**AIR QUALITY ANALYSIS**

**TEAM MEMBER**

**311521205025:JITHENDHARAN S**

**Phase 4 Submission Document**

**Project:**Air Quality Analysis using data analytics

**PROGRAM:**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

df = pd.read\_csv("air\_quality.csv")

average\_pollution = df.groupby('Location of Monitoring Station')[['NO2', 'RSPM/PM10']].mean()

plt.figure(figsize=(12, 6))

plt.bar(average\_pollution.index, average\_pollution['NO2'])

plt.xlabel('Area')

plt.ylabel('Average NO2 Level')

plt.title('Average NO2 Levels Across Areas')

plt.xticks(rotation=90)

plt.show()

plt.figure(figsize=(12, 6))

plt.bar(average\_pollution.index, average\_pollution['RSPM/PM10'])

plt.xlabel('Area')

plt.ylabel('Average RSPM/PM10 Level')

plt.title('Average RSPM/PM10 Levels Across Areas')

plt.xticks(rotation=90)

plt.show()

high\_no2\_threshold = 30

high\_no2\_areas = df[df['NO2'] > high\_no2\_threshold]

plt.figure(figsize=(12, 6))

sns.countplot(data=high\_no2\_areas, x='Location of Monitoring Station', order=high\_no2\_areas['Location of Monitoring Station'].value\_counts().index)

plt.xlabel('Area')

plt.ylabel('Count of High NO2 Readings')

plt.title(f'Areas with NO2 Levels Above {high\_no2\_threshold}')

plt.xticks(rotation=90)

plt.show()

plt.figure(figsize=(10, 2))

sns.scatterplot(data=df, x='NO2', y='RSPM/PM10', hue='Location of Monitoring Station')

plt.xlabel('NO2 Level')

plt.ylabel('RSPM/PM10 Level')

plt.title('Scatter Plot of NO2 vs. RSPM/PM10')

plt.show()

correlation\_matrix = df[['NO2', 'RSPM/PM10']].corr()

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

sns.heatmap(correlation\_matrix, annot=True, cmap='coolwarm', linewidths=0.5)

plt.title('Correlation Heatmap: NO2 vs. RSPM/PM10')

plt.show()

average\_pollution = df.groupby('Stn Code')[['SO2', 'NO2', 'RSPM/PM10']].mean()

import matplotlib.pyplot as plt

plt.figure(figsize=(12, 6))

plt.bar(average\_pollution.index, average\_pollution['SO2'])

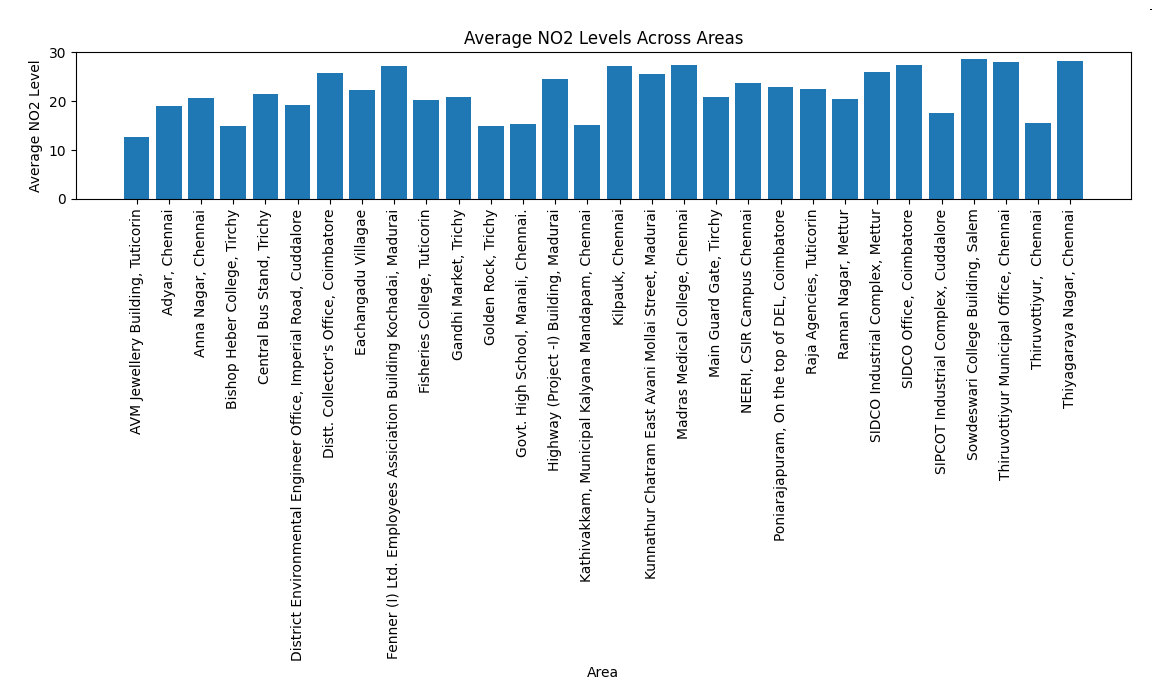
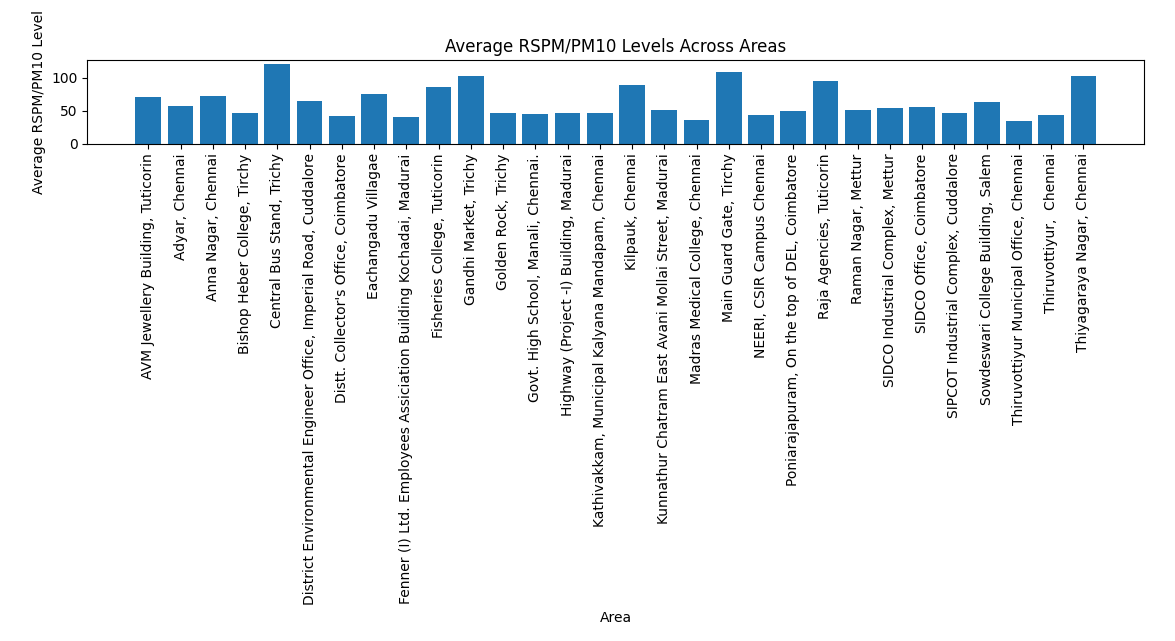
plt.xlabel('Monitoring Station')

plt.ylabel('Average SO2 Level')

plt.title('Average SO2 Levels Across Monitoring Stations')

plt.xticks(rotation=90)

plt.show()

**OUTPUT:**

