

AIR QUALITY ANALYSIS

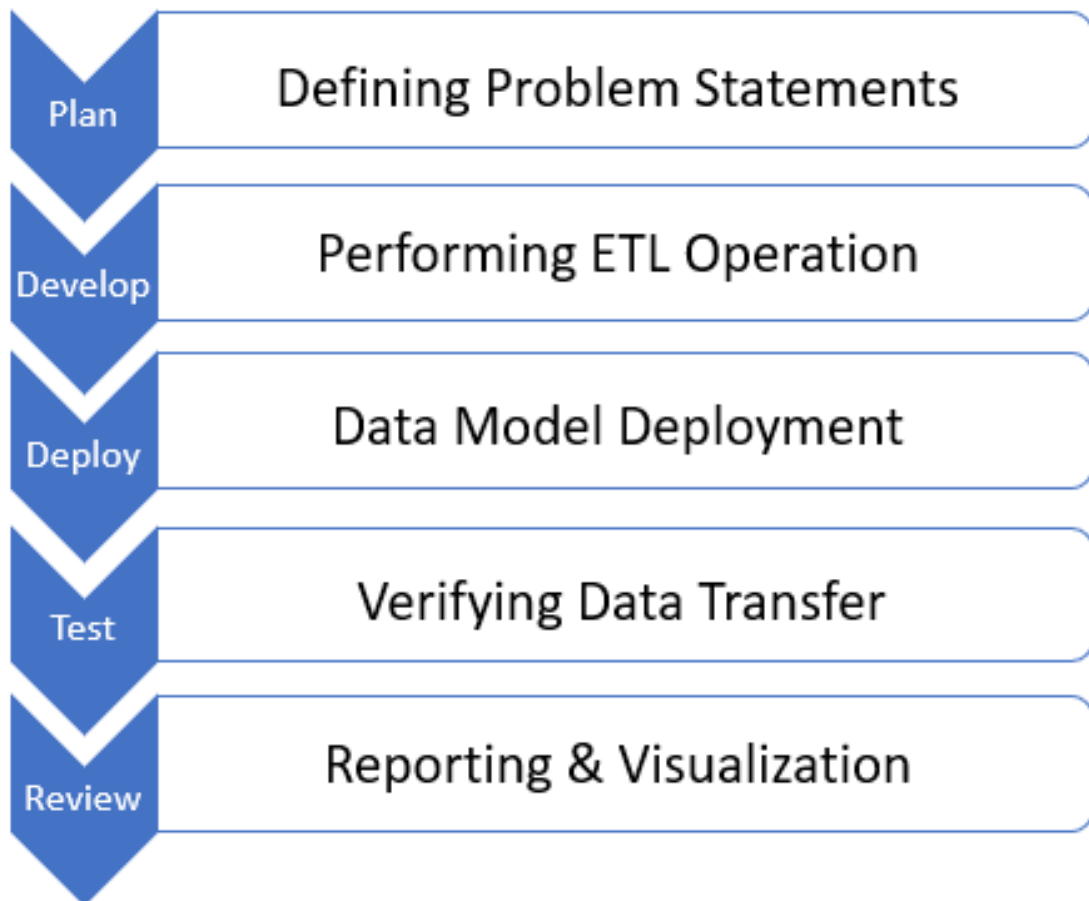
Submitted By:

Kshitiz Bansal

Problem Statements

1. Yearly & Monthly Air Quality Analysis
2. Detailed State & City Selective Air Quality Analysis
3. Impact Of Covid-19 On Air Quality
4. Observing Harmful Gases Content Variation
5. State Ranking On Yearly Basis
6. Analysing AQI Scale Over Period Of Years

Agile Methodology



Blueprint



Azure Data Factory

Extracting Data
From External Source
(HTTP)



Azure Databricks

Cleaning &
Transformation
Operations



Azure Analysis Services

Establishing Internal
Relationships & Data
Model Deployment



Power BI

Visualization &
Reporting

Calculation Metrics

Metrics & Dimensions

Geographical Location

Time Dimension

Parameters

Data Parameters

City

State

Datetime

PM10

PM2.5

NO

NO2

NOx

NH3

CO

SO2

O3

Benzene

Toluene

Xylene

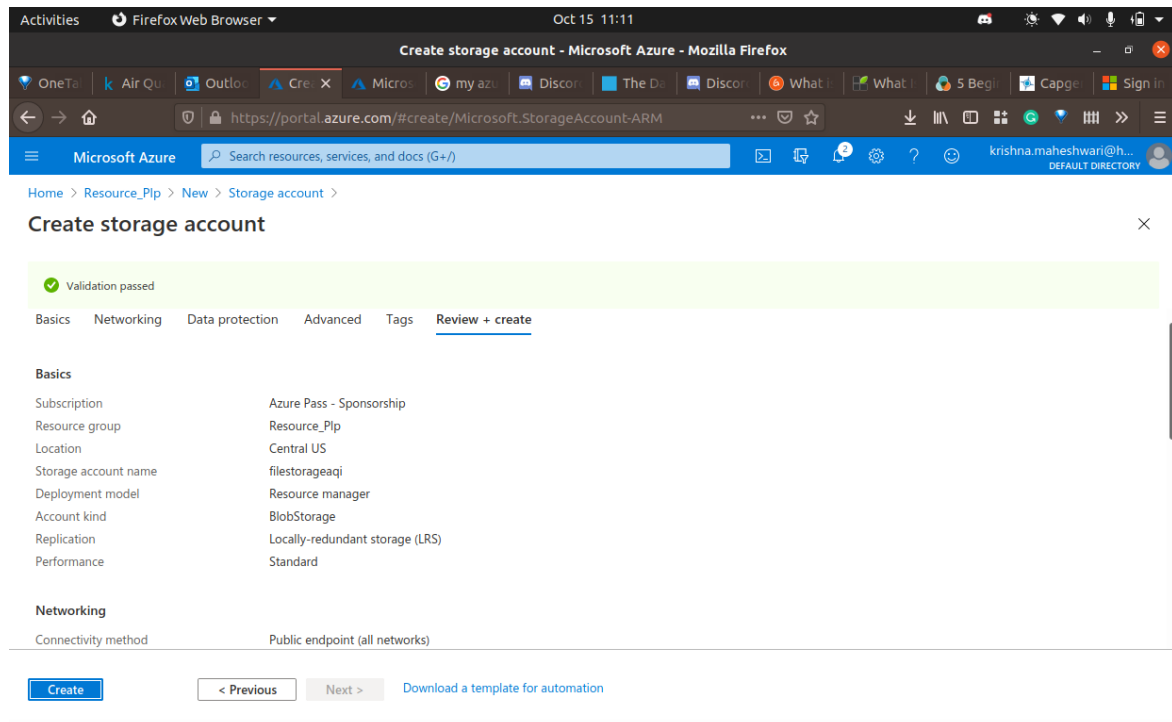
AQI

AQI Bucket

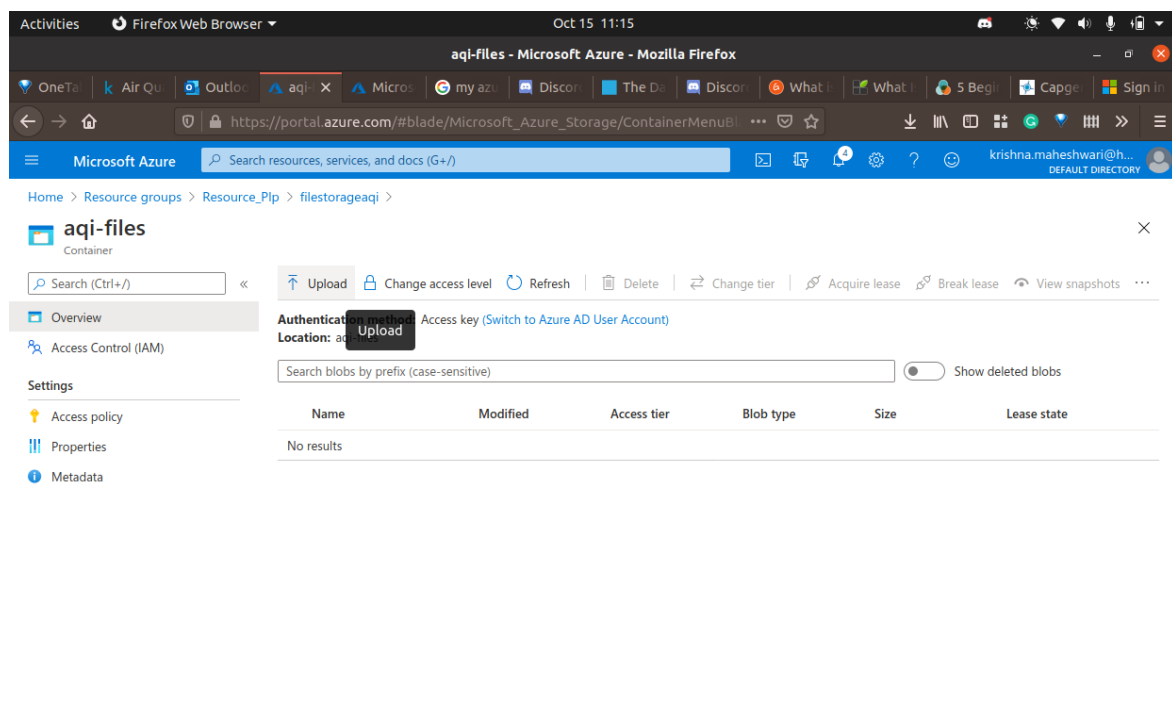
Extracting Data

1. Creating An Azure Data Factory Instance
2. Creating A Linked Service With HTTP Source
(Kaggle)
3. Saving Extracted Data To Blob Storage
4. Creating A Databricks Notebook Activity
5. Mounting Blob Storage With Databricks
Notebook
6. Reading Data Through Triggering Pipeline

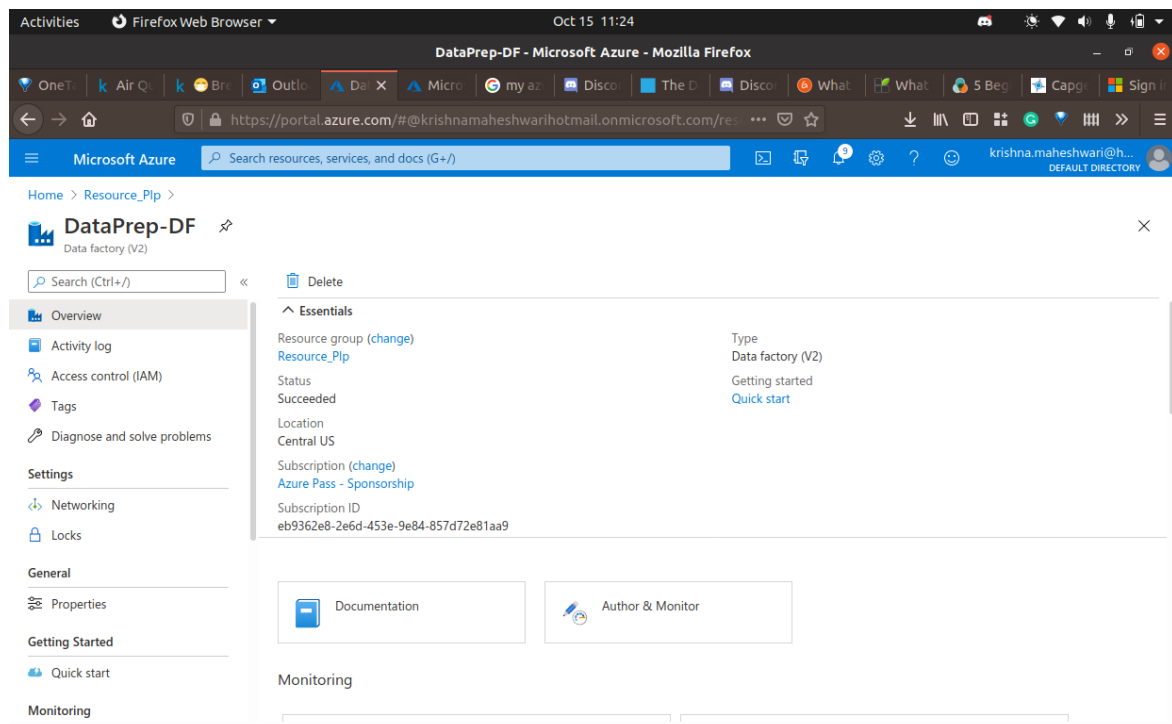
Creating Storage Account



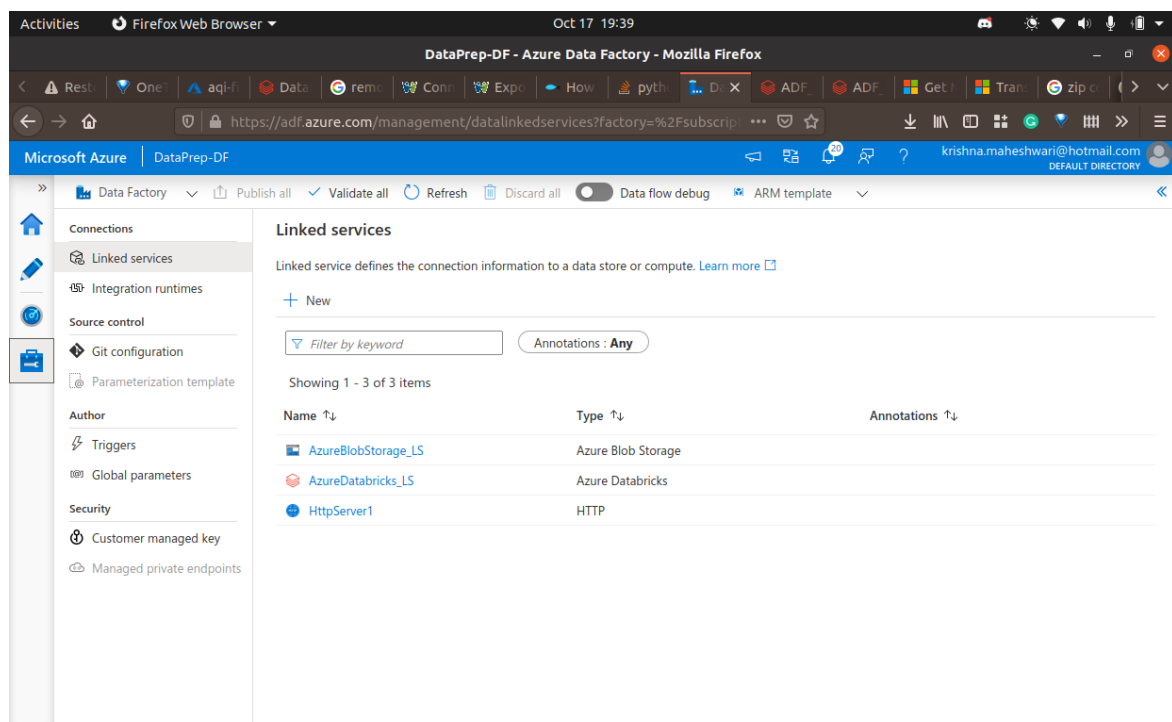
Creating Blob Container



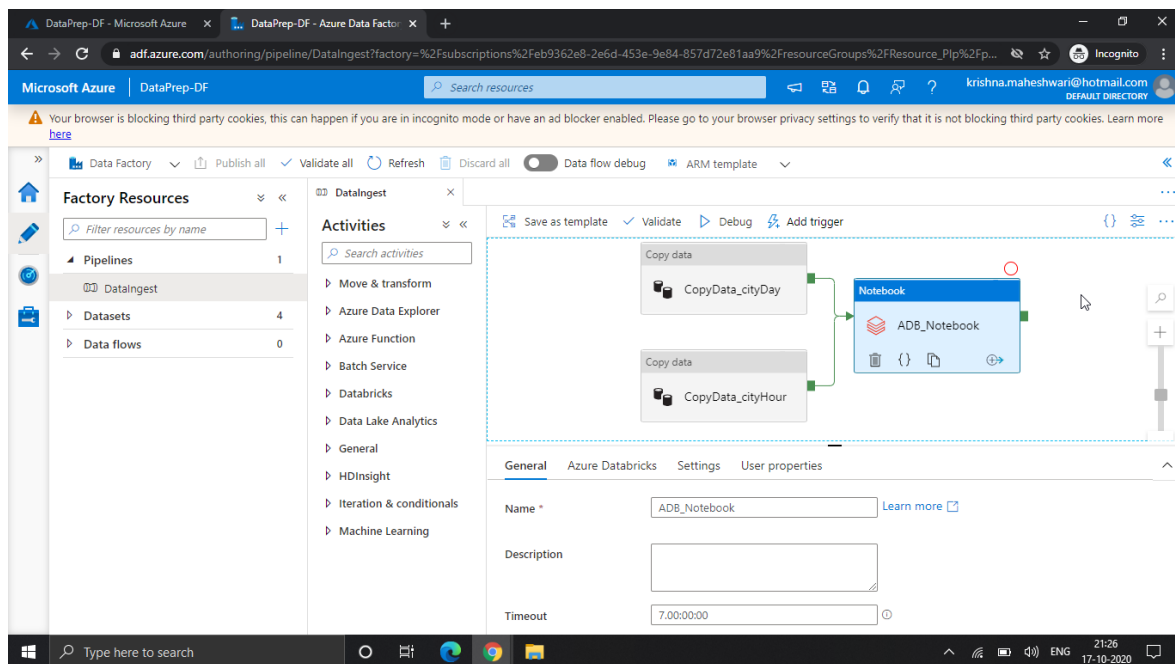
Creating Data Factory



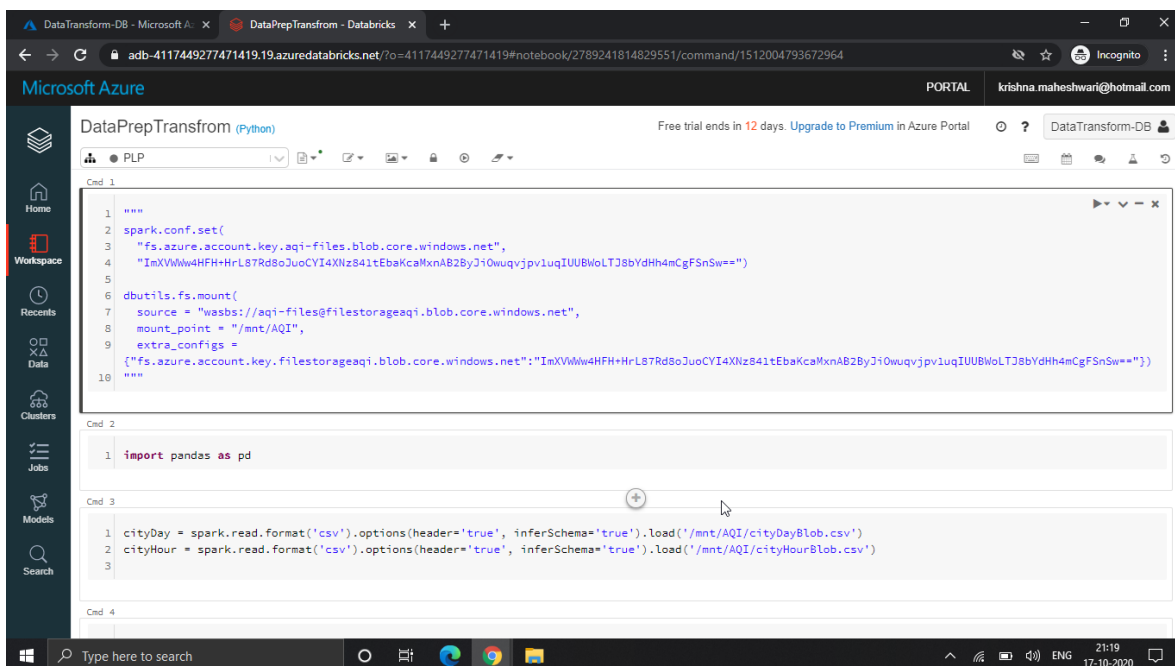
Creating HTTP Linked Service



Creating Databricks Notebook Activity



Mounting Blob Data To Databricks Notebook



Data Transformation

Language Used – Python 3

1. Detecting Schema For Each Dataframe
2. Using Linear Interpolation To Handle Missing Values
3. Creating Desired Calculated Columns
4. Mounting Transformed Data Back To Blob

Creating Azure Databricks Instance

The screenshot shows the 'Create an Azure Databricks workspace' page in the Microsoft Azure portal. The page is titled 'Create an Azure Databricks workspace' and has a breadcrumb trail: Home > Resource_Plp > New > Azure Databricks >. Below the title, there is a 'Project Details' section with the following fields:

- Subscription ***: Azure Pass - Sponsorship
- Resource group ***: Resource_Plp (with a 'Create new' link below it)

Below the 'Project Details' section, there is an 'Instance Details' section with the following fields:

- Workspace name ***: DataTransform-DB
- Region ***: Central US
- Pricing Tier ***: Trial (Premium - 14-Days Free DBUs)

At the bottom of the form, there are three buttons: 'Review + create' (in blue), '< Previous', and 'Next : Networking >'.

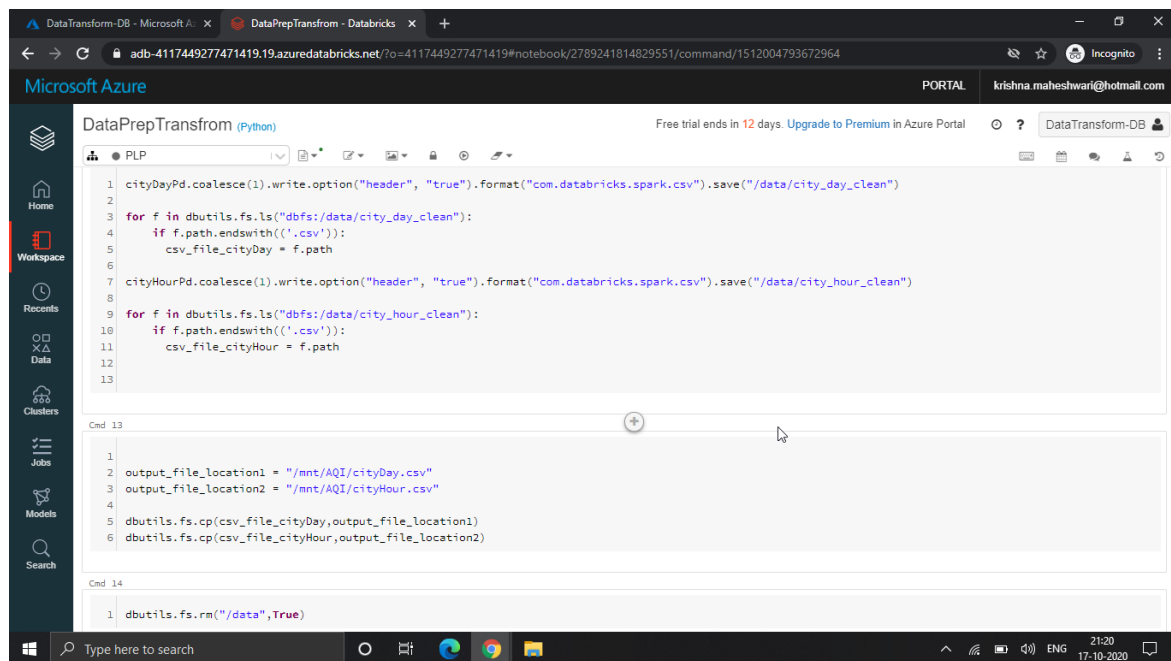
Generating Token

The screenshot shows the 'User Settings' page in the Azure Databricks portal. The page is titled 'User Settings' and has a breadcrumb trail: Home > Workspace > User Settings. The page is divided into two main sections: 'Access Tokens' and 'Git Integration'. The 'Access Tokens' section is active and shows a 'Generate New Token' button. A dialog box titled 'Generate New Token' is open, with the following fields:

- Comment**: Connecting Azure DataBricks Notebook with Data Factory
- Lifetime (days) ***: 90

At the bottom of the dialog box, there are two buttons: 'Cancel' and 'Generate'.

Saving Transformed Data To Blob



The screenshot shows the Microsoft Azure Data Prep Transform notebook interface. The notebook is titled "DataPrepTransform (Python)" and is running on a cluster named "PLP". The code is as follows:

```
1 cityDayPd.coalesce(1).write.option("header", "true").format("com.databricks.spark.csv").save("/data/city_day_clean")
2
3 for f in dbutils.fs.ls("/dbfs:/data/city_day_clean"):
4     if f.path.endswith('.csv'):
5         csv_file_cityDay = f.path
6
7 cityHourPd.coalesce(1).write.option("header", "true").format("com.databricks.spark.csv").save("/data/city_hour_clean")
8
9 for f in dbutils.fs.ls("/dbfs:/data/city_hour_clean"):
10     if f.path.endswith('.csv'):
11         csv_file_cityHour = f.path
12
13
```

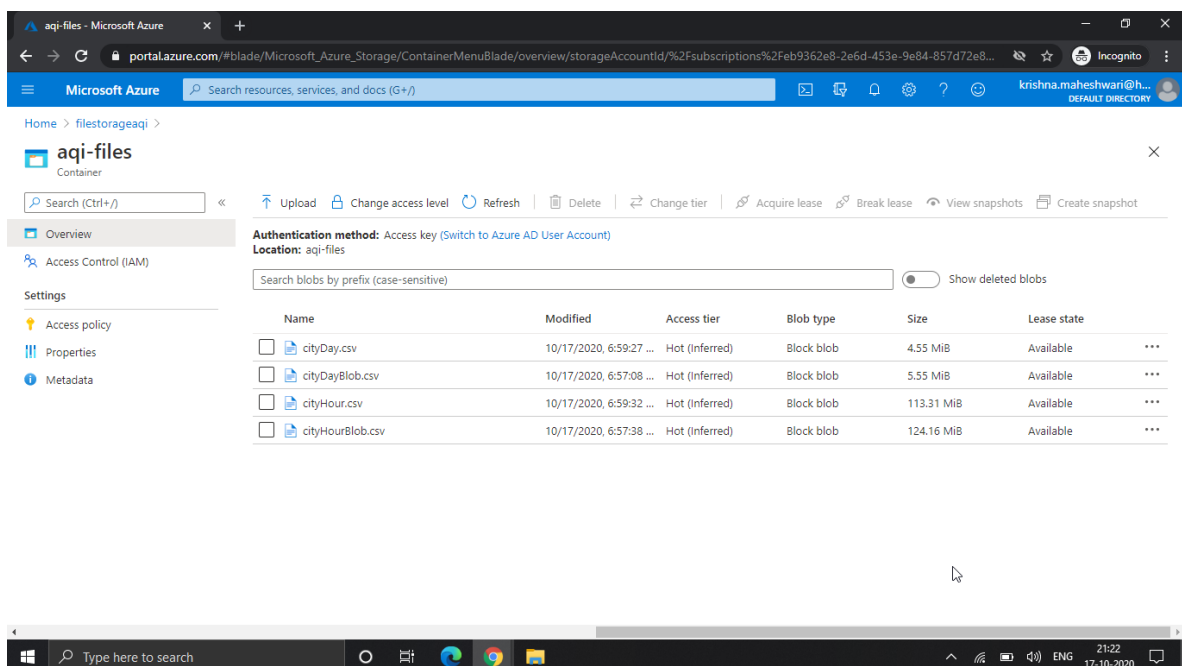
Below the code, there are two command boxes. Command 13 shows the output file locations and the copy command:

```
1
2 output_file_location1 = "/mnt/AQI/cityDay.csv"
3 output_file_location2 = "/mnt/AQI/cityHour.csv"
4
5 dbutils.fs.cp(csv_file_cityDay, output_file_location1)
6 dbutils.fs.cp(csv_file_cityHour, output_file_location2)
```

Command 14 shows the removal of the data directory:

```
1 dbutils.fs.rm("/data", True)
```

Verifying Data Transfer



The screenshot shows the Microsoft Azure portal interface for the 'aqi-files' container. The table below lists the blobs in the container:

Name	Modified	Access tier	Blob type	Size	Lease state
<input type="checkbox"/> cityDay.csv	10/17/2020, 6:59:27 ...	Hot (Inferred)	Block blob	4.55 MiB	Available
<input type="checkbox"/> cityDayBlob.csv	10/17/2020, 6:57:08 ...	Hot (Inferred)	Block blob	5.55 MiB	Available
<input type="checkbox"/> cityHour.csv	10/17/2020, 6:59:32 ...	Hot (Inferred)	Block blob	113.31 MiB	Available
<input type="checkbox"/> cityHourBlob.csv	10/17/2020, 6:57:38 ...	Hot (Inferred)	Block blob	124.16 MiB	Available

Data Model Deployment

Language Used – DAX

IDE Used – Visual Studio 2019

1.Importing Data Into Visual Studio Analysis

Services Project

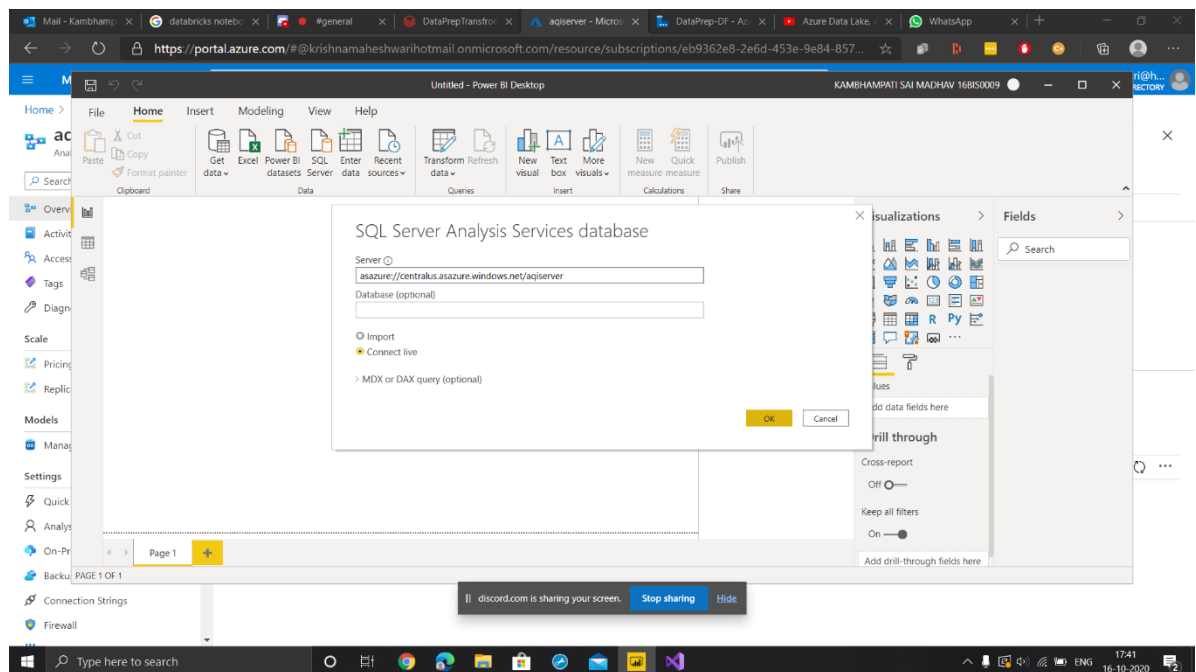
2.Establishing Internal Relationships Among

Tables

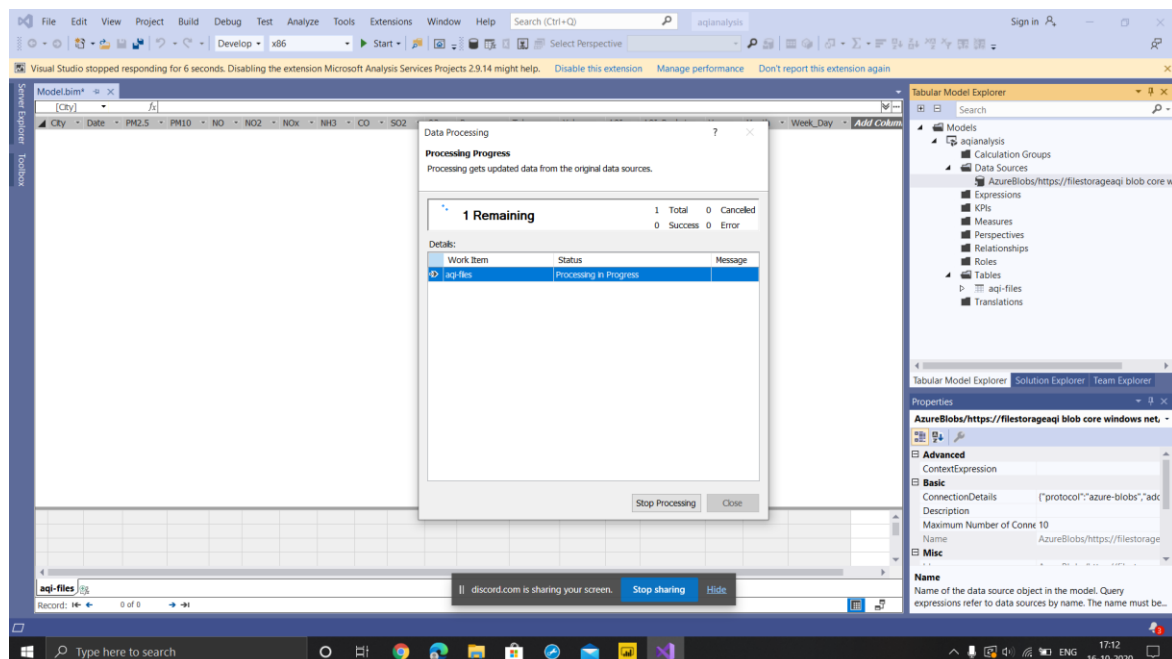
3.Deploying Data Model To Azure Analysis

Service Server

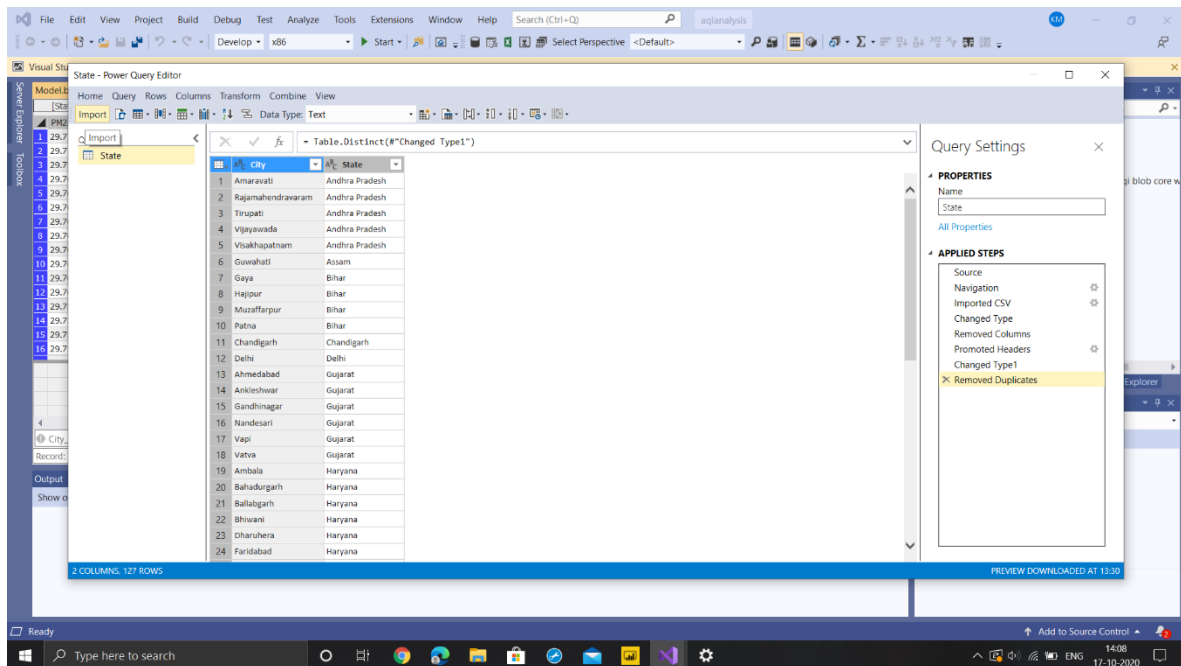
Connecting To Server



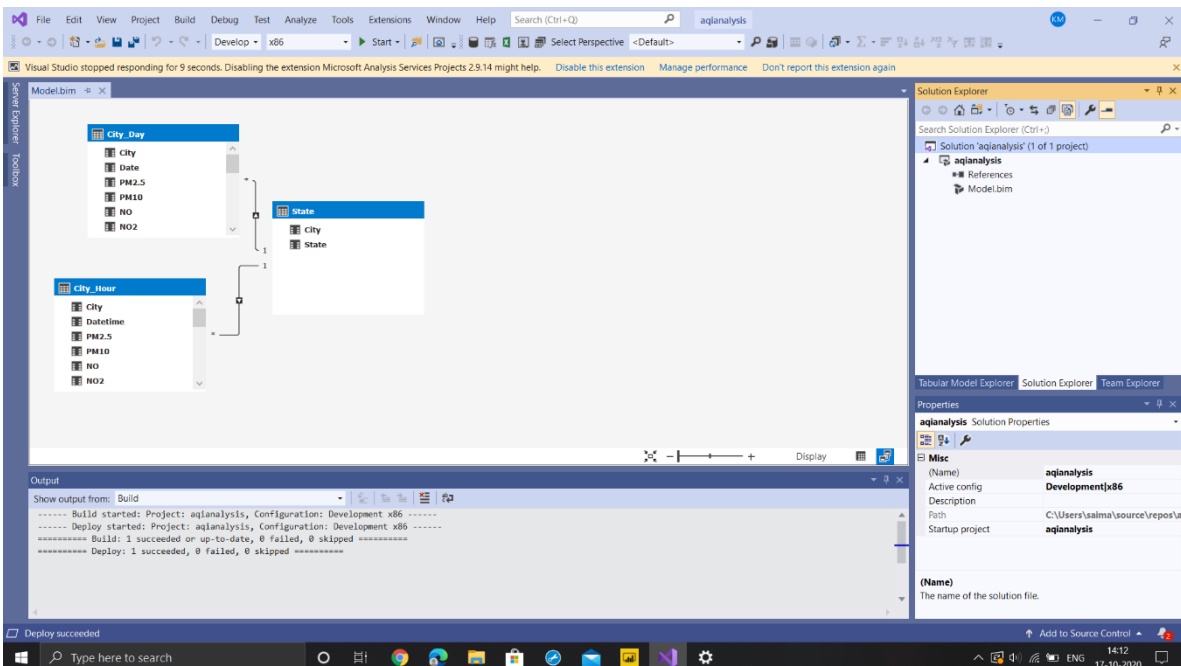
Connecting With Storage Account



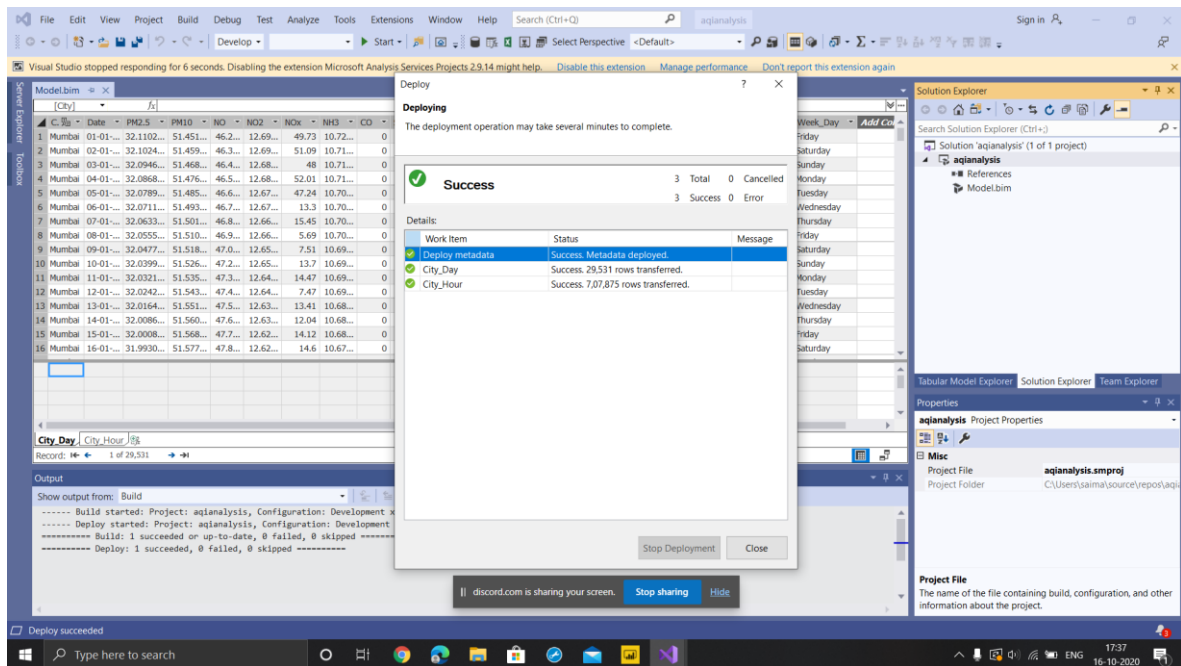
Fetching Data From Storage Account



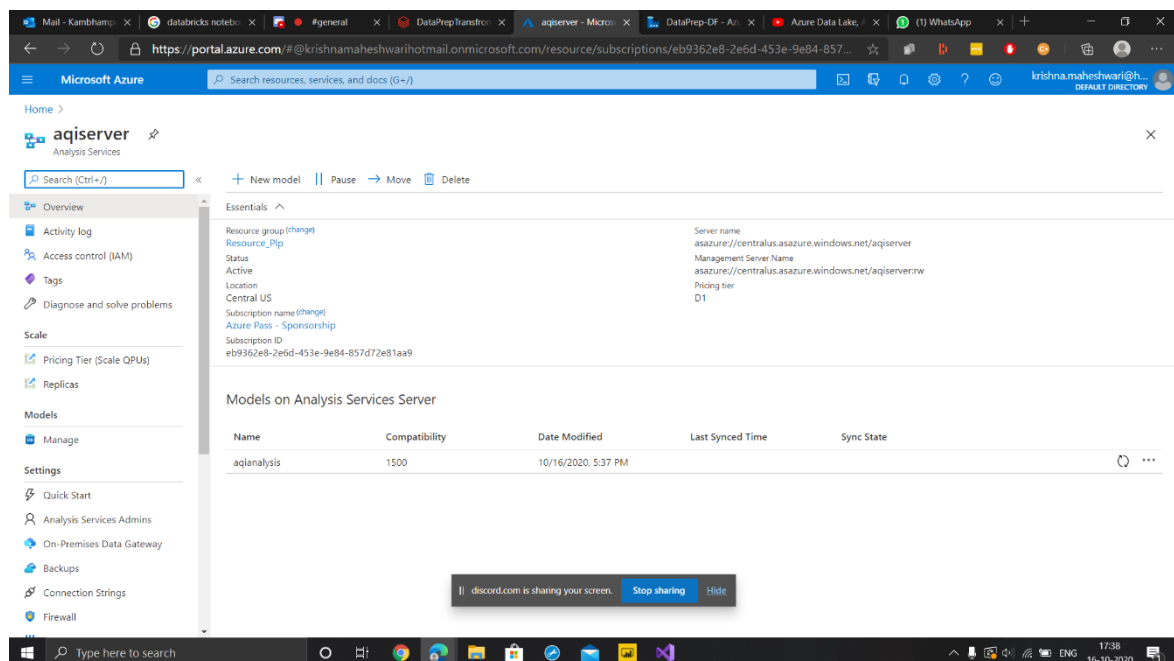
Establishing Internal Relationships Among Tables



Deploying Data Model In Server



Verification In Analysis Server



Data Analysis & Reporting

1.Connecting Power BI With Azure Analysis

Service Server

2.Fetching Data Model From Server & Loading

Data

3.Generating Reports Using Visualizations

4.Categorising Reports On Location & Time

Basis

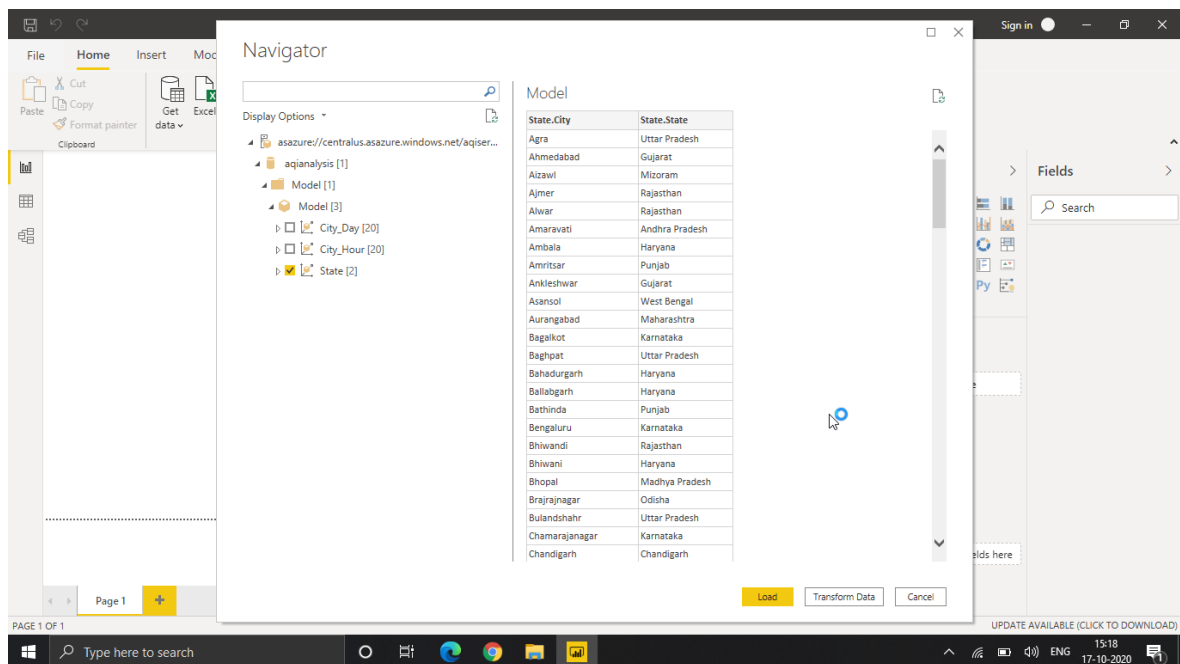
5.Covering Various Trends & Comparison

Scenarios

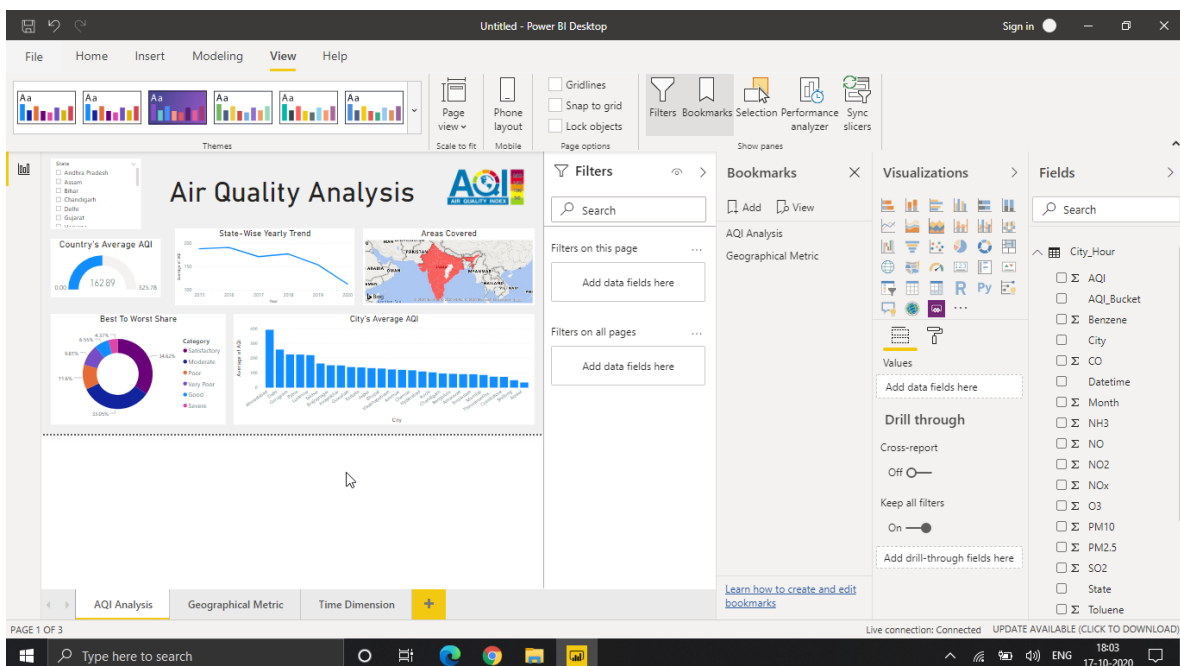
6.Creating Various Slicers With Handled

Interaction & Bookmarks

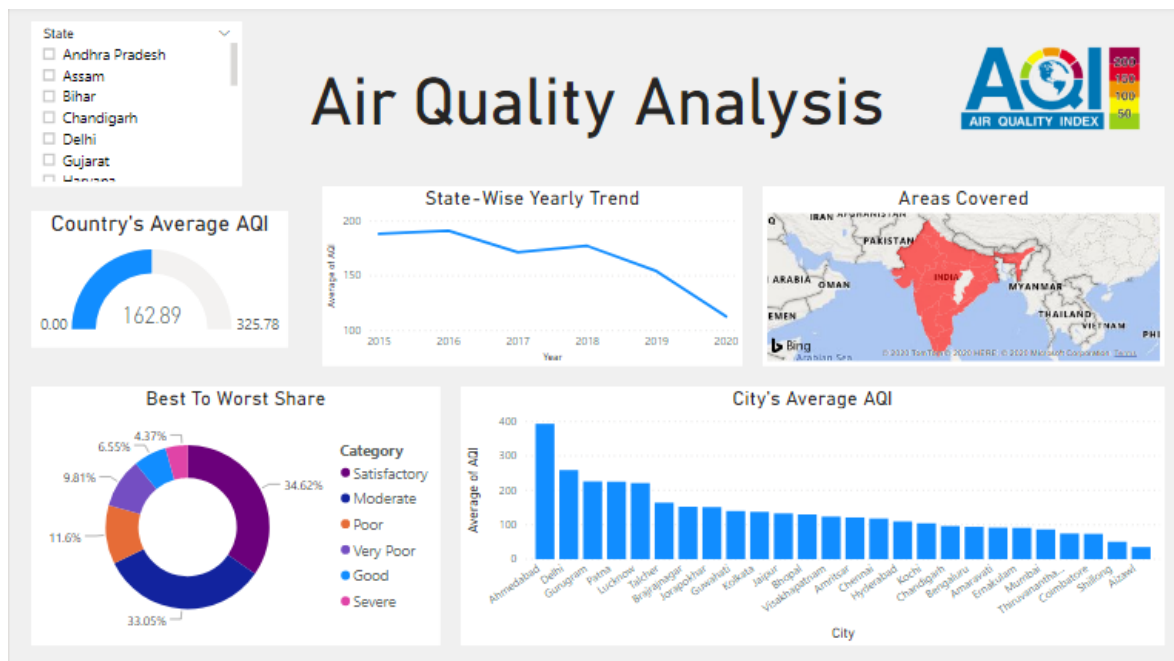
Connecting Power BI with Server



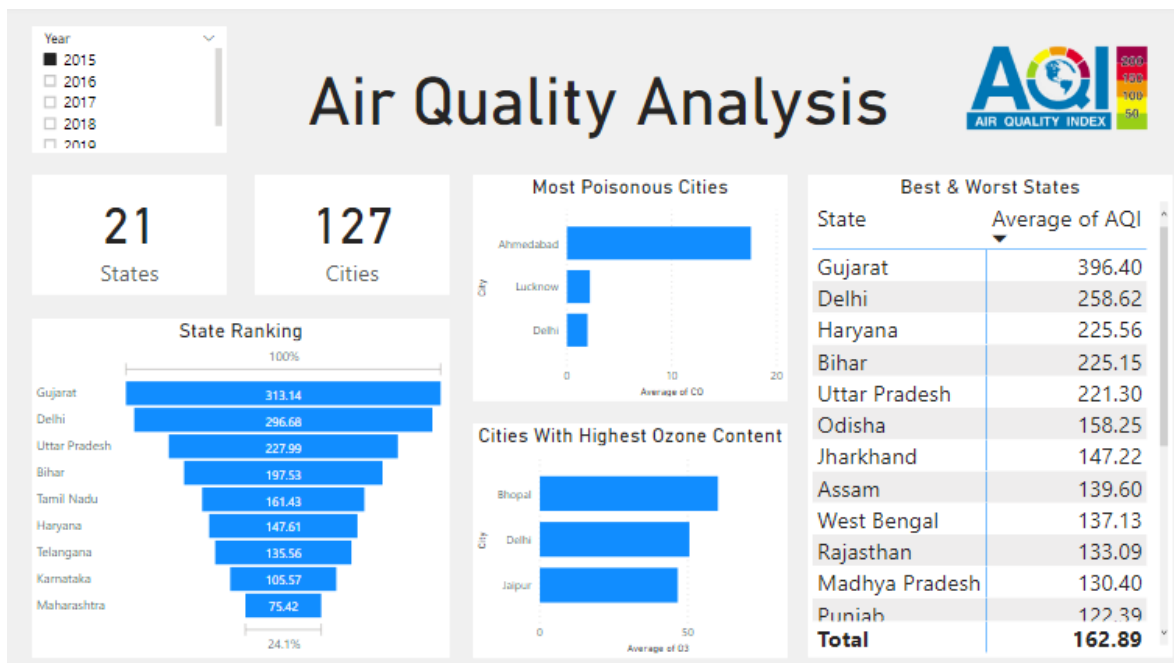
Creating Desired Slicers & Bookmarks



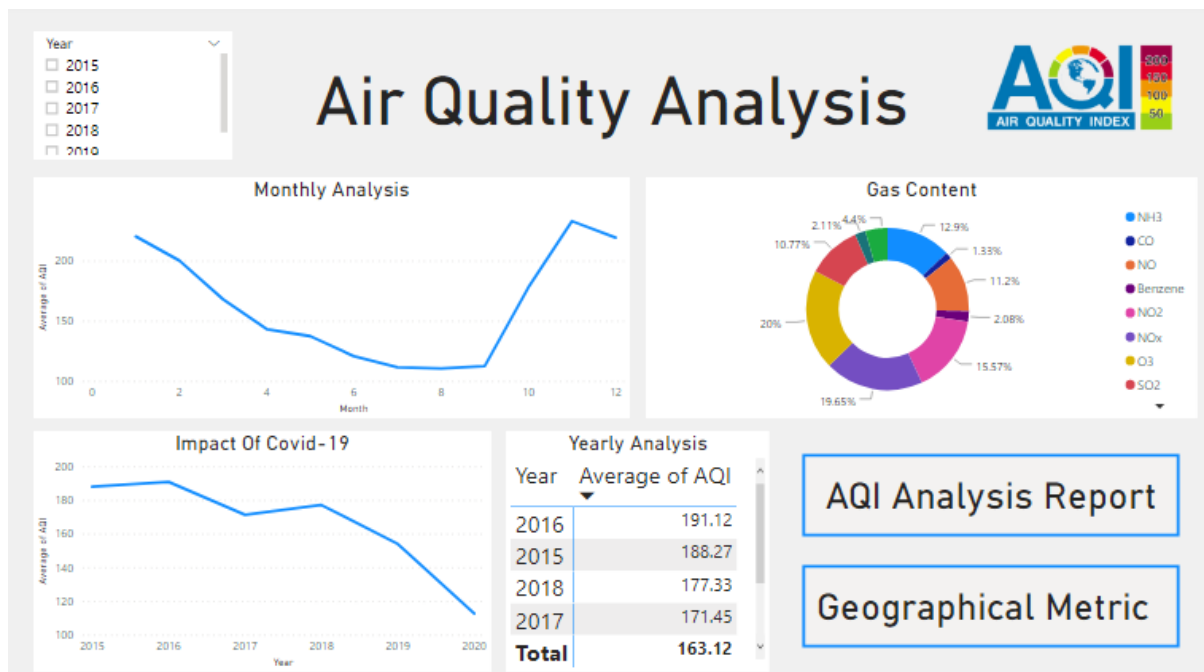
AQI Analysis



Geographical Metric



Time Dimension



- - End - -