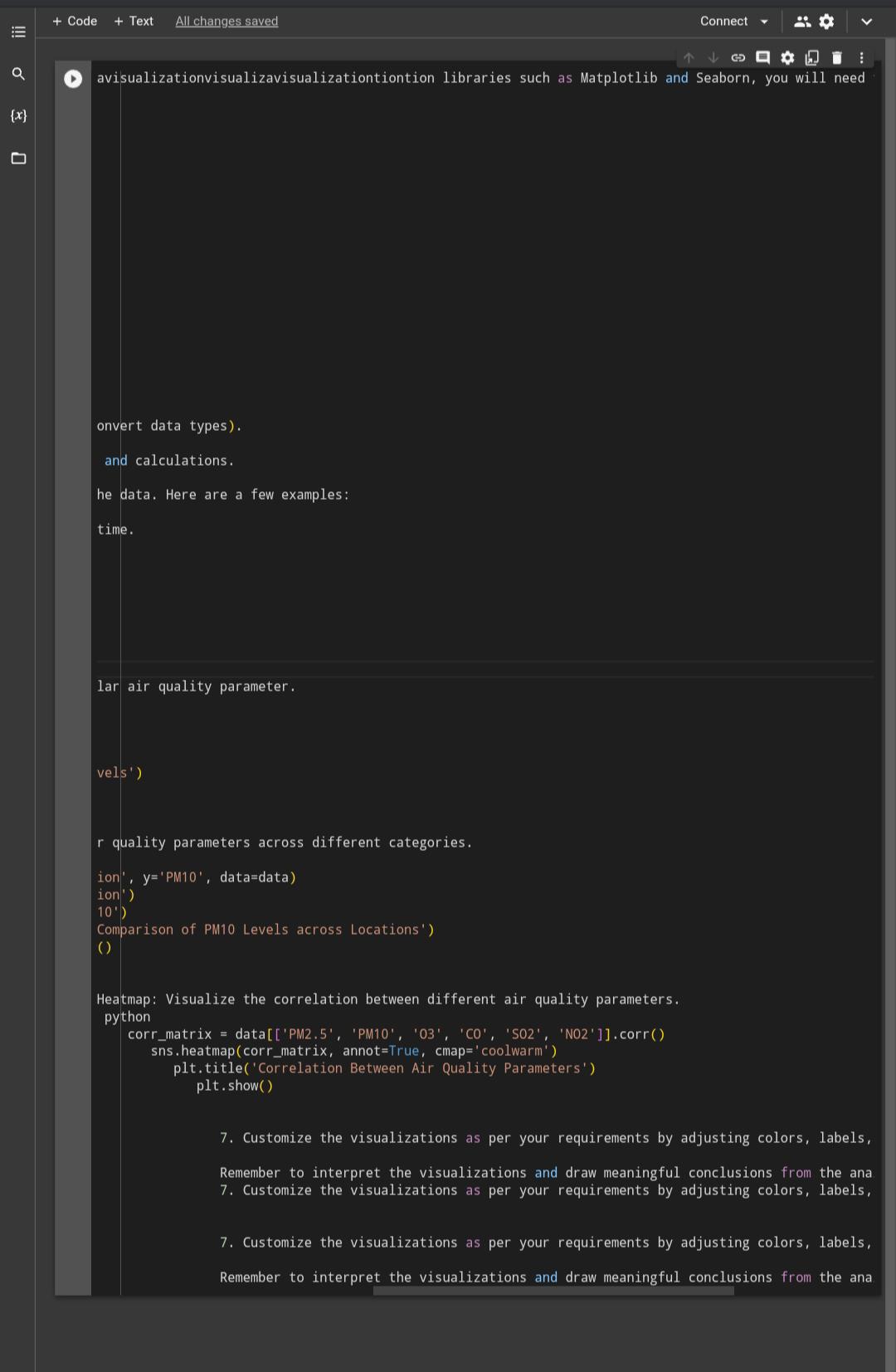
Plt.show()

1. Customize the visualizations as per your requirements by adjusting colors, labels, titles, and other parameters.

Importing this code in Jupiter platform

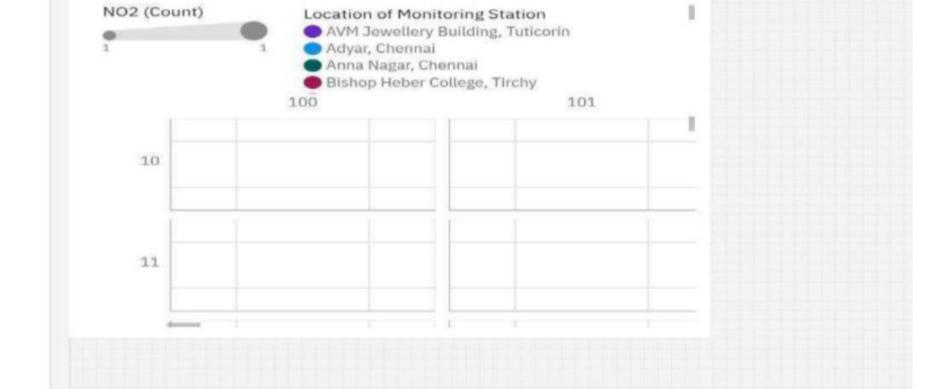






**Referred dataset link**

https://tn.data.gov.in/resource/location-wise-daily-ambient-air-quality-tamil-nadu-year-2014

Pollution trends and area with high pollution level found by using monitoring system with sampling date.

**Conclusion**

By this analysis we have concluded the high pollution level area through given sample dataset and it is implemented in ibm and Jupiter platform for better understanding and it greatly helps us to develop our project innovatively.