Project Title: Air Q Assessment TN

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

Phase 1: Project Definition and Design Thinking

Project Definition:

```
The project aims to analyze and visualize air quality data from
monitoring stations in Tamil Nadu. The objective is to gain
insights into air pollution trends, identify areas with high
pollution levels, and develop a predictive model to estimate
RSPM/PM10 levels based on SO2 and NO2 levels. This project involves
defining objectives, designing the analysis approach, selecting
visualization techniques, and creating a predictive model using
Python and relevant libraries.
```

Design Thinking:

```
    Project Objectives: Define objectives such as analyzing air
quality trends, identifying pollution hotspots, and building a
predictive model for RSPM/PM10 levels.
    Analysis Approach: Plan the steps to load, preprocess, analyze,
and visualize the air quality data.
    Visualization Selection: Determine visualization techniques
(e.g., line charts, heatmaps) to effectively represent air quality
trends and pollution levels.
```

Phase 2: Innovation

Phase 3: Development Part 1

In this phase we'll begin the analysis by loading and preprocessing the air quality dataset with the help of python module named as 'pandas'.

Install pandas module

```
pip install pandas
```

Now, import the module

```
pip import pandad as pd
```

> pip import pandas as pd

Load dataset

> rawdata_df = pd.read_csv(r"C:\Users\vicky\Desktop\Project\dataset.csv") 

Output:  View Columns Data of row 1

In [5]:
```
import pandas as pd
```

In [10]:
```
rawdata_df = pd.read_csv(r"C:\Users\vicky\Desktop\Project\dataset.csv")
```

Data Set

In [11]:
```
rawdata_df
```

Out[11]:

	Stn Code	Sampling Date	State	City/Town /Village/Area	Location of Monitoring Station	Agency	Type of Location	SO2	NO2	RSPM
0	38	01-02-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	11.0	17.0	
1	38	01-07-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	17.0	
2	38	21-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	12.0	18.0	
3	38	23-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	15.0	16.0	
4	38	28-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	14.0	
...	...	...	...	...	...	...	...	...	...	...
2874	773	12-03-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	15.0	18.0	
2875	773	12-10-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	12.0	14.0	
2876	773	17-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	19.0	22.0	
2877	773	24-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	15.0	17.0	

	Stn Code	Sampling Date	State	City/Town /Village/Area	Location of Monitoring Station	Agency	Type of Location	SO2	NO2	RSPM
2878	773	31-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	14.0	16.0	

2879 rows × 11 columns

In [12]:
```
rawdata_df.columns
```

Out[12]:
```
Index(['Stn Code', 'Sampling Date', 'State', 'City/Town/Village/Area',
       'Location of Monitoring Station', 'Agency', 'Type of Location', 'SO2',
       'NO2', 'RSPM/PM10', 'PM 2.5'],
      dtype='object')
```

In [13]:
```
selected_columns=(['Stn Code', 'Sampling Date', 'State', 'City/Town/Village/Area',
       'Location of Monitoring Station', 'Agency', 'Type of Location', 'SO2',
       'NO2', 'RSPM/PM10', 'PM 2.5'])
```

Data Preparation and Cleaning

In [17]:
```
analysis_df = rawdata_df[selected_columns].copy()
```

In [18]:
```
analysis_df
```

Out[18]:

	Stn Code	Sampling Date	State	City/Town /Village/Area	Location of Monitoring Station	Agency	Type of Location	SO2	NO2	RSPM
0	38	01-02-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	11.0	17.0	
1	38	01-07-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	17.0	
2	38	21-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	12.0	18.0	
3	38	23-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	15.0	16.0	
4	38	28-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	14.0	
...	...	...	...	...	...	...	...	...	...	...
2874	773	12-03-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	15.0	18.0	
2875	773	12-10-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	12.0	14.0	
2876	773	17-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	19.0	22.0	
2877	773	24-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	15.0	17.0	

	Stn Code	Sampling Date	State	City/Town /Village/Area	Location of Monitoring Station	Agency	Type of Location	SO2	NO2	RSPM
2878	773	31-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	14.0	16.0	

2879 rows × 11 columns

Let's view some basic information about the data frame.

In [19]:
```
analysis_df.shape
```

Out[19]:
```
(2879, 11)
```

In [20]:
```
analysis_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2879 entries, 0 to 2878
Data columns (total 11 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   Stn Code                        2879 non-null   int64  
 1   Sampling Date                   2879 non-null   object 
 2   State                           2879 non-null   object 
 3   City/Town/Village/Area          2879 non-null   object 
 4   Location of Monitoring Station  2879 non-null   object 
 5   Agency                          2879 non-null   object 
 6   Type of Location                2879 non-null   object 
 7   SO2                             2868 non-null   float64
 8   NO2                             2866 non-null   float64
 9   RSPM/PM10                       2875 non-null   float64
 10  PM 2.5                          0 non-null      float64
dtypes: float64(4), int64(1), object(6)
memory usage: 247.5+ KB
```

Most columns have the data type object, either because they contain values of different types, or they contain empty values, which are represented using NaN. It appears that every column contains some empty values, since the Non-Null count for every column is lower than the total number of rows (29531). We'll need to deal with empty values and manually adjust the data type for each column on a case-by-case basis.

Only two of the columns were detected as contain empty values and we will drop the rows.

let's convert 'Date' columns into datetime64[ns] data type since its data type is objet.

analysis_df.describe()

In [23]:
```
analysis_df.describe()
```

Out[23]:

	Stn Code	SO2	NO2	RSPM/PM10	PM 2.5
count	2879.000000	2868.000000	2866.000000	2875.000000	0.0
mean	475.750261	11.503138	22.136776	62.494261	NaN
std	277.675577	5.051702	7.128694	31.368745	NaN
min	38.000000	2.000000	5.000000	12.000000	NaN
25%	238.000000	8.000000	17.000000	41.000000	NaN
50%	366.000000	12.000000	22.000000	55.000000	NaN
75%	764.000000	15.000000	25.000000	78.000000	NaN
max	773.000000	49.000000	71.000000	269.000000	NaN

In [26]:
```
analysis_df.dropna(subset=['SO2'], inplace=True)
analysis_df.dropna(subset=['NO2'], inplace=True)
```

In [27]:
```
analysis_df
```

Out[27]:

	Stn Code	Sampling Date	State	City/Town /Village/Area	Location of Monitoring Station	Agency	Type of Location	SO2	NO2	RSPM
0	38	01-02-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	11.0	17.0	
1	38	01-07-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	17.0	
2	38	21-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	12.0	18.0	
3	38	23-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	15.0	16.0	
4	38	28-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	14.0	
...	...	...	...	...	...	...	...	...	...	...
2874	773	12-03-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	15.0	18.0	
2875	773	12-10-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	12.0	14.0	
2876	773	17-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	19.0	22.0	
2877	773	24-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	15.0	17.0	

	Stn Code	Sampling Date	State	City/Town /Village/Area	Location of Monitoring Station	Agency	Type of Location	SO2	NO2	RSPM
2878	773	31-12-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	14.0	16.0	

2866 rows × 11 columns

In [29]:
```
analysis_df['Location of Monitoring Station']
```

Out[29]:
```
0       Kathivakkam, Municipal Kalyana Mandapam, Chennai
1       Kathivakkam, Municipal Kalyana Mandapam, Chennai
2       Kathivakkam, Municipal Kalyana Mandapam, Chennai
3       Kathivakkam, Municipal Kalyana Mandapam, Chennai
4       Kathivakkam, Municipal Kalyana Mandapam, Chennai
                           ...
2874                        Central Bus Stand, Trichy
2875                        Central Bus Stand, Trichy
2876                        Central Bus Stand, Trichy
2877                        Central Bus Stand, Trichy
2878                        Central Bus Stand, Trichy
Name: Location of Monitoring Station, Length: 2866, dtype: object
```

Let's view again basic information about the data frame.

In [30]:
```
analysis_df.shape
```

Out[30]:
```
(2866, 11)
```

In [31]:
```
analysis_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 2866 entries, 0 to 2878
Data columns (total 11 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Stn Code                        2866 non-null   int64
 1   Sampling Date                   2866 non-null   object
 2   State                           2866 non-null   object
 3   City/Town/Village/Area          2866 non-null   object
 4   Location of Monitoring Station  2866 non-null   object
 5   Agency                          2866 non-null   object
 6   Type of Location                2866 non-null   object
 7   SO2                             2866 non-null   float64
 8   NO2                             2866 non-null   float64
 9   RSPM/PM10                       2862 non-null   float64
 10  PM 2.5                          0 non-null      float64
dtypes: float64(4), int64(1), object(6)
memory usage: 268.7+ KB
```

In [32]:
```
analysis_df.describe()
```

Out[32]:

	Stn Code	SO2	NO2	RSPM/PM10	PM 2.5
count	2866.000000	2866.000000	2866.000000	2862.000000	0.0
mean	475.153524	11.501047	22.136776	62.437456	NaN
std	277.688772	5.052689	7.128694	31.277419	NaN
min	38.000000	2.000000	5.000000	12.000000	NaN
25%	238.000000	8.000000	17.000000	41.000000	NaN
50%	366.000000	12.000000	22.000000	55.000000	NaN
75%	764.000000	15.000000	25.000000	78.000000	NaN
max	773.000000	49.000000	71.000000	269.000000	NaN

After Data Cleaning like removing empty rows and changing data type.

Exploratory Analysis and Visualization

It's important to Visualize the data to understand the data clearly.For that we will matplotlib / seaborn library.

Let's begin by importing matplotlib.pyplot and seaborn.

In [33]:
```
pip install seaborn
```

```
Requirement already satisfied: seaborn in c:\users\vicky\anaconda3\lib\site-package
s (0.12.2)
Requirement already satisfied: numpy!=1.24.0,>=1.17 in c:\users\vicky\anaconda3\li
b\site-packages (from seaborn) (1.24.3)
Requirement already satisfied: pandas>=0.25 in c:\users\vicky\anaconda3\lib\site-pa
ckages (from seaborn) (2.0.3)
Requirement already satisfied: matplotlib!=3.6.1,>=3.1 in c:\users\vicky\anaconda3\
lib\site-packages (from seaborn) (3.7.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\vicky\anaconda3\lib\sit
e-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.0.5)
Requirement already satisfied: cycler>=0.10 in c:\users\vicky\anaconda3\lib\site-pa
ckages (from matplotlib!=3.6.1,>=3.1->seaborn) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\vicky\anaconda3\lib\si
te-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (4.25.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\vicky\anaconda3\lib\si
te-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\vicky\anaconda3\lib\site
-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (23.1)
Requirement already satisfied: pillow>=6.2.0 in c:\users\vicky\anaconda3\lib\site-p
ackages (from matplotlib!=3.6.1,>=3.1->seaborn) (9.4.0)
Requirement already satisfied: pyparsing<3.1,>=2.3.1 in c:\users\vicky\anaconda3\li
b\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\vicky\anaconda3\li
b\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\vicky\anaconda3\lib\site-pa
ckages (from pandas>=0.25->seaborn) (2023.3.post1)
Requirement already satisfied: tzdata>=2022.1 in c:\users\vicky\anaconda3\lib\site-
packages (from pandas>=0.25->seaborn) (2023.3)
Requirement already satisfied: six>=1.5 in c:\users\vicky\anaconda3\lib\site-packag
es (from python-dateutil>=2.7->matplotlib!=3.6.1,>=3.1->seaborn) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
```

In [34]:
```
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline

sns.set_style('darkgrid')
matplotlib.rcParams['font.size'] = 14
matplotlib.rcParams['figure.figsize'] = (9, 5)
matplotlib.rcParams['figure.facecolor'] = '#00000000'
```

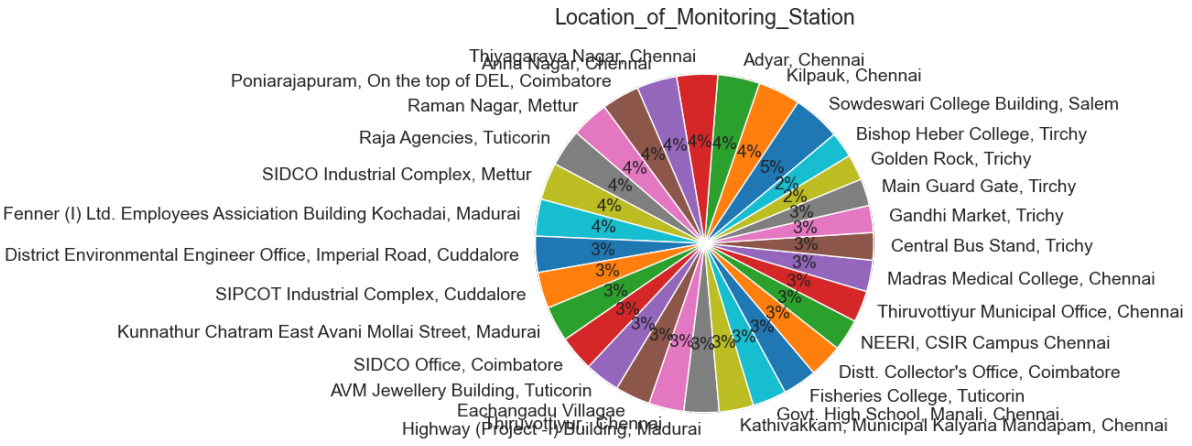
In [40]:
```
Location_of_Monitoring_Station = analysis_df['Location of Monitoring Station'].valu
Location_of_Monitoring_Station
```

Out[40]:

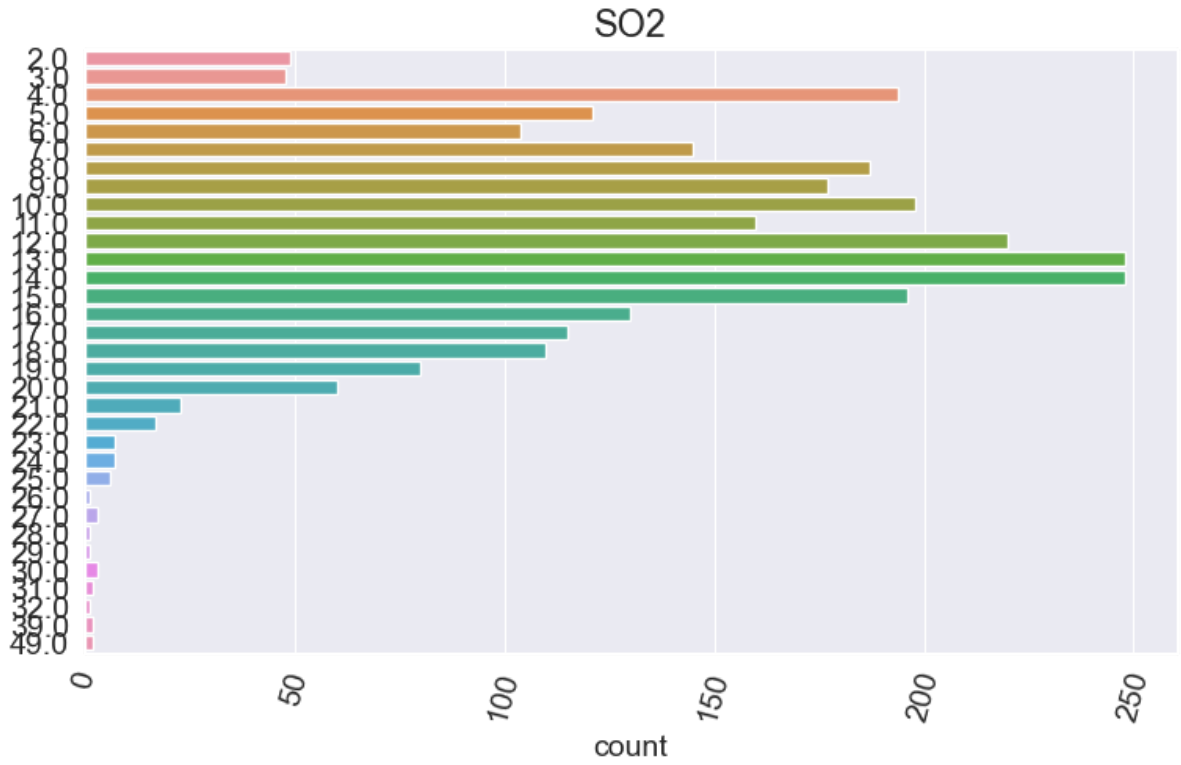
Location of Monitoring Station	
Sowdeswari College Building, Salem	131
Kilpauk, Chennai	116
Adyar, Chennai	115
Thiyagaraya Nagar, Chennai	112
Anna Nagar, Chennai	110
Poniarajapuram, On the top of DEL, Coimbatore	103
Raman Nagar, Mettur	103
Raja Agencies, Tuticorin	102
SIDCO Industrial Complex, Mettur	102
Fenner (I) Ltd. Employees Assiciation Building Kochadai, Madurai	101
District Environmental Engineer Office, Imperial Road, Cuddalore	99
SIPCOT Industrial Complex, Cuddalore	99
Kunnathur Chatram East Avani Mollai Street, Madurai	97
SIDCO Office, Coimbatore	97
AVM Jewellery Building, Tuticorin	96
Eachangadu Villagae	96
Thiruvottiyur, Chennai	96
Highway (Project -I) Building, Madurai	96
Kathivakkam, Municipal Kalyana Mandapam, Chennai	94
Govt. High School, Manali, Chennai.	93
Fisheries College, Tuticorin	93
Distt. Collector's Office, Coimbatore	92
NEERI, CSIR Campus Chennai	87
Thiruvottiyur Municipal Office, Chennai	86
Madras Medical College, Chennai	86
Central Bus Stand, Trichy	75
Gandhi Market, Trichy	74
Main Guard Gate, Tirchy	74
Golden Rock, Trichy	71
Bishop Heber College, Tirchy	70

Name: count, dtype: int64

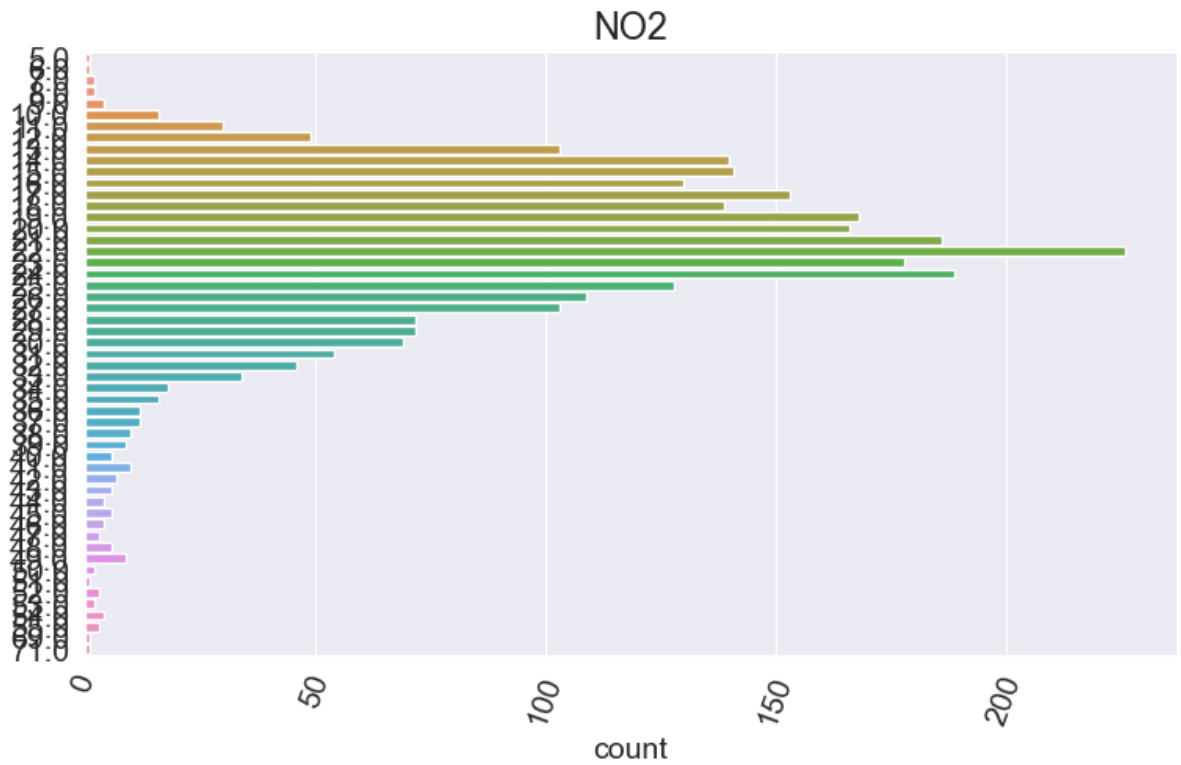
In [57]:
```
plt.figure(figsize=(10,6))
plt.title('Location_of_Monitoring_Station')
plt.pie(Location_of_Monitoring_Station, labels=Location_of_Monitoring_Station.index
```



In [60]:
```
sns.countplot(y=analysis_df.SO2)
plt.xticks(rotation=75);
plt.title('SO2')
plt.ylabel(None);
```



In [65]:
```
sns.countplot(y=analysis_df.NO2)
plt.xticks(rotation=70);
plt.title('NO2')
plt.ylabel(None);
```



In [ ]: