PHASE 4:

AIR QUALITY ANALYSIS

VISUALIZING AND IMPLEMENTING

In this phase we will be visualizing our dataset using python.

```
#The modules we import for the process
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sb

Importing our dataset
data = pd.read_csv("cpcb_dly_aq_tamil_nadu-2014.csv")
display(data.head(), data.tail())
```

	Stn Code	Sampling Date	State	City/Town/Village/Area	Location of Monitoring Station	Agency	Type of Location	SO2	NO2	RSPM/PM10	PM 2.5
0	38	01-02-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	11.0	17.0	55.0	NaN
1	38	01-07-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	17.0	45.0	NaN
2	38	21-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	12.0	18.0	50.0	NaN
3	38	23-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	15.0	16.0	46.0	NaN
4	38	28-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	14.0	42.0	NaN

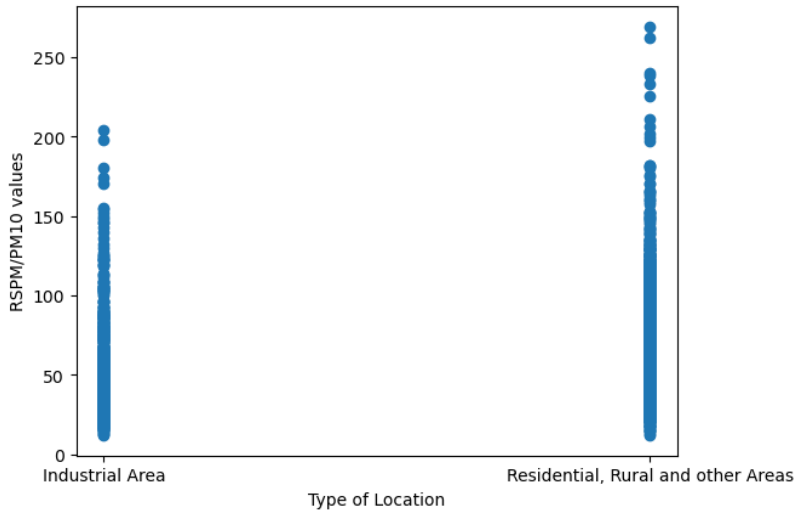
	Stn Code	Sampling Date	State	City/Town/Village/Area	Location of Monitoring Station	Agency	Type of Location	SO2	NO2	RSPM/PM10	PM 2.5
2874	773	12-03-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	15.0	18.0	102.0	NaN
2875	773	12-10-14	Tamil Nadu	Trichy	Central Bus Stand, Trichy	Tamilnadu State Pollution Control Board	Residential, Rural and other Areas	12.0	14.0	91.0	NaN
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	100.0	NaN
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	95.0	NaN
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	94.0	NaN

In [3]:
```
data.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
```

#Visualization

In [4]:
```
plt.scatter(data['Type of Location'], data["RSPM/PM10"])
plt.xlabel('Type of Location')
plt.ylabel('RSPM/PM10 values')
plt.show()
```



#Code to find the area condition based on the RSPM/PM10 value

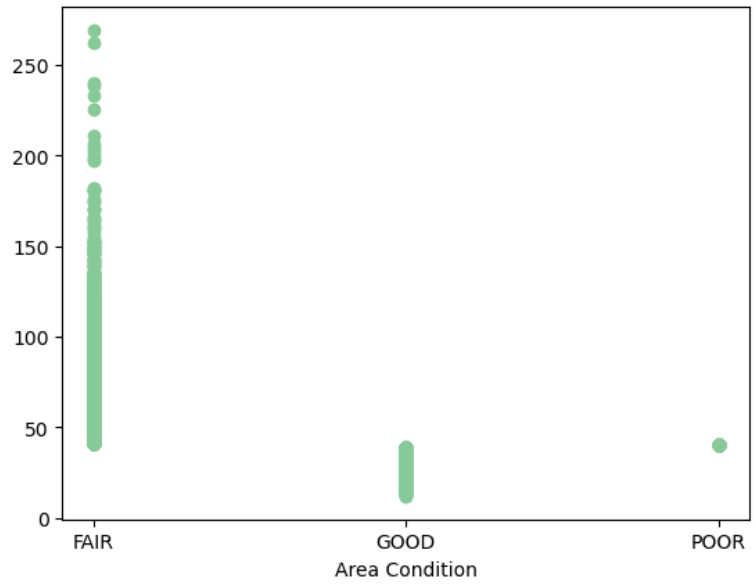
```
allConditions = ["GOOD", "FAIR", "POOR", "VERY POOR", "EXTREMELY POOR"]
Condition = "NOT SURE"
mean_value=data['RSPM/PM10'].mean()
data['RSPM/PM10'].fillna(value=mean_value, inplace=True)
Conditionsx = []

g,f,p,vp,ep = 0,0,0,0,0
for i in data["RSPM/PM10"]:
    j = int(i)
    if j < 40:
        Condition = allConditions[0]
        g += 1
    elif j > 40 & j < 80:
        Condition = allConditions[1]
        f += 1
    elif j > 80 & j < 120:
        Condition = allConditions[2]
        p += 1
    elif j > 120 &  j < 300:
        Condtion = allConditions[3]
        vp += 1
    else:
        Condition = allConditions[4]
        ep += 1
    Conditionsx.append(str(Condition))
```

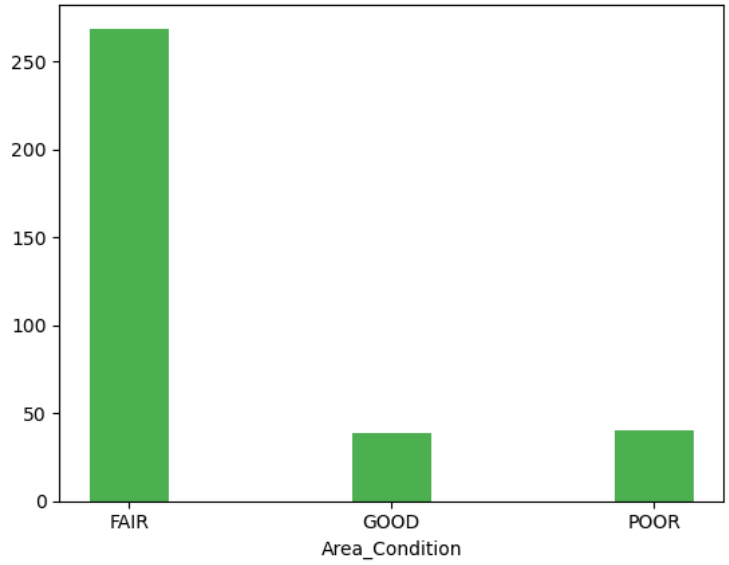
```
#Area_Condition = {"Area_Condition" : Conditionsx}
data['Area_Condition_based on RSPM/PM10'] = Conditionsx
display(data.head())
```

	Stn Code	Sampling Date	State	City/Town/Village/Area	Location of Monitoring Station	Agency	Type of Location	SO2	NO2	RSPM/PM10	PM 2.5	Area_Condition	Area_Condition_based on RSPM/PM10
0	38	01-02-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	11.0	17.0	55.0	0.0	FAIR	FAIR
1	38	01-07-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	17.0	45.0	0.0	FAIR	FAIR
2	38	21-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	12.0	18.0	50.0	0.0	FAIR	FAIR
3	38	23-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	15.0	16.0	46.0	0.0	FAIR	FAIR
4	38	28-01-14	Tamil Nadu	Chennai	Kathivakkam, Municipal Kalyana Mandapam, Chennai	Tamilnadu State Pollution Control Board	Industrial Area	13.0	14.0	42.0	0.0	FAIR	FAIR

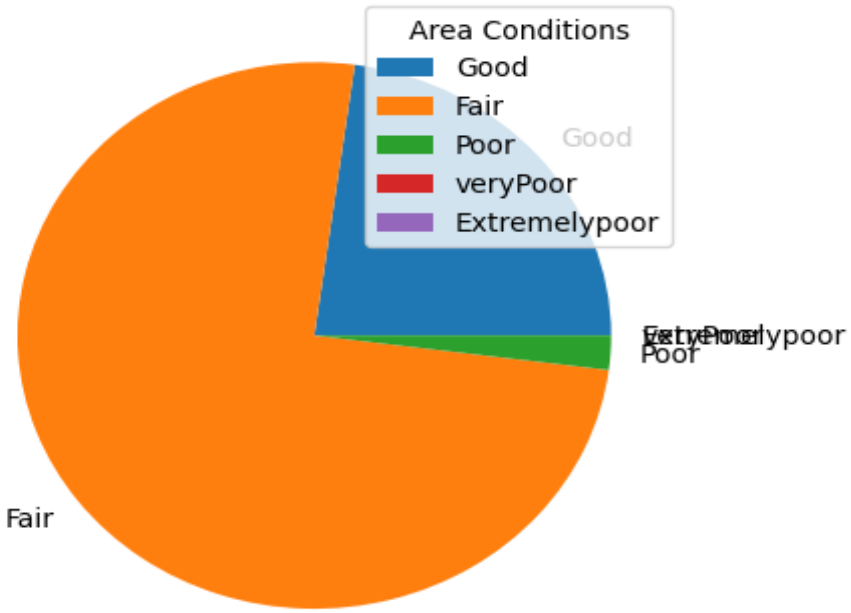
```
plt.scatter(data['Area_Condition_based on RSPM/PM10'], data["RSPM/PM10"],color = '#88c999')
plt.xlabel("Area Condition")
plt.show()
```



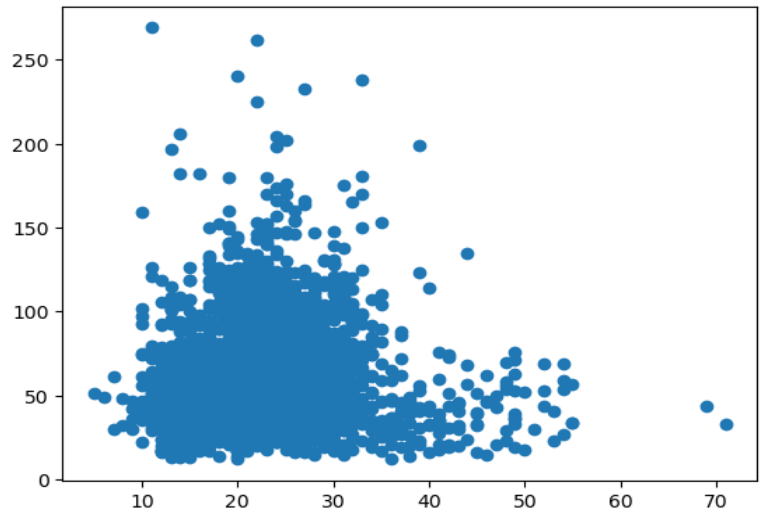
```
plt.bar(data['Area_Condition_based on RSPM/PM10'], data["RSPM/PM10"],width = 0.3, color = "#4CAF50")
plt.xlabel("Area_Condition")
plt.show()
```




```
cnd = np.array([g,f,p,vp,ep])
mylabels = ["Good","Fair","Poor", "veryPoor","Extremelypoor"]
myexp =[0,0,0,0,0]
plt.pie(cnd,labels = mylabels,explode = myexp,startangle = 0)
plt.legend(mylabels, title = "Area Conditions")
plt.show()
```



```
plt.scatter(x = data["NO2"], y = data["RSPM/PM10"])
plt.show()
```



```
plt.scatter(y = data["SO2"], x = data["RSPM/PM10"])
plt.show()
```