Multi-factor analysis of the air quality index in different cities of India

Dated: 8 September 2022

Section 1

Data Analysis Using Python (DAP)

Term 4

By

Aditya Mande
Harshita Nagaich
Ishan Kapse
Priyansha Sharma
Siddharth Sarawade
Meet Shah

Indian Institute of Management Shillong

Introduction to Data Analytics

What is Data Analytics?

Data analytics is the science of analysing data to draw inferences and conclusions. Many information analytics methods and procedures are automated into mechanical workflows and algorithms that process data for human consumption.

Data analytics methods can show trends and indicators that may have been a little obscured by the wealth of information. Using this information, operations can then be optimised to increase a business's or a system's efficiency.

The term "data analytics" refers to the methods used to analyse data in order to boost productivity and financial gain. Massive amounts of unstructured data must be sorted through in order to extract important insights. The decision-making process at businesses of all sizes can greatly benefit from these insights. These insights are enormously valuable for decision-making at companies of all sizes. Data is extracted from varied sources and is then cleaned and categorized to investigate various behavioural patterns. The techniques and therefore the tools used vary per the organization or individual.

Why is Data Analytics important?

The uses for knowledge analytics are numerous. Analyzing the big data can optimize efficiency in many various industries. Businesses can succeed in an environment where competition is escalating by improving performance. In a plethora of different industries, data analytics is being applied quite successfully.

The banking industry was one of the first to adopt. In the banking and financial sectors, data analytics plays a significant role in predicting market trends and evaluating risk. Everyone is impacted by information analytics, such as credit ratings. These scores assess lending risk using a variety of data points. For financial organisations to increase efficiency and lower risk, data analytics is also used to detect and stop fraud. However, the utilisation of information analytics goes beyond increasing revenue and ROI. For the purposes of environmental protection, criminal prevention, and health informatics, data analytics can give vital information. These information analytics applications improve our world by utilising these methods. Despite the fact that statistics and data analysis have long been used in research, modern analytic methods and vast amounts of data now yield many new insights.

Process generally followed in Data Analytics:

1. The first step is to work out the information requirements or how the information is grouped. Data could also be separated by age, demographic, income, or gender. Data values could also be numerical or be divided by category.
2. The second step in data analytics is that the process of collecting it. This could be done through a range of sources like computers, online sources, cameras, environmental sources, or through personnel.
3. Once the info is collected, it must be organized so it is analyzed. Organization may happen on a spreadsheet or other sort of software that may take statistical data.
4. The data is then cleaned up before analysis. this suggests it's scrubbed and checked to make sure there's no duplication or error, which it's not incomplete. This step helps correct any errors before it goes on to an information analyst to be analyzed.

Types of Data Analytics:

The field of data analytics may be very vast. Descriptive, diagnostic, predictive, and prescriptive analytics are the four main types of data analysis. Each category has a distinct objective and role in the data analysis process. These are the first commercial applications of data analytics.

1. Descriptive Analytics
2. Diagnostic Analytics
3. Predictive Analytics
4. Prescriptive Analytics

Tools used in Data Analytics:

With the increasing demand for DA in the market, many tools have emerged. Either open-source or user-friendly, the top tools are R programming, Python ,Tableau Public, SAS, Microsoft Excel etc.

Title of the Project

Multi-factor analysis of the air quality index in different cities of India from 2015 to 2020 and to analyze the trends observed in the growth or reduction of different pollutant particulate matter in the atmosphere of these cities during the given time period.

Problem Statement

India has been facing a major issue with its air quality and has been having many cities in the top 100 most polluted cities across the globe. In order to evaluate the major reasons for the poor air quality for various cities across India, with the help of multi-factor analysis, we aim to understand various environmental factors and parameters which affect the air quality in cities and suggest what should be focussed on by the country and different cities separately to improve the air quality in the cities across India.

Advantages of Solving the problem:

- Understand the reasons that certain cities do exceptionally well on the air quality index as compared to the other cities.
- Break myths about air quality index. For example, the myth that metro cities are usually the most polluted in terms of air and have poor air quality as a result.
- Understand the improvement areas that Indians can focus their attention on to improve the overall well-being of society. Also, to understand that parameters on which a few cities score well and what can other cities learn from them.
- Track the progress of different cities on the air quality scores to identify cities who have made strong progress between 2015 and 2019.

Data Description:

Data Source - https://www.kaggle.com/datasets/rohanrao/air-quality-data-in-india?select=city_day.csv

The chosen dataset has 4 separate files for the years from 2015 till 2020. We have selected the city_day dataset which consists of 16 columns which talk about the cities, dates and other components of air that constitute the quality of air in the region. The factors include PM 2.5, PM 10, NO, NO2, NOx, NH3, CO, SO2, O3, Benzene, Toluene, Xylene, AQI, AQI Bucket. This set of data gives a detail of all the factors scaled to a numeric value for all cities across years. Hence, using this dataset, the data can be directly analyzed to gather interrelations and weightage of various factors affecting the air quality index of the cities.

Methodology of Data Collection

The data is sourced from kaggle.com, where it has been aggregated in turn from trak.in

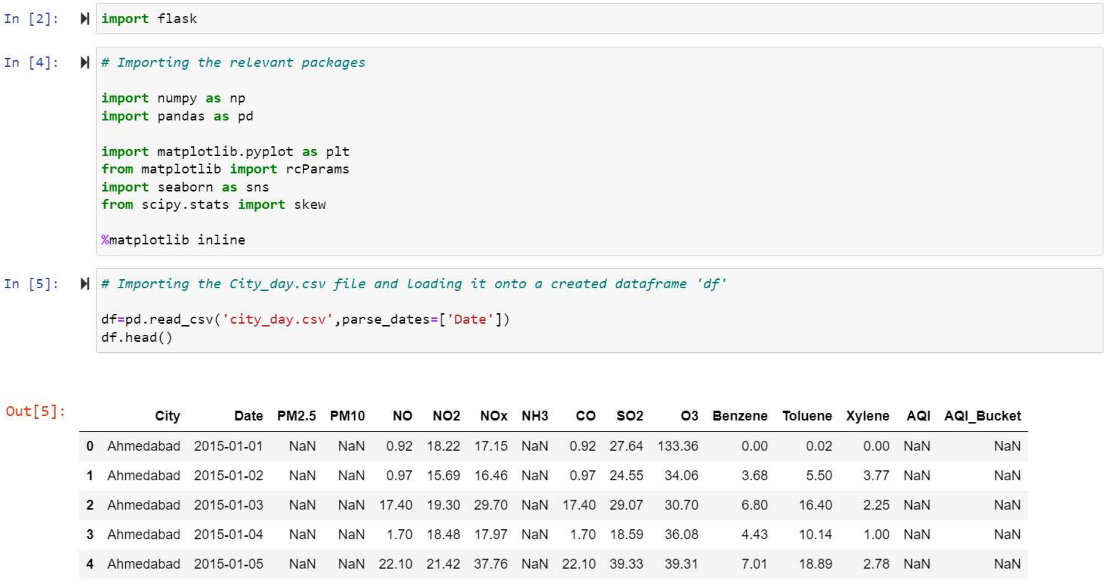
Data Link - https://www.kaggle.com/sudalairajkumar/indian-startup-funding/

The data consists of the following columns -

1. **Cities –** Name of the City
2. **Date dd/mm/yyyy -** Date in dd/mm/yyyy format
3. **PM 2.**5- Levels of PM 2.5
4. **PM 10-** Levels of PM 10
5. **NO-** Levels of NO
6. **NO2-** Levels of NO2
7. **NOx-** Levels of NOx
8. **NH3-** Levels of NH3
9. **CO -** Levels of CO
10. **SO2-** Levels of SO2
11. **O3-** Levels of O3
12. **Benzene-** Levels of Benzene
13. **Toluene-** Levels of Toluene
14. **Xylene-** Levels of Xylene
15. **AQI-** The AQI considering all the factors from 3-14
16. **AQI Bucket-** Calculated using AQI Bucket

Here all the columns from 3 to 16 are the ones which contribute to the AQI calculation which gives the air quality index of a city and further helps in analyzing the overall AQI of all cities over the years and gain some insights for further research and analysis.

Data Cleaning and Analytics:



```
In [2]:  import flask
```

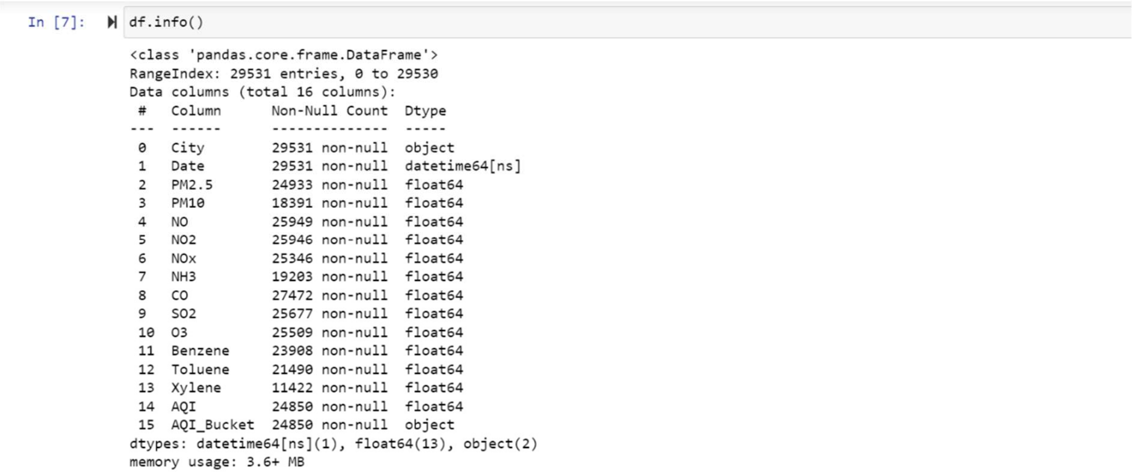
```
In [4]:  # Importing the relevant packages

         import numpy as np
         import pandas as pd

         import matplotlib.pyplot as plt
         from matplotlib import rcParams
         import seaborn as sns
         from scipy.stats import skew

         %matplotlib inline
```

```
In [5]:  # Importing the City_day.csv file and loading it onto a created dataframe 'df'

         df=pd.read_csv('city_day.csv',parse_dates=['Date'])
         df.head()
```

Out[5]:

	City	Date	PM2.5	PM10	NO	NO2	NOx	NH3	CO	SO2	O3	Benzene	Toluene	Xylene	AQI	AQI_Bucket
0	Ahmedabad	2015-01-01	NaN	NaN	0.92	18.22	17.15	NaN	0.92	27.64	133.36	0.00	0.02	0.00	NaN	NaN
1	Ahmedabad	2015-01-02	NaN	NaN	0.97	15.69	16.46	NaN	0.97	24.55	34.06	3.68	5.50	3.77	NaN	NaN
2	Ahmedabad	2015-01-03	NaN	NaN	17.40	19.30	29.70	NaN	17.40	29.07	30.70	6.80	16.40	2.25	NaN	NaN
3	Ahmedabad	2015-01-04	NaN	NaN	1.70	18.48	17.97	NaN	1.70	18.59	36.08	4.43	10.14	1.00	NaN	NaN
4	Ahmedabad	2015-01-05	NaN	NaN	22.10	21.42	37.76	NaN	22.10	39.33	39.31	7.01	18.89	2.78	NaN	NaN

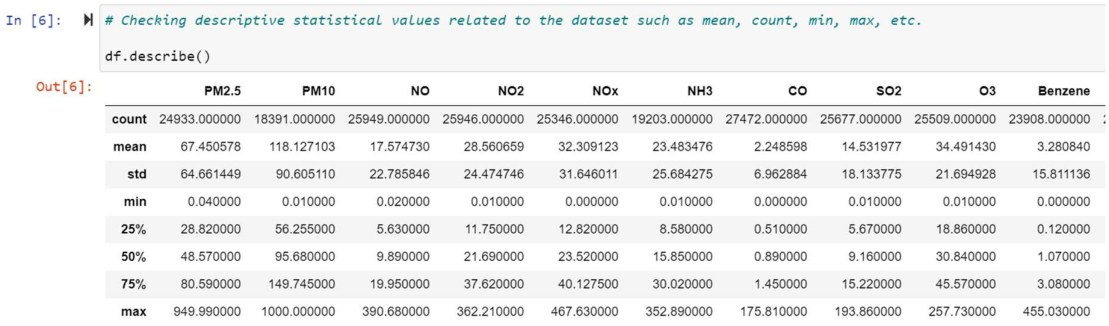
Here we have imported the various libraries that would be required for the analysis and read the CSV file for getting the data where we see that data having 16 columns have been imported for analysis.



```
In [7]:  df.info()

         <class 'pandas.core.frame.DataFrame'>
         RangeIndex: 29531 entries, 0 to 29530
         Data columns (total 16 columns):
          #   Column      Non-Null Count  Dtype
         ---  ------      --------------  -----
          0   City        29531 non-null  object
          1   Date        29531 non-null  datetime64[ns]
          2   PM2.5       24933 non-null  float64
          3   PM10        18391 non-null  float64
          4   NO          25949 non-null  float64
          5   NO2         25946 non-null  float64
          6   NOx         25346 non-null  float64
          7   NH3         19203 non-null  float64
          8   CO          27472 non-null  float64
          9   SO2         25677 non-null  float64
          10  O3          25509 non-null  float64
          11  Benzene     23908 non-null  float64
          12  Toluene     21490 non-null  float64
          13  Xylene      11422 non-null  float64
          14  AQI         24850 non-null  float64
          15  AQI_Bucket  24850 non-null  object
         dtypes: datetime64[ns](1), float64(13), object(2)
         memory usage: 3.6+ MB
```

Here, we can see that the data has a total of 29531 rows and 16 columns covering the data of City, Date, PM2.5, PM10, NO, NO2, NO3, NOx, NH3, CO, SO2, O3, Benzene, Toluene, Xylene, AQI and AQI_Bucket

In this we also find that some of the data spaces are null or blank. We will be handling this data further.



```
In [6]:  # Checking descriptive statistical values related to the dataset such as mean, count, min, max, etc.

         df.describe()
```

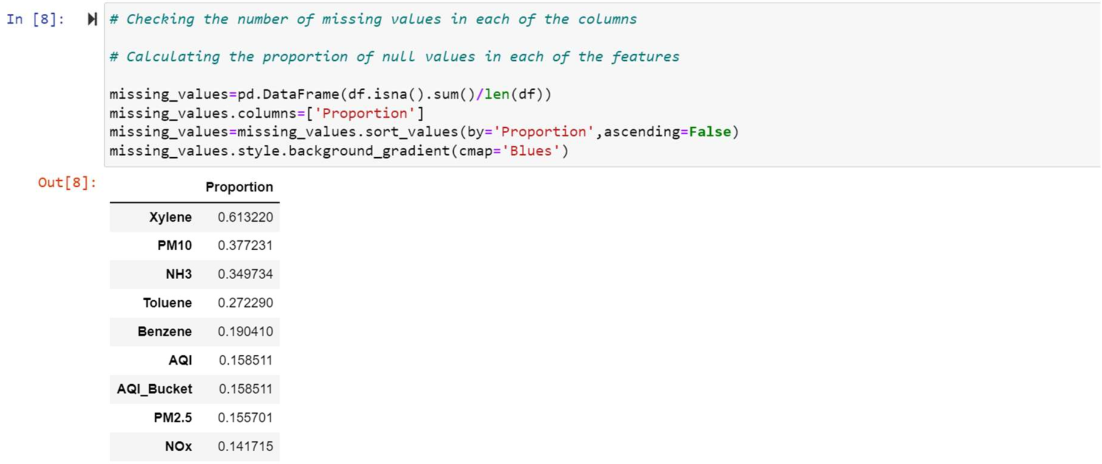
Out[6]:

	PM2.5	PM10	NO	NO2	NOx	NH3	CO	SO2	O3	Benzene
count	24933.000000	18391.000000	25949.000000	25946.000000	25346.000000	19203.000000	27472.000000	25677.000000	25509.000000	23908.000000
mean	67.450578	118.127103	17.574730	28.560659	32.309123	23.483476	2.248598	14.531977	34.491430	3.280840
std	64.661449	90.605110	22.785846	24.474746	31.646011	25.684275	6.962884	18.133775	21.694928	15.811136
min	0.040000	0.010000	0.020000	0.010000	0.000000	0.010000	0.000000	0.010000	0.010000	0.000000
25%	28.820000	56.255000	5.630000	11.750000	12.820000	8.580000	0.510000	5.670000	18.860000	0.120000
50%	48.570000	95.680000	9.890000	21.690000	23.520000	15.850000	0.890000	9.160000	30.840000	1.070000
75%	80.590000	149.745000	19.950000	37.620000	40.127500	30.020000	1.450000	15.220000	45.570000	3.080000
max	949.990000	1000.000000	390.680000	362.210000	467.630000	352.890000	175.810000	193.860000	257.730000	455.030000

Here, we can see that the data about mean, max, min, std dev etc. has been calculated for all the parameters that contribute to the calculation of the Air Quality Index. Hence, we calculate the AQI for all the cities on the days to for further analysis as they reciprocate the levels of all other factors using which it is calculated.

From the above table, we can see that the average AQI score is approximately 166, the minimum AQI score is 13 and the maximum AQI score is 2049. Since the 75 percentile AQI value is 208, which is very close to the average of 166, we can deduce that majority of the points on the dataset fall within the 0-200 range.

This puts these cities in a Moderate AQI range. Moderate AQI levels can possibly cause breathing discomfort to people with lung diseases, Asthma, and Heart diseases.

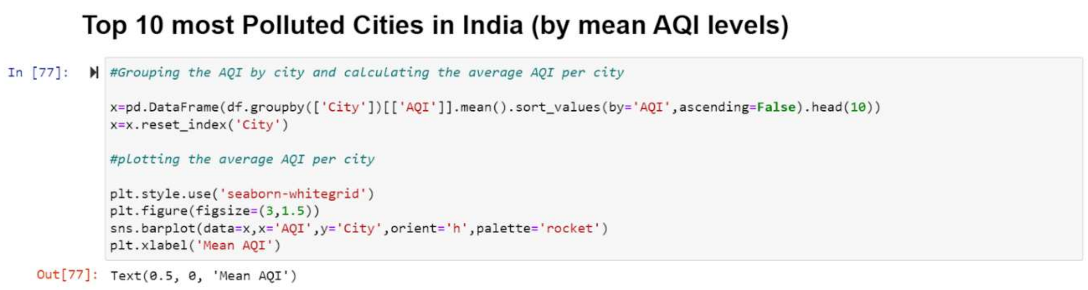


```
In [8]:  # Checking the number of missing values in each of the columns

         # Calculating the proportion of null values in each of the features

         missing_values=pd.DataFrame(df.isna().sum()/len(df))
         missing_values.columns=['Proportion']
         missing_values=missing_values.sort_values(by='Proportion',ascending=False)
         missing_values.style.background_gradient(cmap='Blues')
```

Out[8]:

	Proportion
Xylene	0.613220
PM10	0.377231
NH3	0.349734
Toluene	0.272290
Benzene	0.190410
AQI	0.158511
AQI_Bucket	0.158511
PM2.5	0.155701
NOx	0.141715

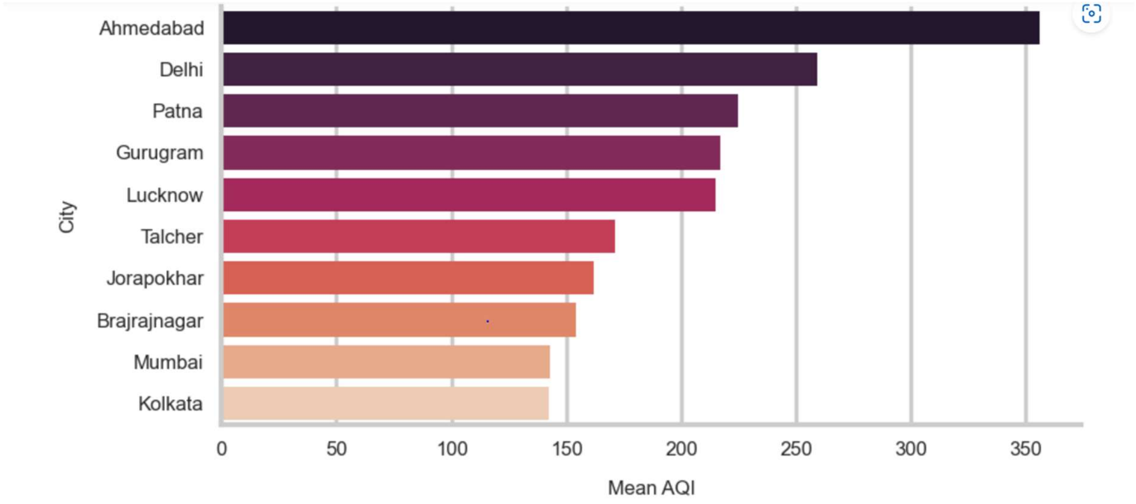
Certain columns in the dataset have a higher percentage of missing values. For example Xylene has 61.32% missing values. PM10 has 37.72% missing values.
Since AQI is a combined metric inclusive of the particulate matters, we focus on the AQI column. The AQI columns has 15.85% missing values.
Since we will be using the mean values in our AQI analysis, we will use a Simple Imputer to fill up the missing values in the AQI column.
We will leave the other columns as they are since they have a small role to play in our analysis.



Top 10 most Polluted Cities in India (by mean AQI levels)

```
In [77]:  #Grouping the AQI by city and calculating the average AQI per city

          x=pd.DataFrame(df.groupby(['City'])[['AQI']].mean().sort_values(by='AQI',ascending=False).head(10))
          x=x.reset_index('City')

          #plotting the average AQI per city

          plt.style.use('seaborn-whitegrid')
          plt.figure(figsize=(3,1.5))
          sns.barplot(data=x,x='AQI',y='City',orient='h',palette='rocket')
          plt.xlabel('Mean AQI')

Out[77]:  Text(0.5, 0, 'Mean AQI')
```

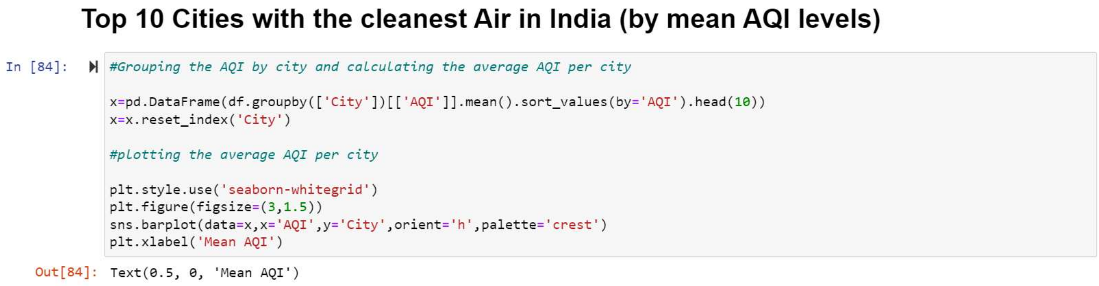
Results are as follows:



Hence, here we see that the AQI values for Ahmedabad is the highest across all the cities in India followed by Delhi and Patna.

Also we can see the majority of metro-cities in the top 10, not necessarily by population or industry though as Mumbai and Delhi are down in the list. Chennai is not seen in the top 10 list as well. At the same time smaller cities like Talcher, Brajrajnagar, Jorpokhar are in the list of the top 10 cities.

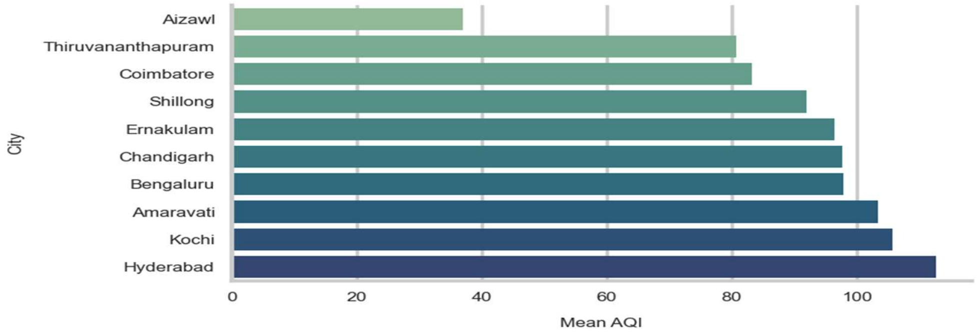
Through the above graph, we can see that among the top 10 most polluted cities based on Mean AQI, the mean AQI of the top 5 cities (Ahmedabad, Delhi, Patna, Gurugram, and Lucknow) is above 200. This places those cities in the 'Poor' category on the AQI scale where the health risks would be breathing discomfort and development of breathing problems in most people on prolonged exposure.



Top 10 Cities with the cleanest Air in India (by mean AQI levels)

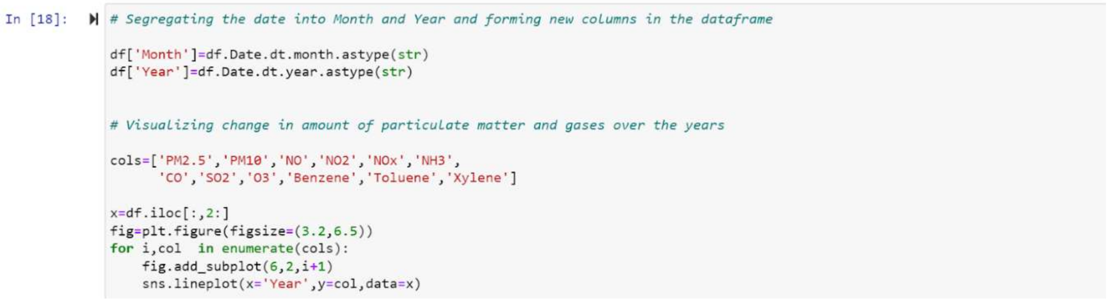
```
In [84]:  #Grouping the AQI by city and calculating the average AQI per city

          x=pd.DataFrame(df.groupby(['City'])[['AQI']].mean().sort_values(by='AQI').head(10))
          x=x.reset_index('City')

          #plotting the average AQI per city

          plt.style.use('seaborn-whitegrid')
          plt.figure(figsize=(3,1.5))
          sns.barplot(data=x,x='AQI',y='City',orient='h',palette='crest')
          plt.xlabel('Mean AQI')

Out[84]:  Text(0.5, 0, 'Mean AQI')
```

The results are as follows:

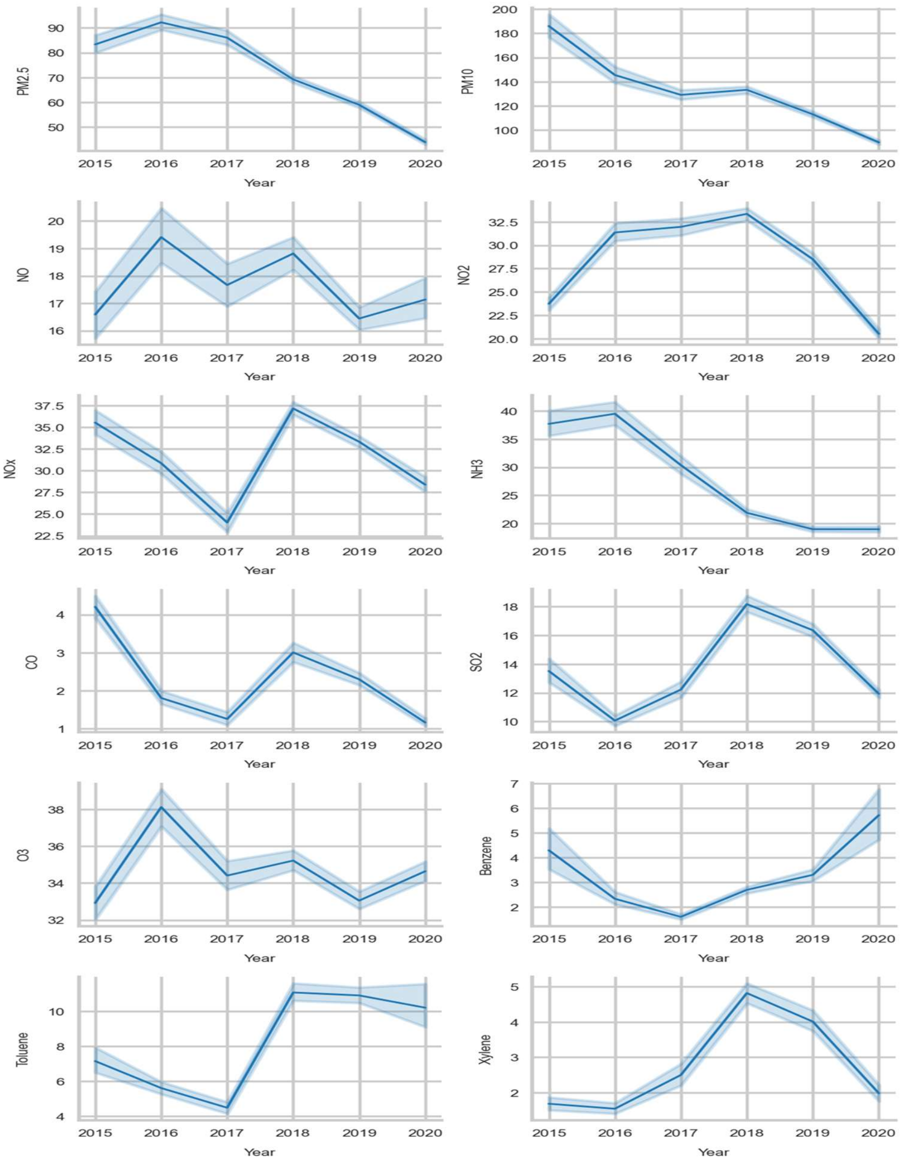


Through the above graph, we can see that among the top 10 least polluted cities based on Mean AQI, the mean AQI of the top 7 cities with cleanest air (Aizawl, Thiruvanthapuram, Coimbatore, Shillong, Ernakulam, Chandigarh, and Bengaluru) is below 100. This places those cities in the 'Satisfactory' air quality category where only sensitive people are prone to experiencing a minor level of discomfort breathing.

Change in the concentration of individual Particulate matter over the years



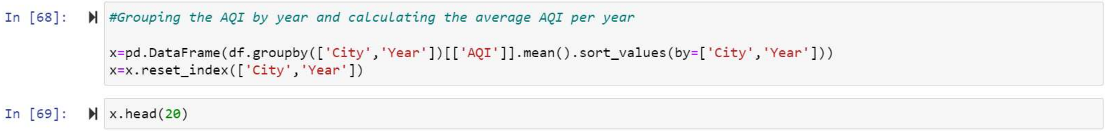
```
In [18]:  # Segregating the date into Month and Year and forming new columns in the dataframe

           df['Month']=df.Date.dt.month.astype(str)
           df['Year']=df.Date.dt.year.astype(str)


           # Visualizing change in amount of particulate matter and gases over the years

           cols=['PM2.5','PM10','NO','NO2','NOx','NH3',
                 'CO','SO2','O3','Benzene','Toluene','Xylene']

           x=df.iloc[:,2:]
           fig=plt.figure(figsize=(3.2,6.5))
           for i,col  in enumerate(cols):
               fig.add_subplot(6,2,i+1)
               sns.lineplot(x='Year',y=col,data=x)
```



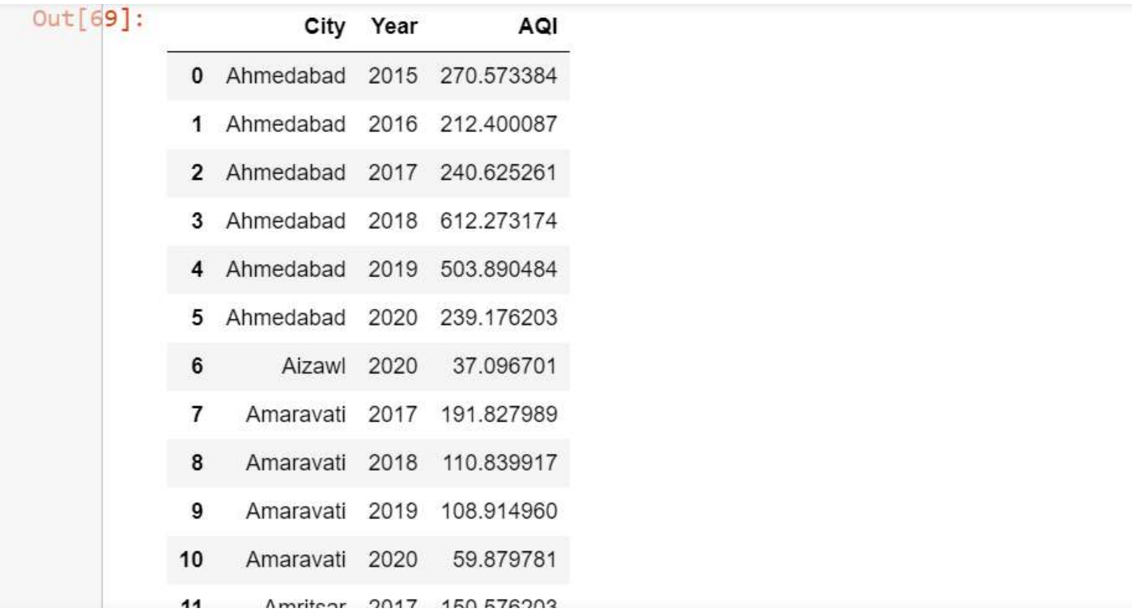
Of these 12 types of particulate matters, 7 types of particulate matter primarily arise from Vehicular pollution and the other 5 types of particulate matter primarily arise from Industrial pollution.

Vehicular pollution particulate matter - 'PM2.5','PM10','NO','NO2','NOx','NH3','CO'
Industrial pollution particulate matter - 'SO2','O3','Benzene','Toluene','Xylene'

We can see that from 2015 to 2020, the amount of particulates relating to vehicular pollution have gone down. However, when we look at the amount of particulates related to industrial pollution, we can see that the amount of Benzene and Toluene in the air has shot up. We need to investigate the reason for this increase.



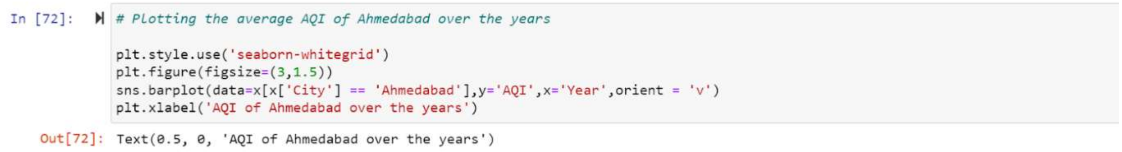
```
In [68]:  #Grouping the AQI by year and calculating the average AQI per year

           x=pd.DataFrame(df.groupby(['City','Year'])[['AQI']].mean().sort_values(by=['City','Year']))
           x=x.reset_index(['City','Year'])

In [69]:  x.head(20)
```



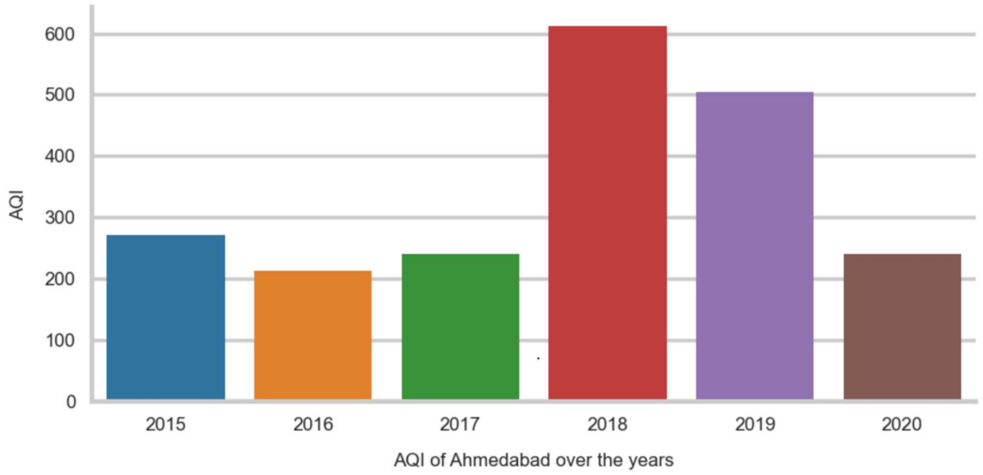
Out[69]:

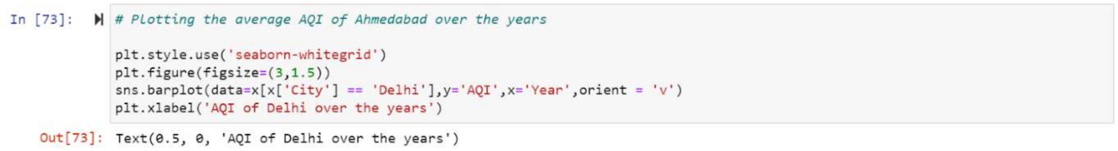
	City	Year	AQI
0	Ahmedabad	2015	270.573384
1	Ahmedabad	2016	212.400087
2	Ahmedabad	2017	240.625261
3	Ahmedabad	2018	612.273174
4	Ahmedabad	2019	503.890484
5	Ahmedabad	2020	239.176203
6	Aizawl	2020	37.096701
7	Amaravati	2017	191.827989
8	Amaravati	2018	110.839917
9	Amaravati	2019	108.914960
10	Amaravati	2020	59.879781
11	Amritsar	2017	150.576203

This gives the value of all the AQI of the cities over the years which help is observe the change in the AQI value from year to year. Through this we get key insights like The AQI of Ahmedabad has increased in the initial years but have seen a drop in the last year of 2020. On the other hand, Amravati, Amritsar and Bengaluru have seen a constant decline over the years.

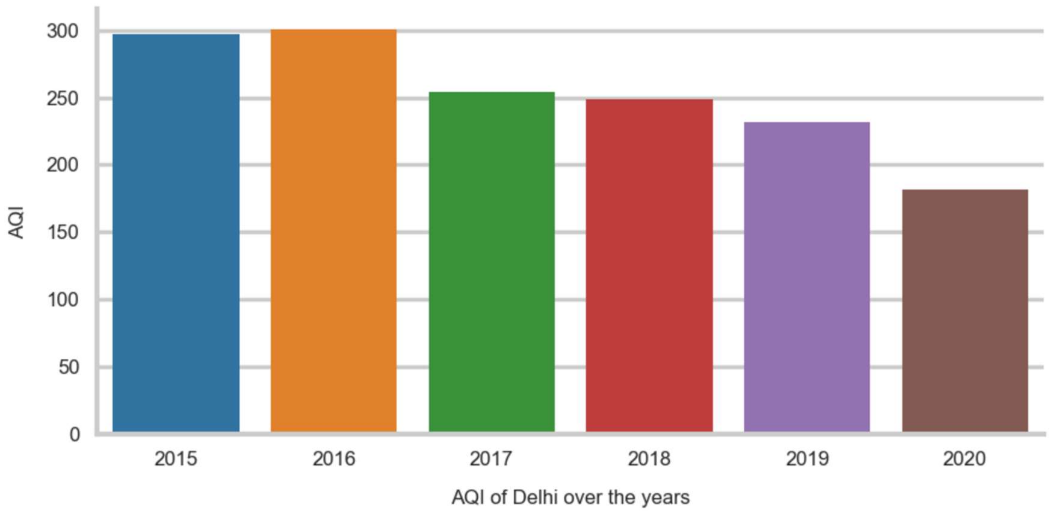


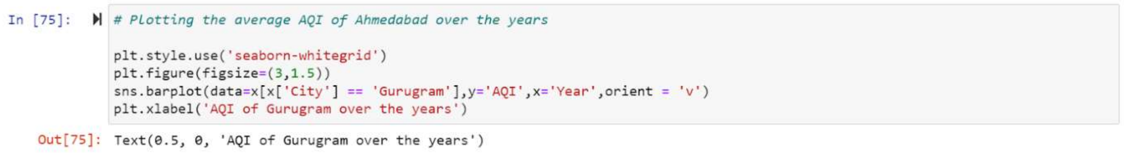
```
In [72]:  # Plotting the average AQI of Ahmedabad over the years

          plt.style.use('seaborn-whitegrid')
          plt.figure(figsize=(3,1.5))
          sns.barplot(data=x[x['City'] == 'Ahmedabad'],y='AQI',x='Year',orient = 'v')
          plt.xlabel('AQI of Ahmedabad over the years')

Out[72]: Text(0.5, 0, 'AQI of Ahmedabad over the years')
```



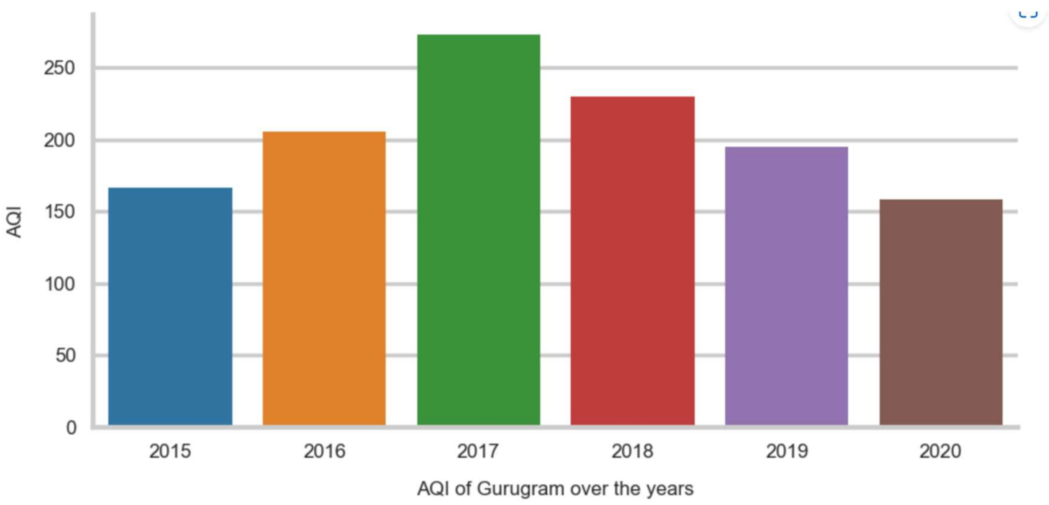


```
In [73]:  # Plotting the average AQI of Ahmedabad over the years

          plt.style.use('seaborn-whitegrid')
          plt.figure(figsize=(3,1.5))
          sns.barplot(data=x[x['City'] == 'Delhi'],y='AQI',x='Year',orient = 'v')
          plt.xlabel('AQI of Delhi over the years')

Out[73]: Text(0.5, 0, 'AQI of Delhi over the years')
```

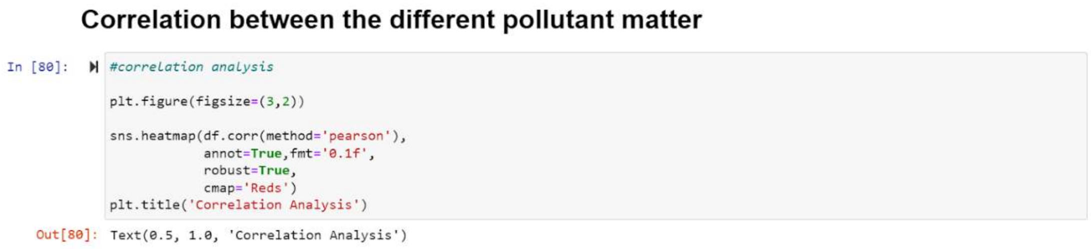




```
In [75]:  # Plotting the average AQI of Ahmedabad over the years

          plt.style.use('seaborn-whitegrid')
          plt.figure(figsize=(3,1.5))
          sns.barplot(data=x[x['City'] == 'Gurugram'],y='AQI',x='Year',orient = 'v')
          plt.xlabel('AQI of Gurugram over the years')

Out[75]: Text(0.5, 0, 'AQI of Gurugram over the years')
```

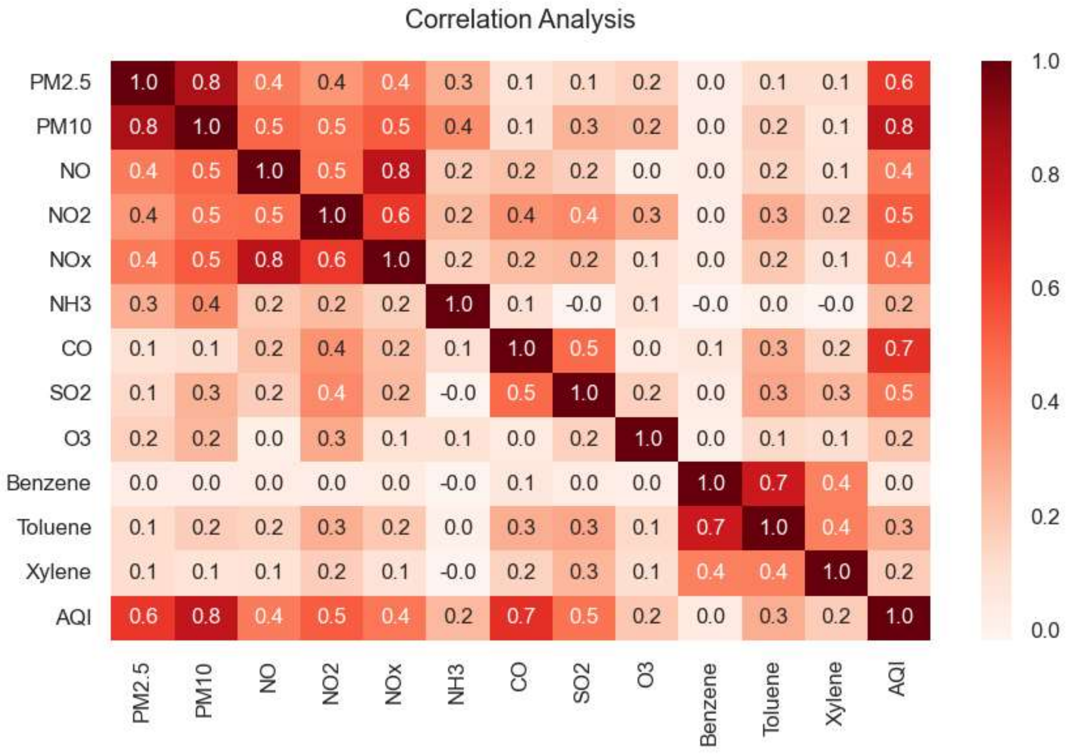


These show the AQI of key cities over the years. There has been a certain trend that can be observed where we see that cities like Lucknow, Gurugram, Patna, Ahmedabad have shown a constant AQI in the middle years after the starting growth but have recently shown a fall in the AQI level in the last 1-2 years.

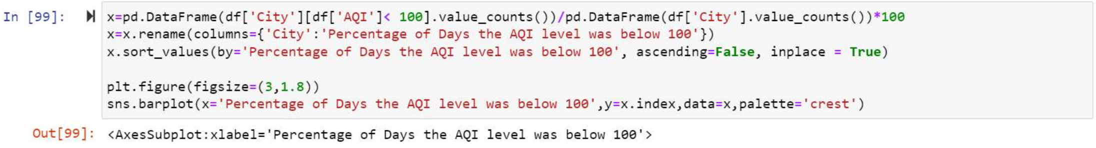


Correlation between the different pollutant matter

```
In [80]:  #correlation analysis

          plt.figure(figsize=(3,2))

          sns.heatmap(df.corr(method='pearson'),
                      annot=True,fmt='0.1f',
                      robust=True,
                      cmap='Reds')
          plt.title('Correlation Analysis')
Out[80]:  Text(0.5, 1.0, 'Correlation Analysis')
```



Cities that enjoy the most number of clean days

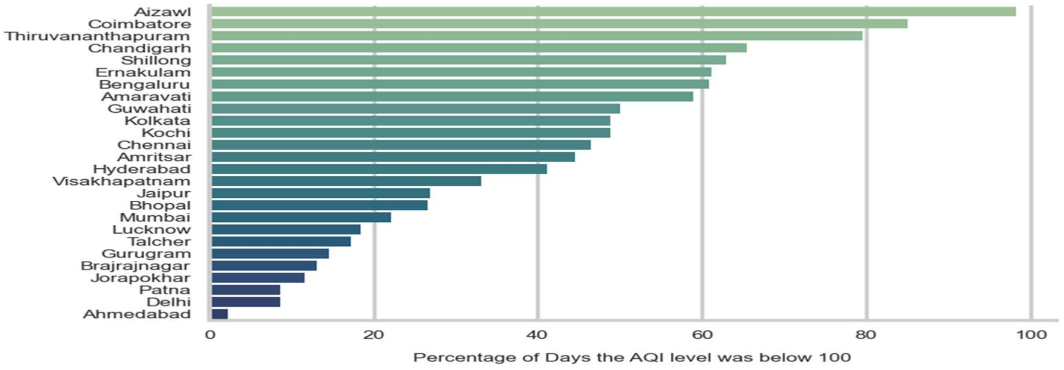


In [99]:

```
x=pd.DataFrame(df['City'][df['AQI']< 100].value_counts())/pd.DataFrame(df['City'].value_counts())*100
x=x.rename(columns={'City':'Percentage of Days the AQI level was below 100'})
x.sort_values(by='Percentage of Days the AQI level was below 100', ascending=False, inplace = True)

plt.figure(figsize=(3,1.8))
sns.barplot(x='Percentage of Days the AQI level was below 100',y=x.index,data=x,palette='crest')
```

Out[99]:

```
<AxesSubplot:xlabel='Percentage of Days the AQI level was below 100'>
```



It can be seen that, out of the 26 cities mentioned in the dataset, 11 cities enjoy at least a 'Satisfactory' level of Air Quality on almost 50% of the days. The top 5 cleanest cities would be Aizawl, Coimbatore, Thiruvananthapuram, Chandigarh, and Shillong.

Conclusions from the Data Analysis

The outcomes from the analysis is that we understand that the overall AQI index for the 26 cities in the data set show a similar trend on a lot of key points and hence similar steps can be taken for cities across the country. However, there are also certain key issues observed in certain cities for which special steps will have to be taken by the local governments of the place.