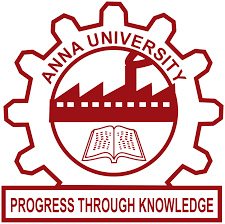
Anna University Regional Campus Coimbatore

Anna University: Chennai-600 025

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING



IBM Naan Mudhalvan Phase5 Submission

**Title**: **AIR QUALITY ANALYSIS AND PREDICTION IN TAMILNADU**

Name : M R ARTHI

Register Number: 710021106003

Department :ECE

Sem/year :5,3

Air Quality Analysis and Prediction in Tamil Nadu

**Objective:**

The objective of this project is to analyze and visualize air quality data from various monitoring stations in Tamil Nadu. The dataset contains measurement of Sulfur Dioxide (SO2), Nitrogen Dioxide (NO2), and Respirable Suspended Particulate Matter 10(RSPM/PM10) levels in different cities, towns, villages and areas. The project aims to gain insights into air pollution trends, identify areas with high pollution trends, identify areas with high pollution levels and cerate a predictive model to estimate RSPM/PM10 levels based on S02 and NO2 levels.

**Abstract:**

An index which is used to report air quality is called the air quality index (AQI). It measures the impact of air pollution on a person’s health over a short period of time. The purpose of the AQI is to educate the public on the negative health effects of local air pollution. Air Pollution implies a great significant on environmental and health challenge and it demands on comprehensive and prediction efforts. This project,**” Air Quality Analysis and Prediction in Tamil Nadu,”** focuses on leveraging data science techniques to access and predict the air quality levels across various regions in Tamil Nadu, India

**Data set description and Sample Data:**

The link to the dataset for this chosen project is given below:

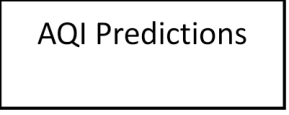
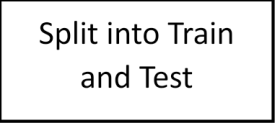
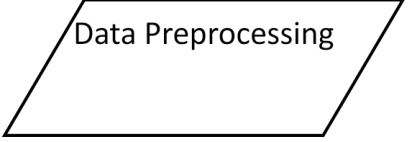
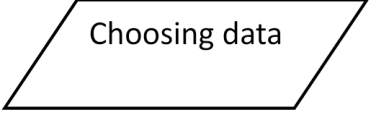
[https://tn.data.gov.in/resource/location-wise-daily-ambient-air-quality-tamil-nadu-year-2014](https://tn.data.gov.in/resource/location-wise-daily-ambient-air-quality-tamil-nadu-year-2014indias-state-wise-air-quality-data-tamil-nadu)

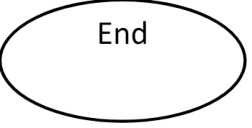
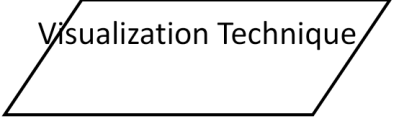
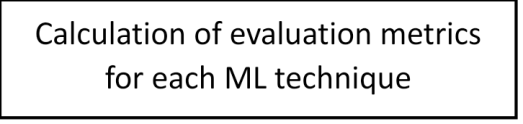
The above dataset contains the combined version of air quality of Tamil Nadu from 2014. This contains some district wise data for the prediction of air quality parameter in the state of Tamil Nadu. This data was released by the Ministry of Environment and Forests and Central Pollution Control Board of India under the National Data Sharing and Accessibility Policy (NDSAP).

**Methodology:**

Flow Chart for the proposed system





**STEP1: Choosing a dataset**

Choosing the proper dataset for implementing the project

**STEP2: Data Pre-processing**

In data pre-processing we have selected data for the analysis of air quality in the various district of Tamil Nadu. Each of the dataset was cleaned by remove null values of the chosen dataset. Microsoft Excel Software was used to remove unnecessary, irrelevant and erroneous data.

**STEP3: Splitting of the dataset**

The chosen datasets are split into training and test data. These are used to train the model and then test it against the original data. The values predicted by the machine learning algorithm and to predict accuracy of the data.

**STEP4: Training the dataset**

Empirical studies show the best results which are obtained if 80% of the data is used for training. Random sampling is used as a way to divide the data into train and test sections. It is widely accepted and is very popular.

**STEP5: Testing the dataset**

Empirical studies show the best results that are obtained if the remaining 20% of the data is used for testing. Random sampling is used as a way to divide the data into train and test sections. It is widely accepted and is very popular.

**STEP6: Feature Scaling**

The data should be normalized in order to make the dataset more flexible and more consistent. Standard Scalar from Scikit-Learn Library has been used to do so. It normalizes the features by deleting the mean and scaling the unit variance

**STEP7: Applying various Machine learning techniques**

After the normalization, we need to apply the various machine learning technique for analysing the data. Some of the machine learning technique random forest regression, support vector regression which are used to analysis the air quality index.

STEP8:**Applying ML technique-random forest regression**  
Random forest is a supervised machine learning algorithm that is used for classification and regression problems. It creates decision trees from several samples, using the majority vote for classification and the average in the case of regression. A random forest produces precise predictions that are easy to understand. Effective handling of large datasets is possible.

STEP 9:**Calculation of evaluation metric for each ML techniques**  
The metrics used for the proposed work are R-SQUARE, MSE, RMSE, MAE, and the accuracy of various algorithm.

**STEP 10: Determine the efficient Visualization techniques**

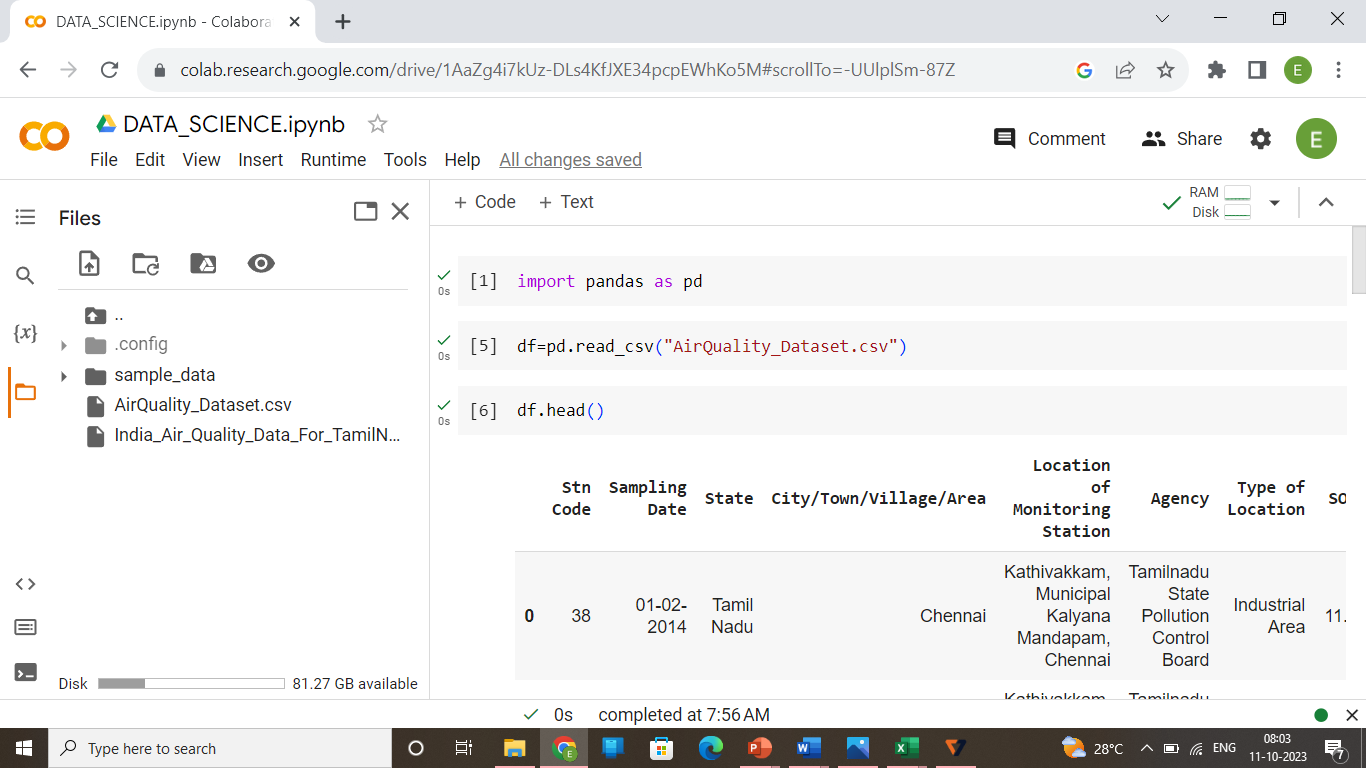
Visualizations play a crucial role in conveying insights from air quality data analysis. Here are some visualization methods and techniques that can be employed in the **“Air Quality Analysis and Prediction in Tamil Nadu”**

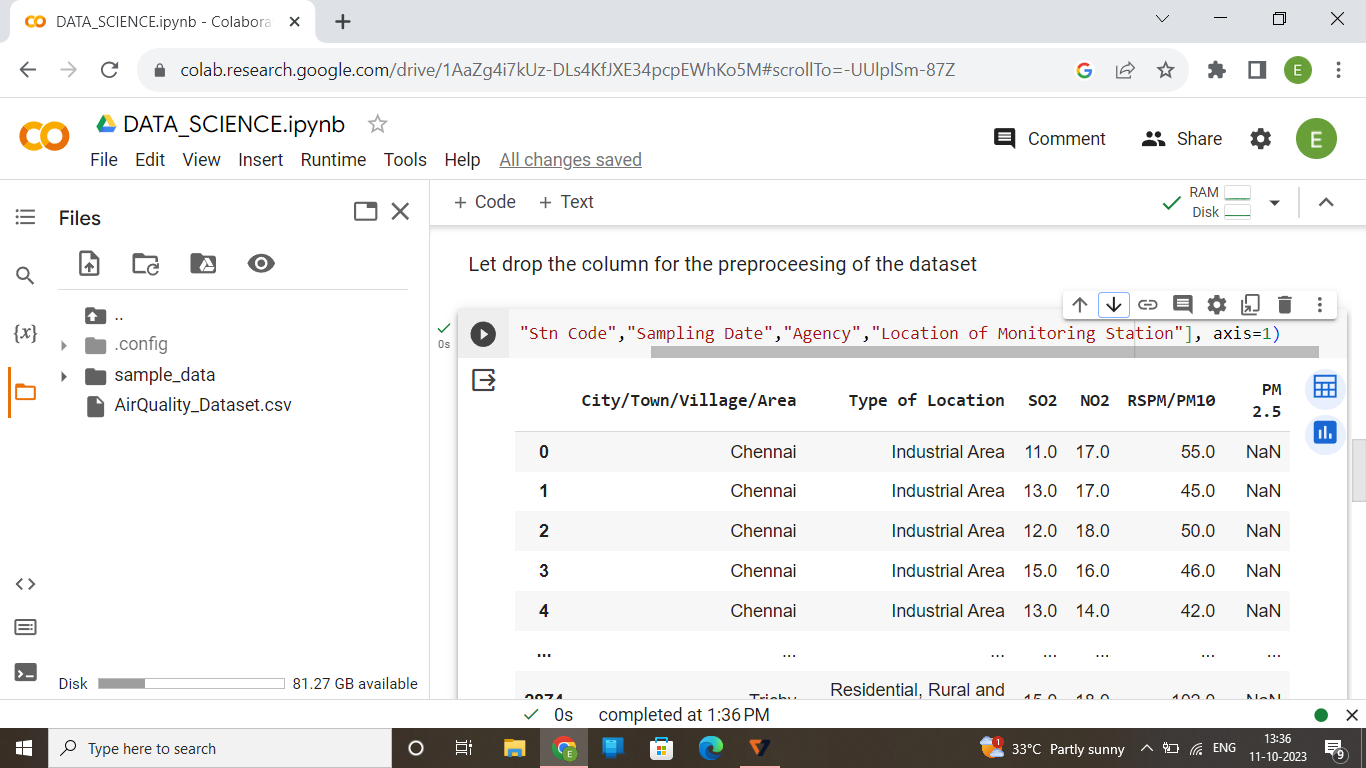
* **Time Series Plots**- Plot historical trends of SO2, NO2 and RSPM/PM10 levels over time. Use **line charts** to illustrate daily, monthly or seasonal variations
* **Heatmaps**-Create heatmaps to visualize pollutant concentrations across different monitoring stations and geographical areas.
* **Scatter Plots**-Use scatter plots to explore correlations between air quality parameters.

Explanation:

The given dataset contains the different columns with their specific details. The different columns are **“stn code, sampling date, state, city/town, Location of monitoring stations, Agency, Type of location, SO2, NO2, RSPM/PM, SPM”**

To further proceeding of the project let us drop the unwanted columns that is unnecessary for analysis of the air quality in Tamil Nadu





By this, we are going to use the specific columns for the preprocessing of the data to predict and analyze the air quality in various regions in Tamil Nadu.

**1.Data Collection:**

Monitoring Stations: Establish a network of air quality monitoring stations across Tamil Nadu. These stations should be strategically located in urban, industrial, and rural areas to capture a representative sample of air quality conditions.

* **Parameters:** Measure various air quality parameters, including particulate matter (PM 2.5 and PM 10), nitrogen dioxide(NO2), sulphur dioxide(SO2), carbon monoxide(CO), ozone(O3) and other volatile organic compounds.
* **Meteorological Data:** Collect meteorological data, such as temperature, humidity, wind speed,

and wind direction, as these factors can influence air quality.

* **Historical Data:** Gather historical air quality data to establish trends and identify areas with chronic air quality problems.

**2. Data Analysis:**

**Air Quality Index (AQI):** Calculate the AQI for different locations in Tamil Nadu to provide a clear and understandable representation of air quality to the public.

* **Identify Hotspots:** Identify areas with consistently poor air quality, such as major cities or industrial zones, and pinpoint the key pollutants responsible.
* **Seasonal Trends:** Analyse seasonal variations in air quality, as well as the factors contributing to these variations, such as agricultural burning, weather conditions, or industrial activity.

**3. Pollution Sources:**

**Industrial Emissions:** Emissions from industrial facilities, such as factories and power plants, and assess compliance with emission standards. Examine

* **Vehicle Emissions:** Evaluate the impact of vehicular emissions on air quality, considering the prevalence of different types of vehicles and fuel types.
* **Agricultural Practices:** Investigate the role of agriculture in air quality, including the use of pesticides and burning of crop residues.
* **Waste Management:** Assess waste disposal practices and their impact on air quality, especially in urban areas.

**4. Health Impact Assessment:**

Collaborate with healthcare institutions to study the health effects of poor air quality on the population of Tamil Nadu. Identify vulnerable groups, such as children, the elderly, and individuals with pre-existing respiratory conditions, and assess their exposure and health outcomes.

**5. Policy and Regulation:**

Review existing air quality regulations and policies in Tamil Nadu to identify gaps or areas for improvement. Develop or update regulations to control emissions from various sources, and enforce strict compliance measures.

**6. Public Awareness:**

Launch public awareness campaigns to educate residents about the health risks associated with poor air quality and ways to protect themselves. Provide real-time air quality information through websites, apps, and public displays.

**7. Mitigation Strategies:**

Implement pollution control technologies in industries and encourage the use of cleaner fuels. Promote sustainable urban planning, public transportation, and green spaces to reduce vehicle emissions and enhance air quality. Encourage agricultural practices that minimize burning and promote sustainable waste management.

**8. International Cooperation:**

Collaborate with neighbouring states and countries to address transboundary air pollution issues, especially during cross-border events like crop burning. This air quality analysis is the first part of a comprehensive strategy to improve air quality in Tamil Nadu. It is essential to monitor progress over time and adjust strategies as needed to ensure cleaner air for the people and the environment.

**PROGRAM:**

**Import the necessary libraries:**

import pandas as pd

import scipy

import numpy as np

import matplotlib.pyplot as plt

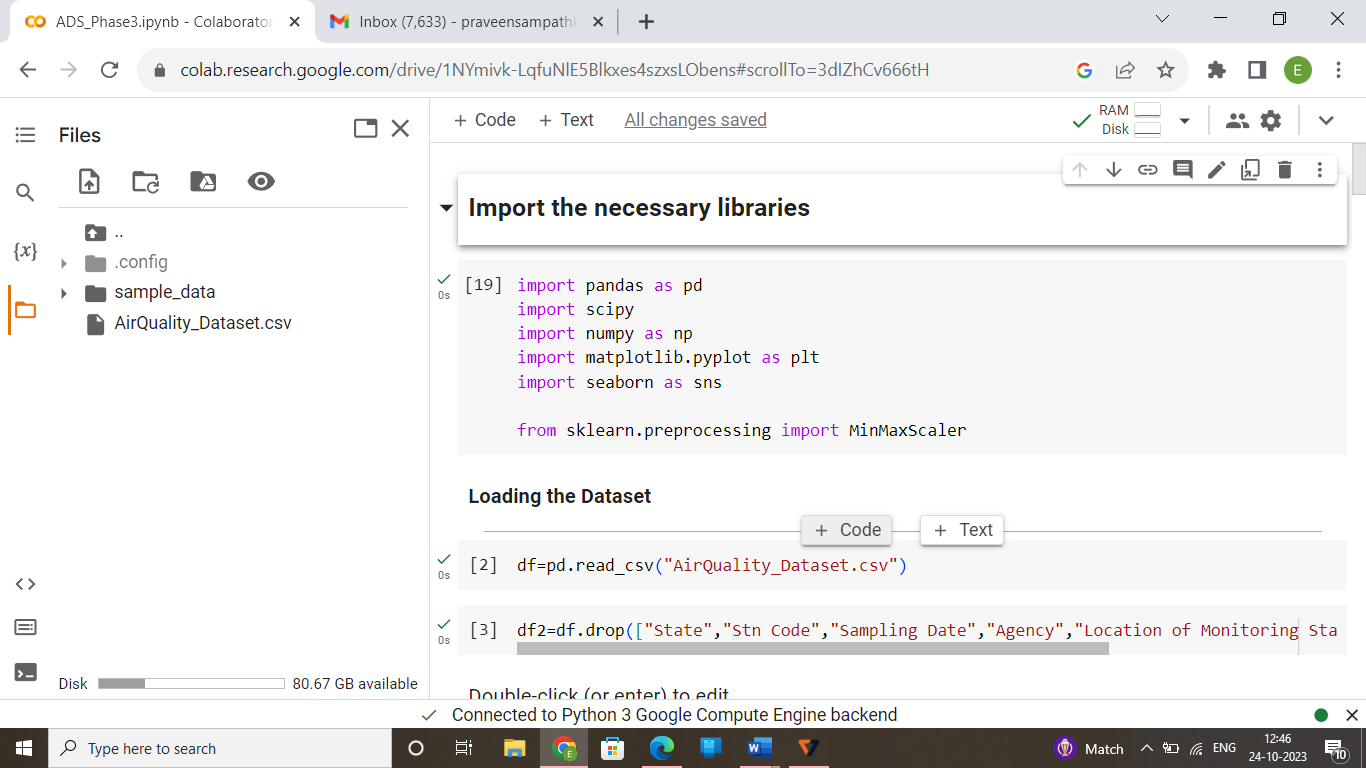
import seaborn as sns

from sklearn.preprocessing import MinMaxScaler

**Loading the Dataset:**

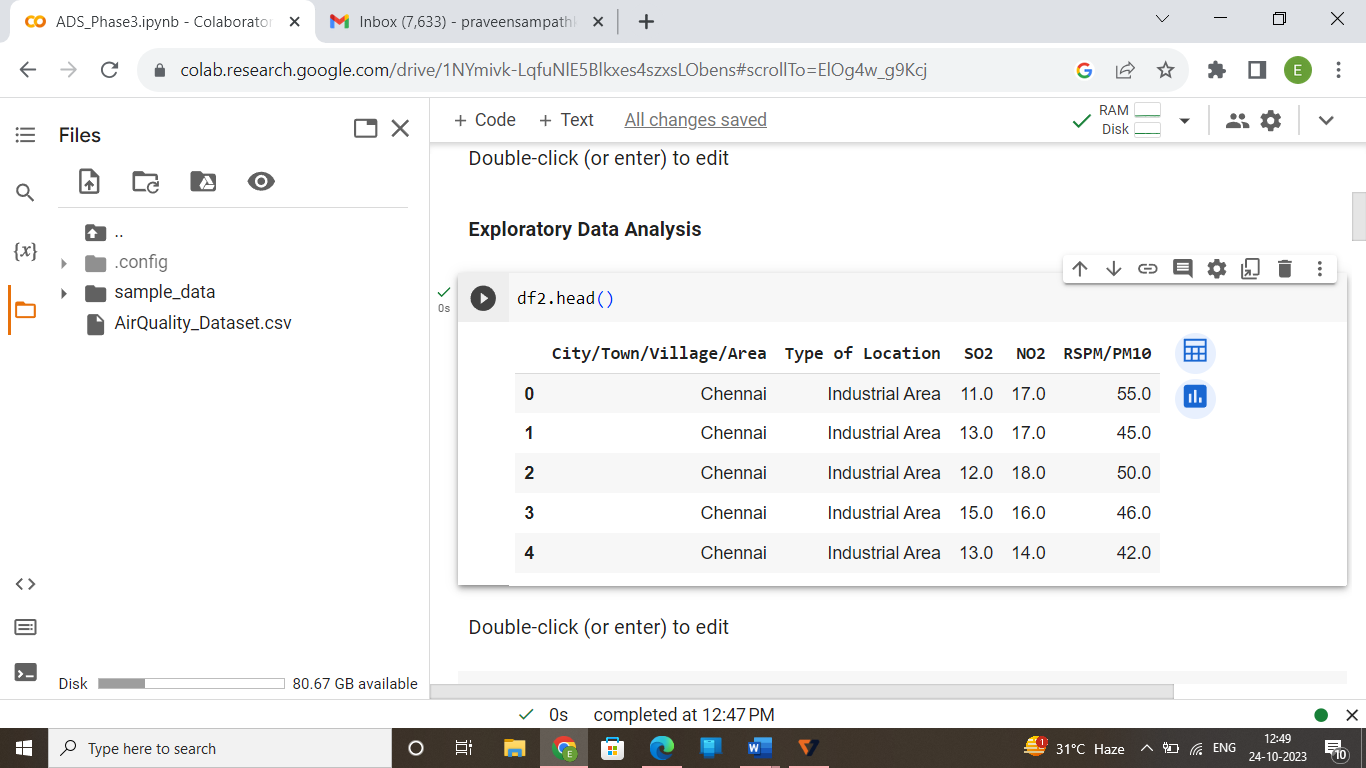
df=pd.read\_csv("AirQuality\_Dataset.csv")

df2=df.drop(["State","Stn Code","Sampling Date","Agency","Location of Monitoring Station","PM 2.5"], axis=1)

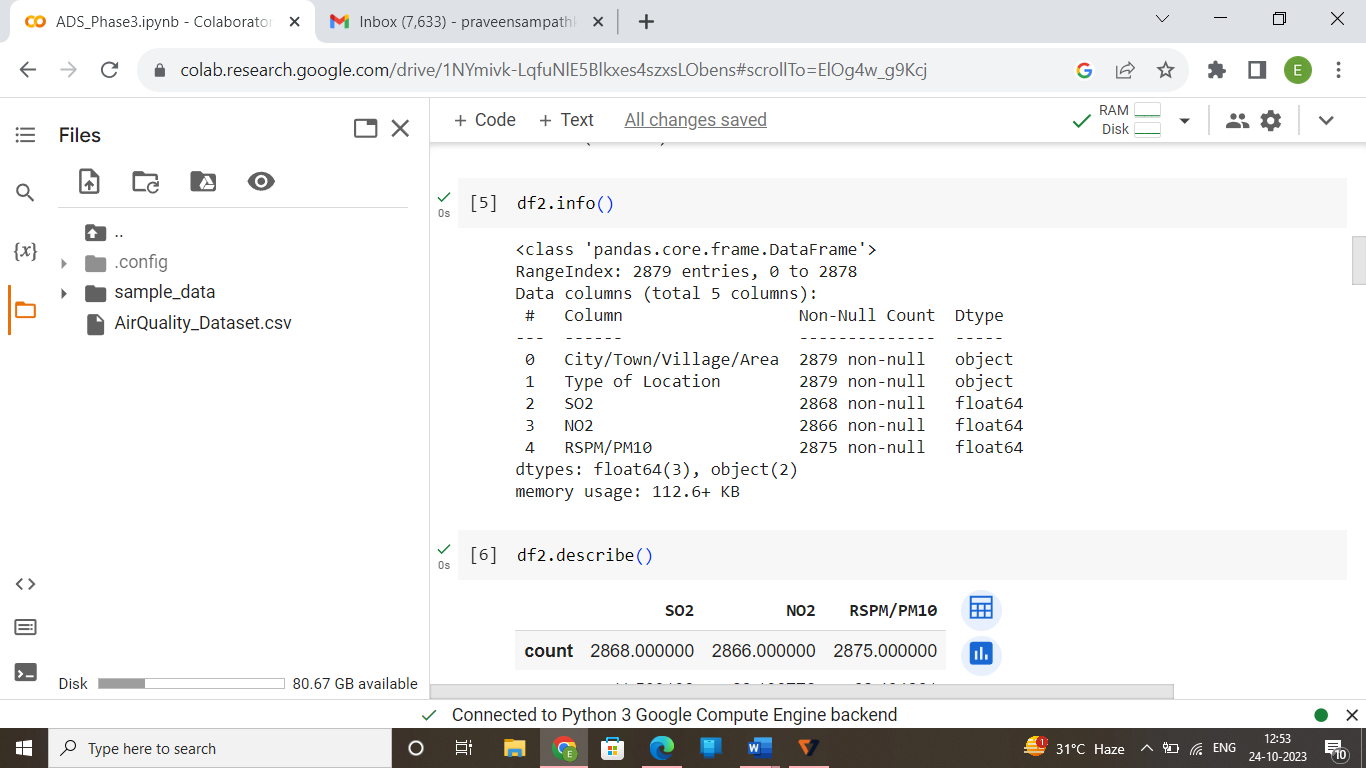


**Exploratory Data Analysis:**

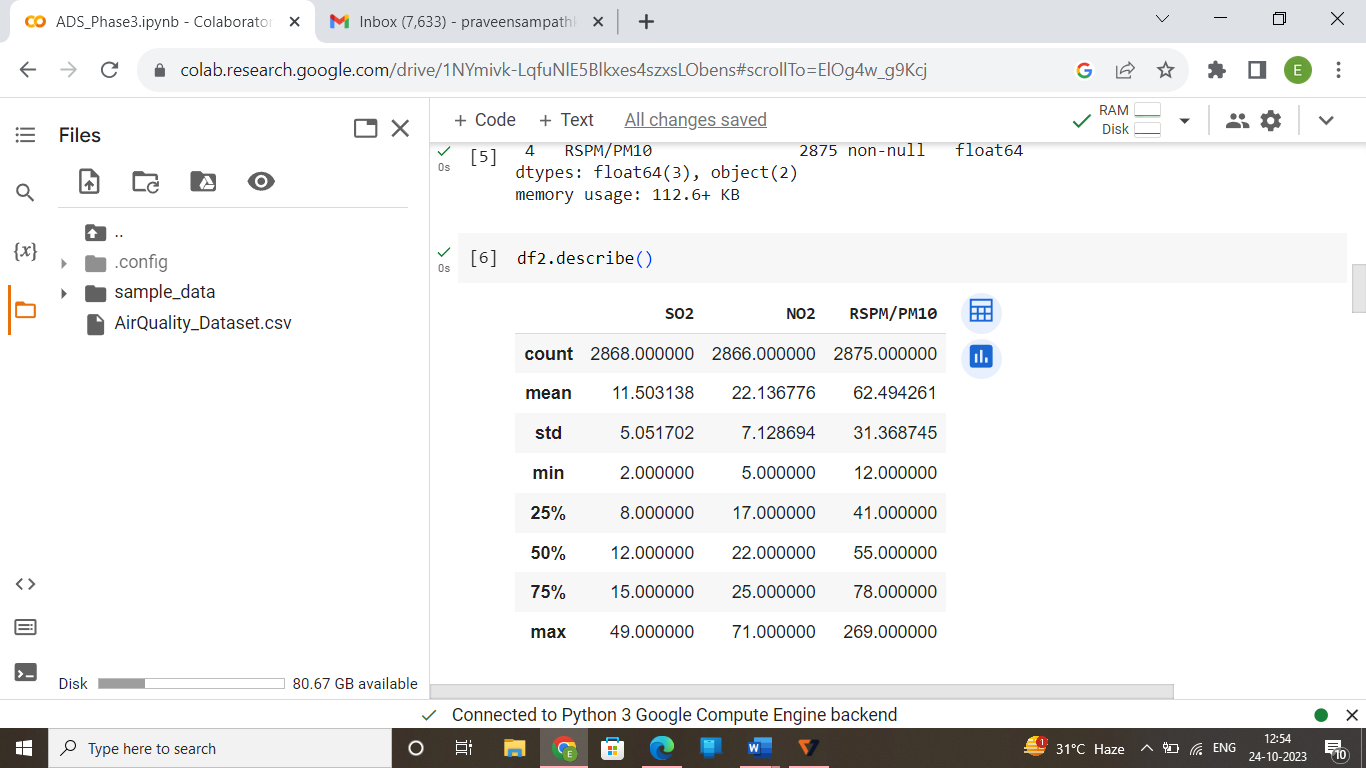
df2.head()



df2.info()

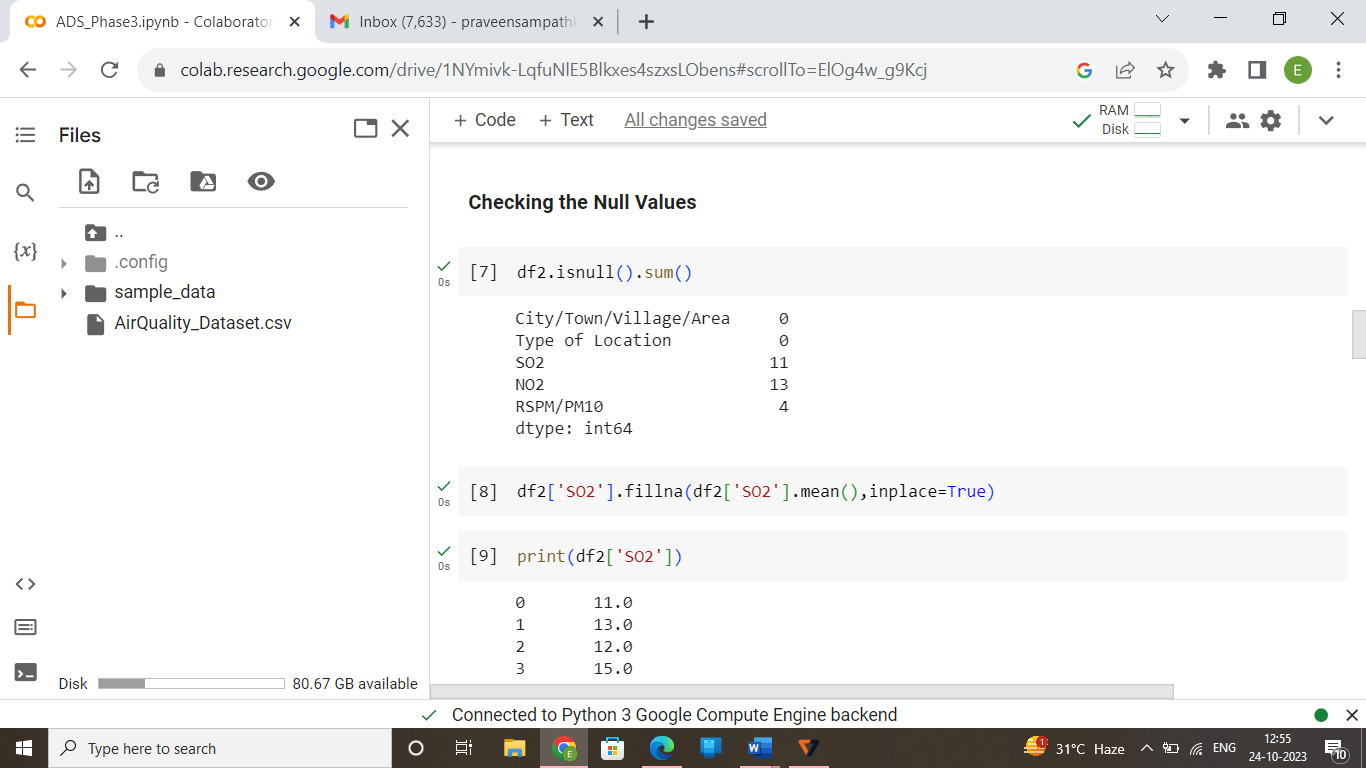


df2.describe()

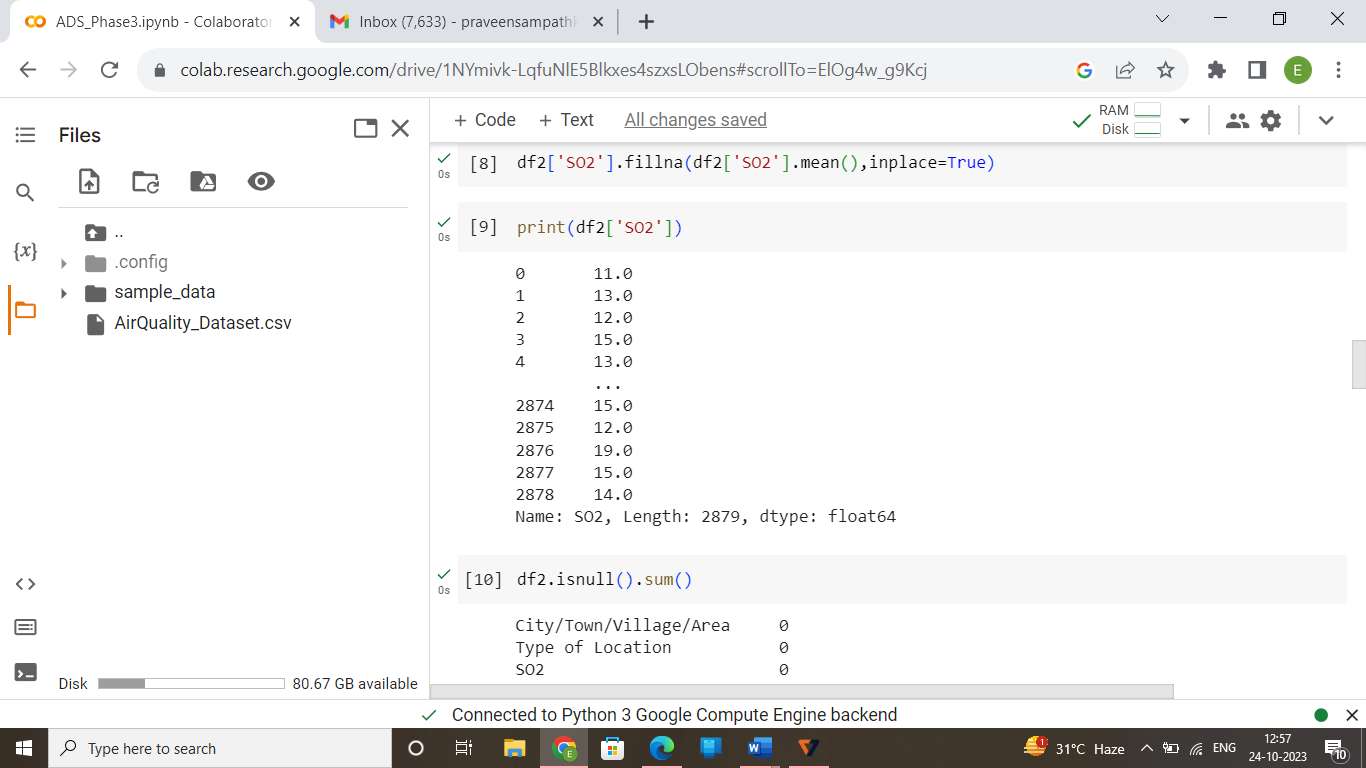


**Checking the Null Values:**

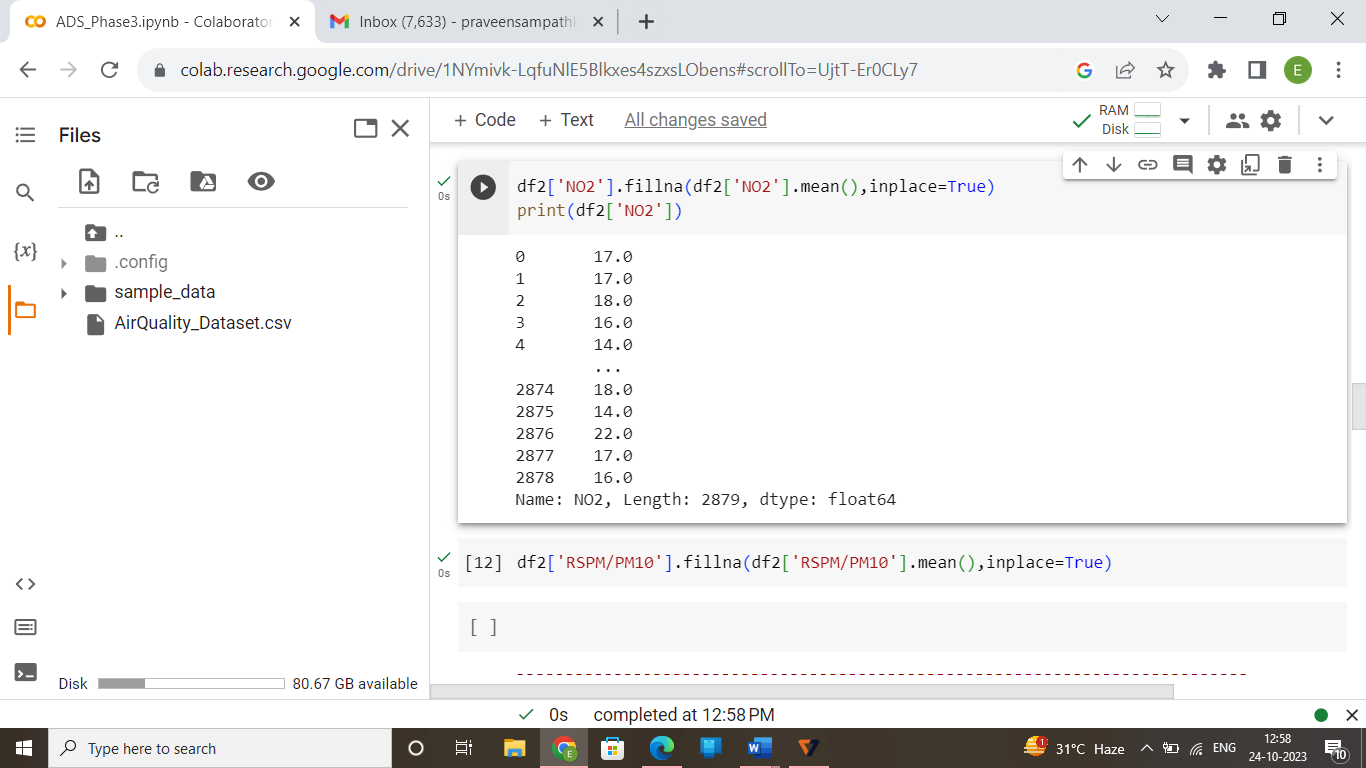
df2.isnull().sum()



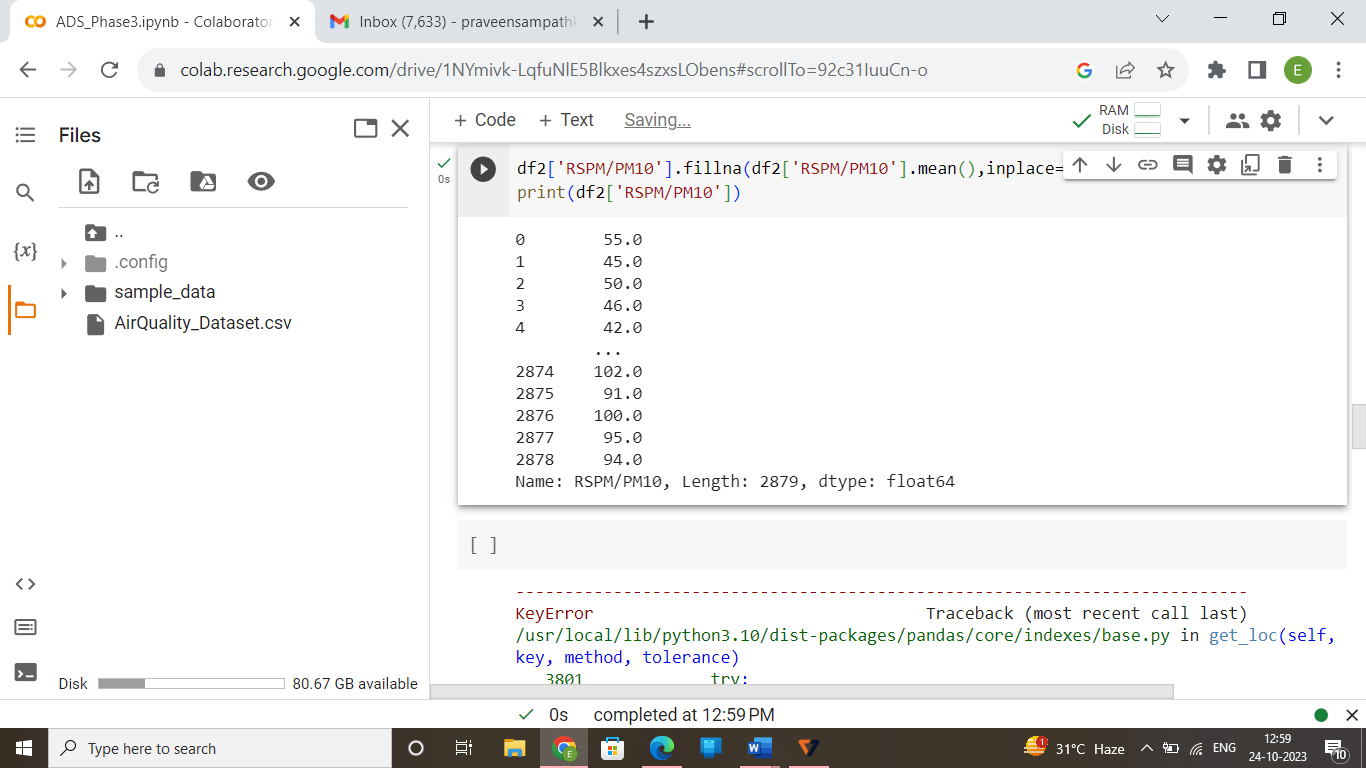
df2['SO2'].fillna(df2['SO2'].mean(),inplace=True)



df2['NO2'].fillna(df2['NO2'].mean(),inplace=True)



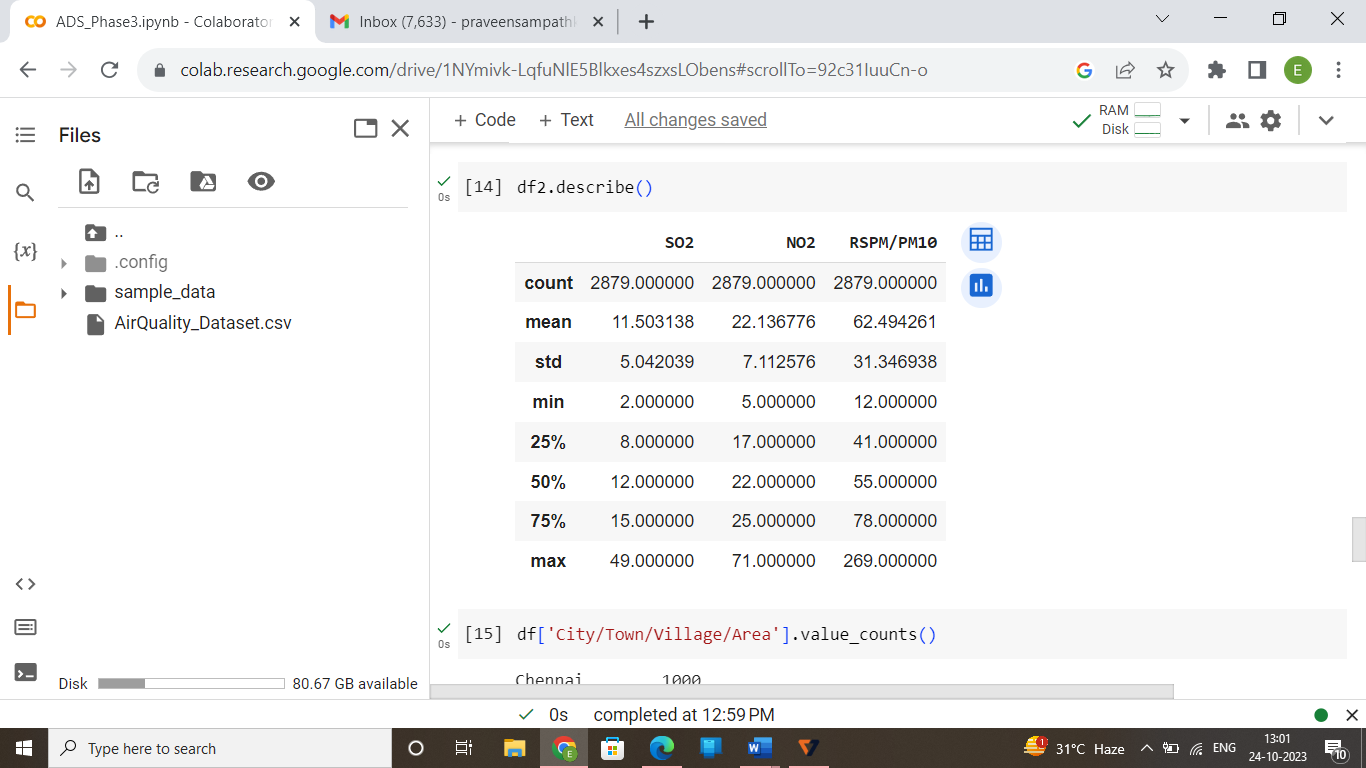
df2['RSPM/PM10'].fillna(df2['RSPM/PM10'].mean(),inplace=True)



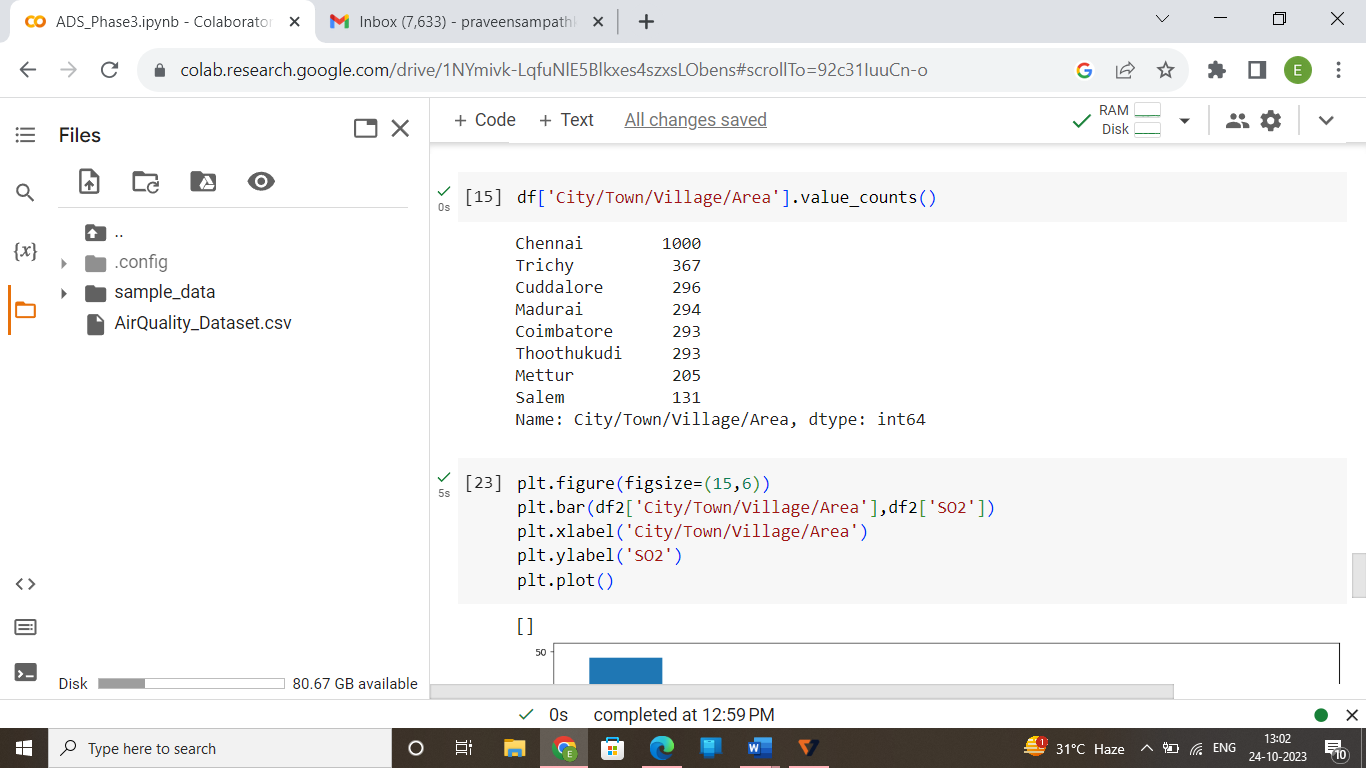
df2.isnull().sum()



df2.describe()



df['City/Town/Village/Area'].value\_counts()



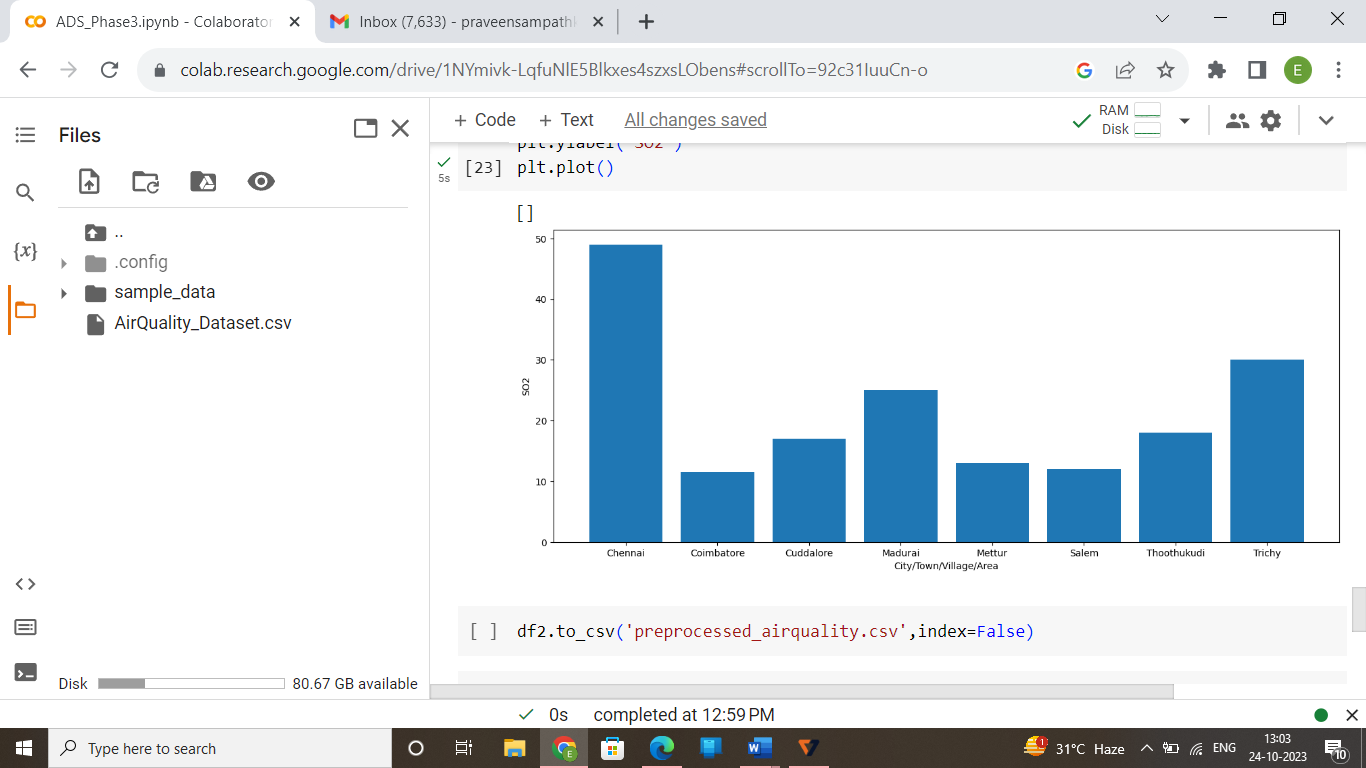
plt.figure(figsize=(15,6))

plt.bar(df2['City/Town/Village/Area'],df2['SO2'])

plt.xlabel('City/Town/Village/Area')

plt.ylabel('SO2')

plt.plot()



**CODE:**

Loading the pre-processed dataset:

import pandas as pd

df=pd.read\_csv('preprocessed\_airquality.csv')

**One Hot Encoding:**

For categorical variables where no such ordinal relationship exists, the integer encoding is not enough.

In fact, using the one hot encoding technique and allowing the model to assume a natural ordering between categories may result in poor performance or unexpected results. This can be applied to the integer representation.

pd.get\_dummies(df,columns=['City/Town/Village/Area','Type of Location'])

dist=(df['City/Town/Village/Area'])

distset=set(dist)

dd=list(distset)

dict0fwords={dd[i]:i for i in range(0,len(dd))}

df['City/Town/Village/Area']=df['City/Town/Village/Area'].map(dict0fwords)

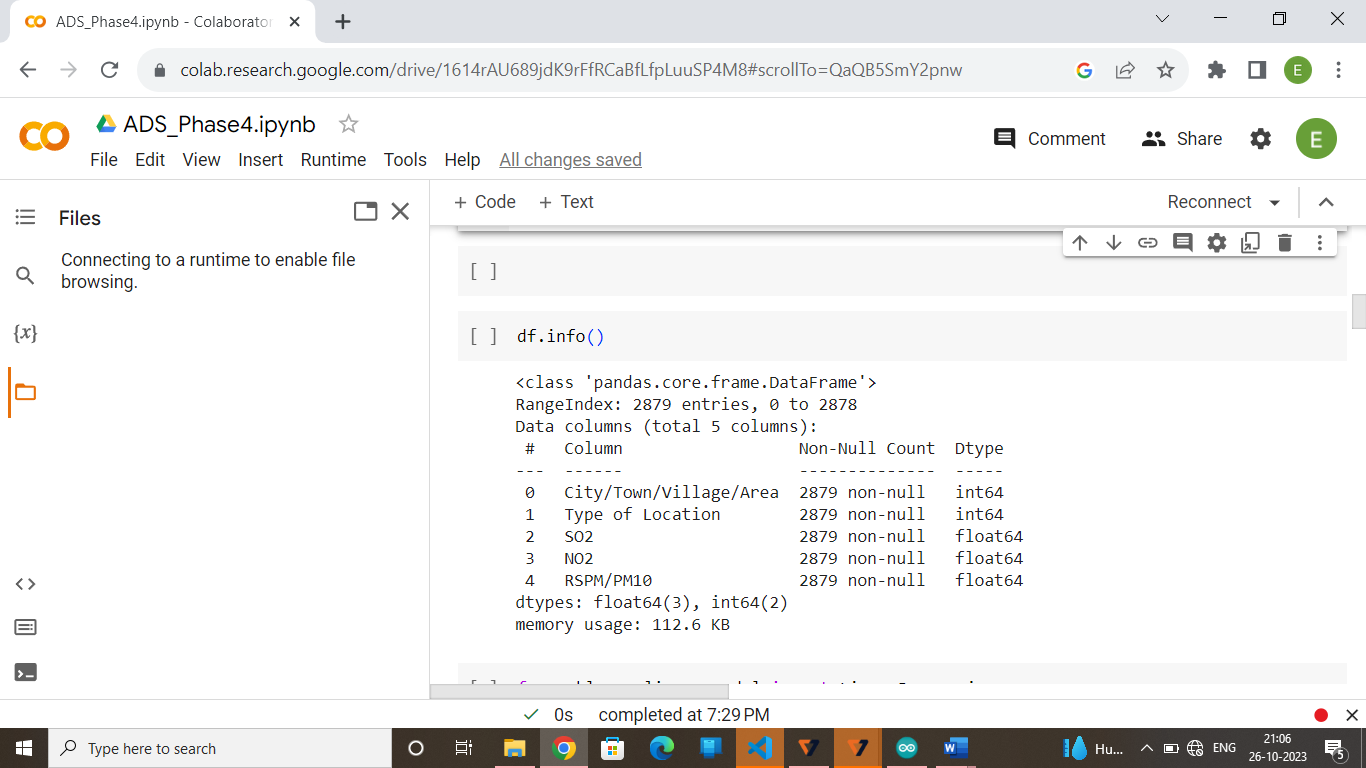
dist=(df['Type of Location'])

distset=set(dist)

dd=list(distset)

dict0fwords={dd[i]:i for i in range(0,len(dd))}

df['Type of Location']=df['Type of Location'].map(dict0fwords)



Model Training:

* Choose a machine learning algorithm
* There are number of different machine learning algorithms that can be used for air quality analysis such as linear regression, KNN, Lasso Regression and Random Forests.

Model Evaluation:

* Model Evaluation is the process of assessing the performance of a machine learning model on unseen data. This is important to ensure that the model will generalize well to the new data.
* There are number of different metrics that can be used to evaluate the performance of air quality analysis and prediction model
* Some of the most common metrics include:
* **Mean Squared Error (MSE):** This metric measures the average squared difference between the different areas/cities in Tamil Nadu and the various pollutants such as SO2, NO2 and RSPM/PM10.
* **Root Mean Squared Error (RMSE):** This metric is the square root of the MSE
* **Mean Absolute Error (MAE):** This metric measures the average absolute difference between the different areas/cities in Tamil Nadu and the various pollutants such as SO2, NO2 and RSPM/PM10.
* **R-Squared:** This metric measures how well the model explains the variation in the pollutants in the different areas.

Linear Regression:

from sklearn.linear\_model import LinearRegression

from sklearn.model\_selection import train\_test\_split

X=df[['City/Town/Village/Area']].values

y=df[['SO2','NO2','RSPM/PM10']].values

X\_train,X\_test,y\_train,y\_test=train\_test\_split(X,y,test\_size=0.2,shuffle=False)

lr\_model=LinearRegression()

lr\_model.fit(X\_train,y\_train)

from sklearn.metrics import mean\_squared\_error,mean\_absolute\_error,mean\_absolute\_percentage\_error,r2\_score

from sklearn.preprocessing import MinMaxScaler

scaler=MinMaxScaler()

X=scaler.fit\_transform(X)

y=scaler.fit\_transform(y.reshape(-1,1))

y\_pred=lr\_model.predict(X\_test)

lr\_model.fit(X\_train,y\_train)



y\_pred=scaler.inverse\_transform(y\_pred)

import numpy as np

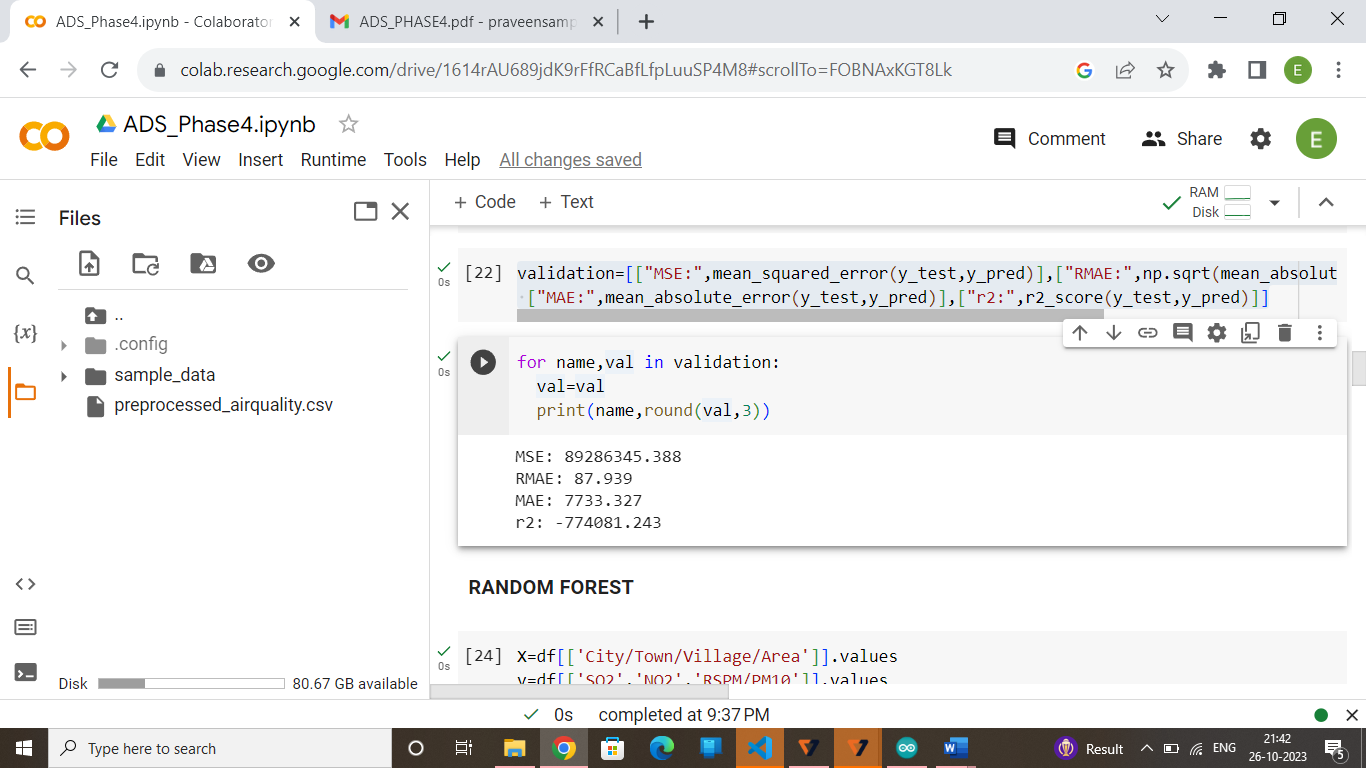
validation=[["MSE:",mean\_squared\_error(y\_test,y\_pred)],["RMAE:",np.sqrt(mean\_absolute\_error(y\_test,y\_pred))],

 ["MAE:",mean\_absolute\_error(y\_test,y\_pred)],["r2:",r2\_score(y\_test,y\_pred)]]

for name,val in validation:

  val=val

  print(name,round(val,3))



**RANDOM FORESTS:**

X=df[['City/Town/Village/Area']].values

y=df[['SO2','NO2','RSPM/PM10']].values

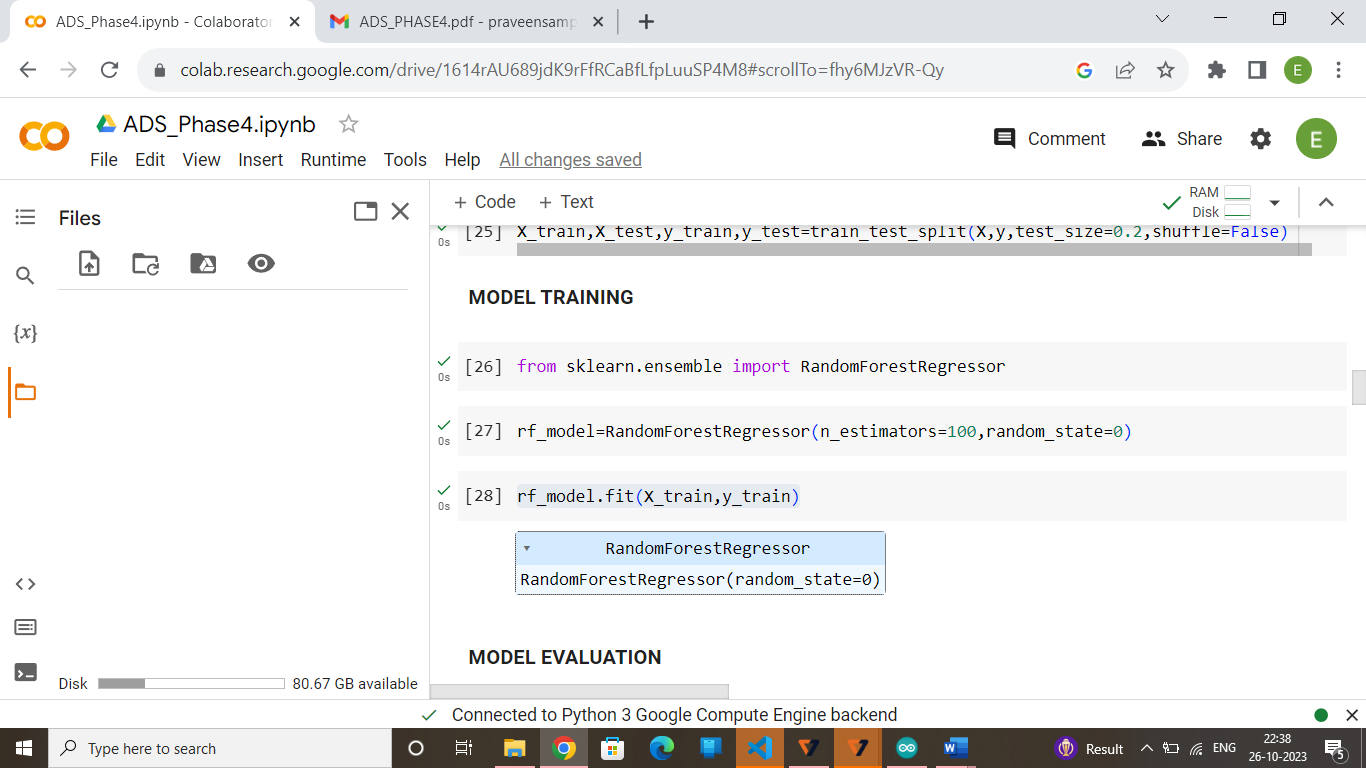
X\_train,X\_test,y\_train,y\_test=train\_test\_split(X,y,test\_size=0.2,shuffle=False)

**Model Training**

from sklearn.ensemble import RandomForestRegressor

rf\_model=RandomForestRegressor(n\_estimators=100,random\_state=0)

rf\_model.fit(X\_train,y\_train)



**Model Evaluation**

from sklearn.metrics import mean\_squared\_error

import math

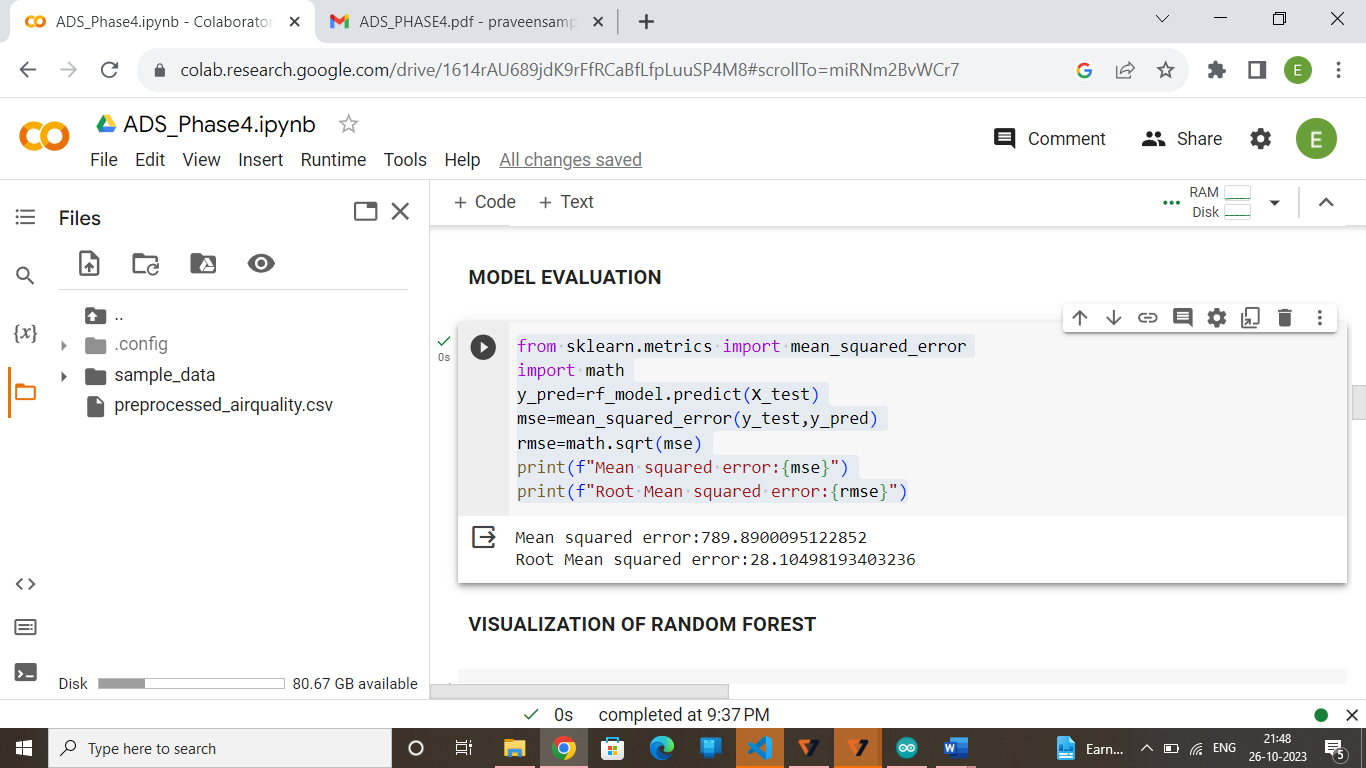
y\_pred=rf\_model.predict(X\_test)

mse=mean\_squared\_error(y\_test,y\_pred)

rmse=math.sqrt(mse)

print(f"Mean squared error:{mse}")

print(f"Root Mean squared error:{rmse}")



**Visualization of Random Forest**

import matplotlib.pyplot as plt

data\_range=df.index[-len(y\_test):]

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

plt.plot(df['City/Town/Village/Area'],label='Areas',linewidth=2)

plt.plot(df['NO2'],label='Pollution of NO2',linewidth=2)

plt.title("AIR QUALITY ANALYSIS")

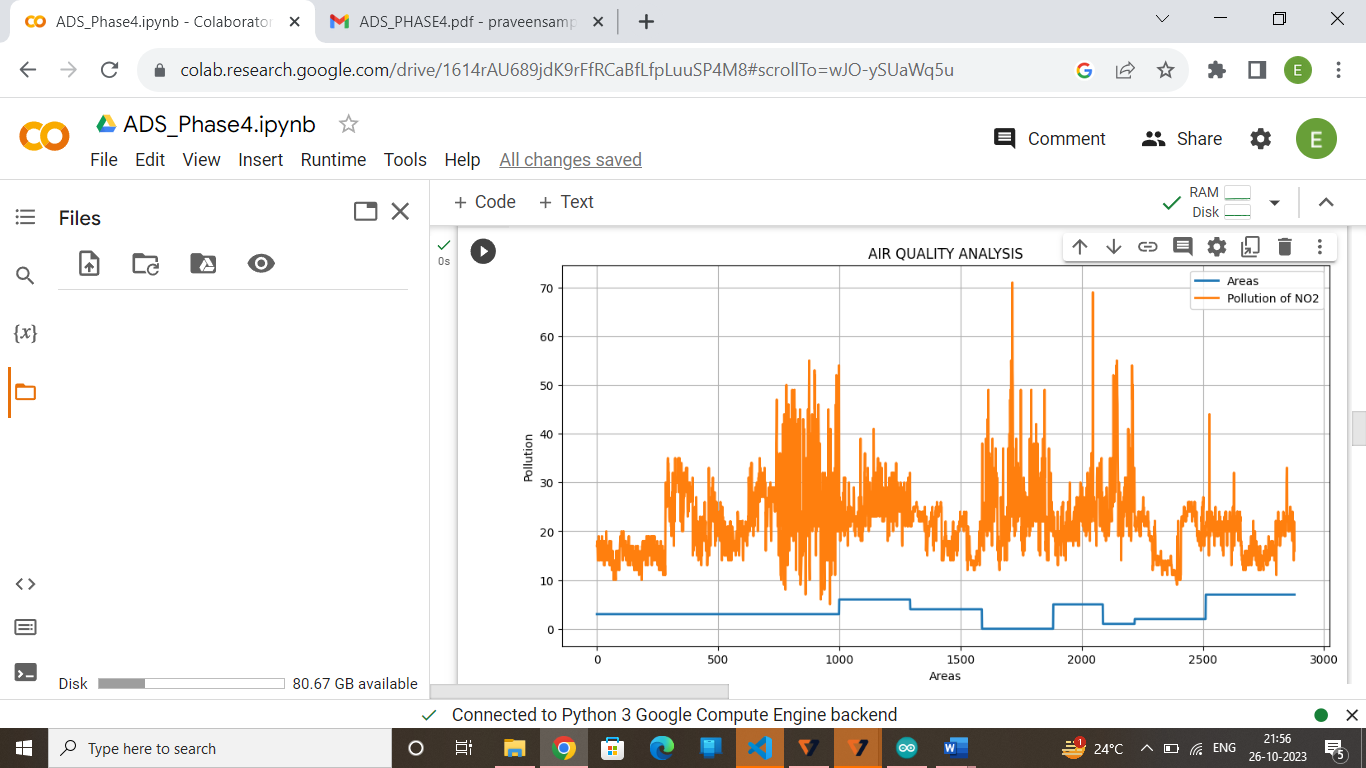
plt.xlabel('Areas')

plt.ylabel('Pollution')

plt.legend()

plt.grid()

plt.show()



import matplotlib.pyplot as plt

data\_range=df.index[-len(y\_test):]

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

plt.plot(df['City/Town/Village/Area'],label='Areas',linewidth=2)

plt.plot(df['SO2'],label='Pollution of SO2',linewidth=2)

plt.title("AIR QUALITY ANALYSIS")

plt.xlabel('Areas')

plt.ylabel('Pollution')

plt.legend()

plt.grid()

plt.show()



import matplotlib.pyplot as plt

data\_range=df.index[-len(y\_test):]

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

plt.plot(df['City/Town/Village/Area'],label='Areas',linewidth=2)

plt.plot(df['RSPM/PM10'],label='Pollution of RSPM/PM10',linewidth=2)

plt.title("AIR QUALITY ANALYSIS")

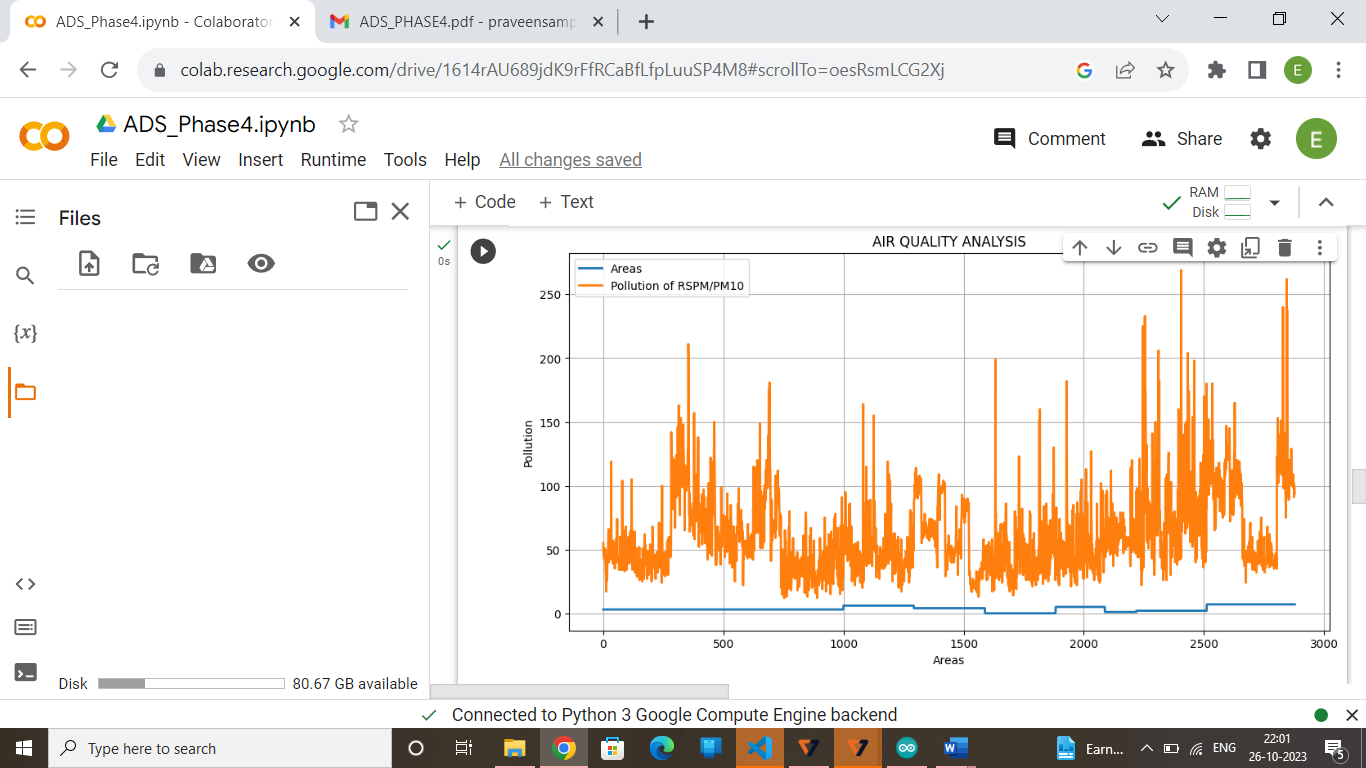
plt.xlabel('Areas')

plt.ylabel('Pollution')

plt.legend()

plt.grid()

plt.show()

****

**KNN:**

X=df[['City/Town/Village/Area']].values

y=df[['SO2','NO2','RSPM/PM10']].values

X\_train,X\_test,y\_train,y\_test=train\_test\_split(X,y,test\_size=0.2,shuffle=False)

from sklearn.neighbors import KNeighborsRegressor

knn\_model=KNeighborsRegressor(n\_neighbors=5)

knn\_model.fit(X\_train,y\_train)

from sklearn.metrics import mean\_squared\_error

import math

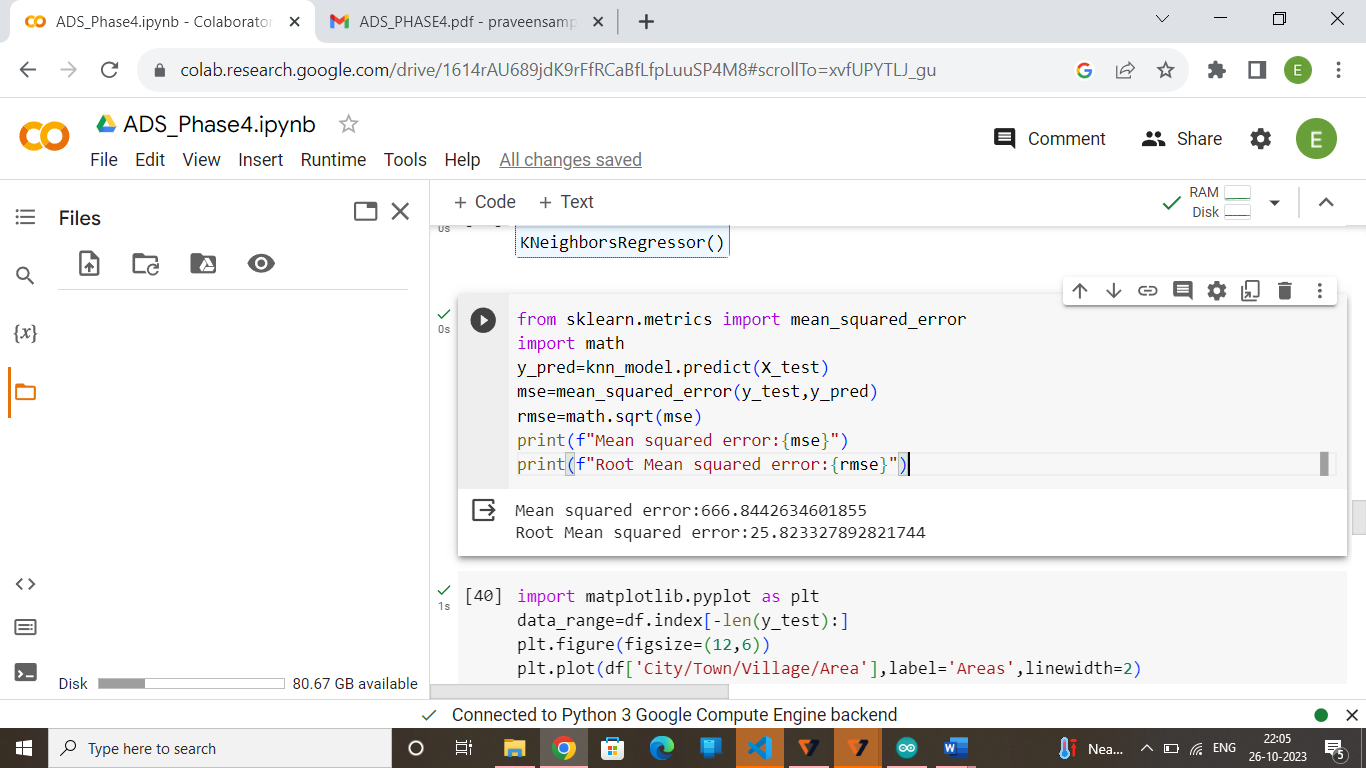
y\_pred=knn\_model.predict(X\_test)

mse=mean\_squared\_error(y\_test,y\_pred)

rmse=math.sqrt(mse)

print(f"Mean squared error:{mse}")

print(f"Root Mean squared error:{rmse}")



import matplotlib.pyplot as plt

data\_range=df.index[-len(y\_test):]

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

plt.plot(df['City/Town/Village/Area'],label='Areas',linewidth=2)

plt.plot(df['NO2'],label='Pollution of NO2',linewidth=2)

plt.title("AIR QUALITY ANALYSIS")

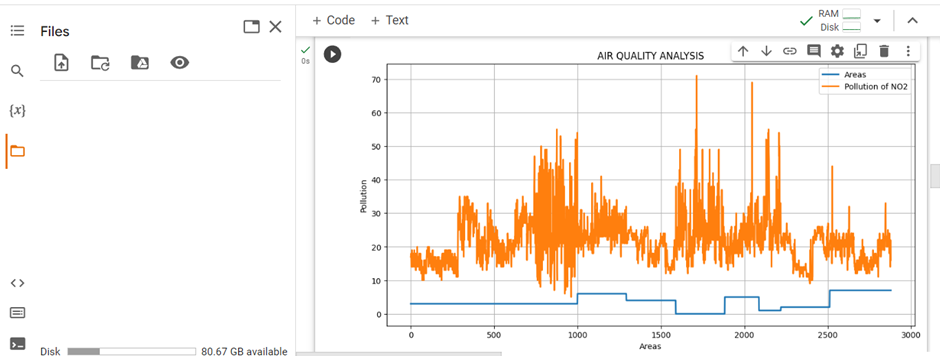
plt.xlabel('Areas')

plt.ylabel('Pollution')

plt.legend()

plt.grid()

plt.show()



import matplotlib.pyplot as plt

data\_range=df.index[-len(y\_test):]

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

plt.plot(df['City/Town/Village/Area'],label='Areas',linewidth=2)

plt.plot(df['SO2'],label='Pollution of SO2',linewidth=2)

plt.title("AIR QUALITY ANALYSIS")

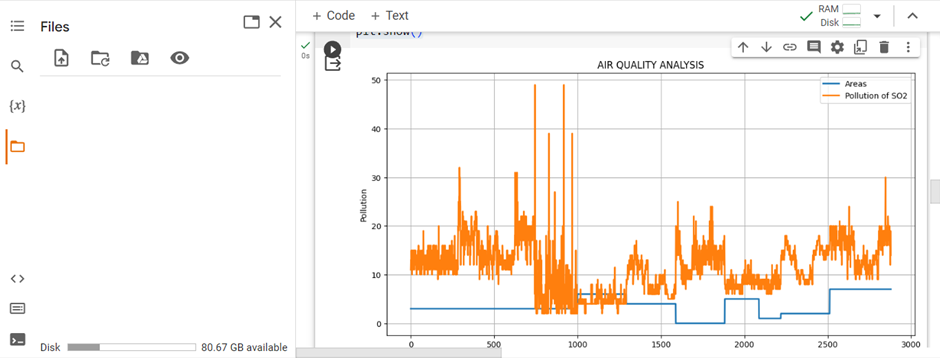
plt.xlabel('Areas')

plt.ylabel('Pollution')

plt.legend()

plt.grid()

plt.show()

****

import matplotlib.pyplot as plt

data\_range=df.index[-len(y\_test):]

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

plt.plot(df['City/Town/Village/Area'],label='Areas',linewidth=2)

plt.plot(df['RSPM/PM10'],label='Pollution of RSPM/PM10',linewidth=2)

plt.title("AIR QUALITY ANALYSIS")

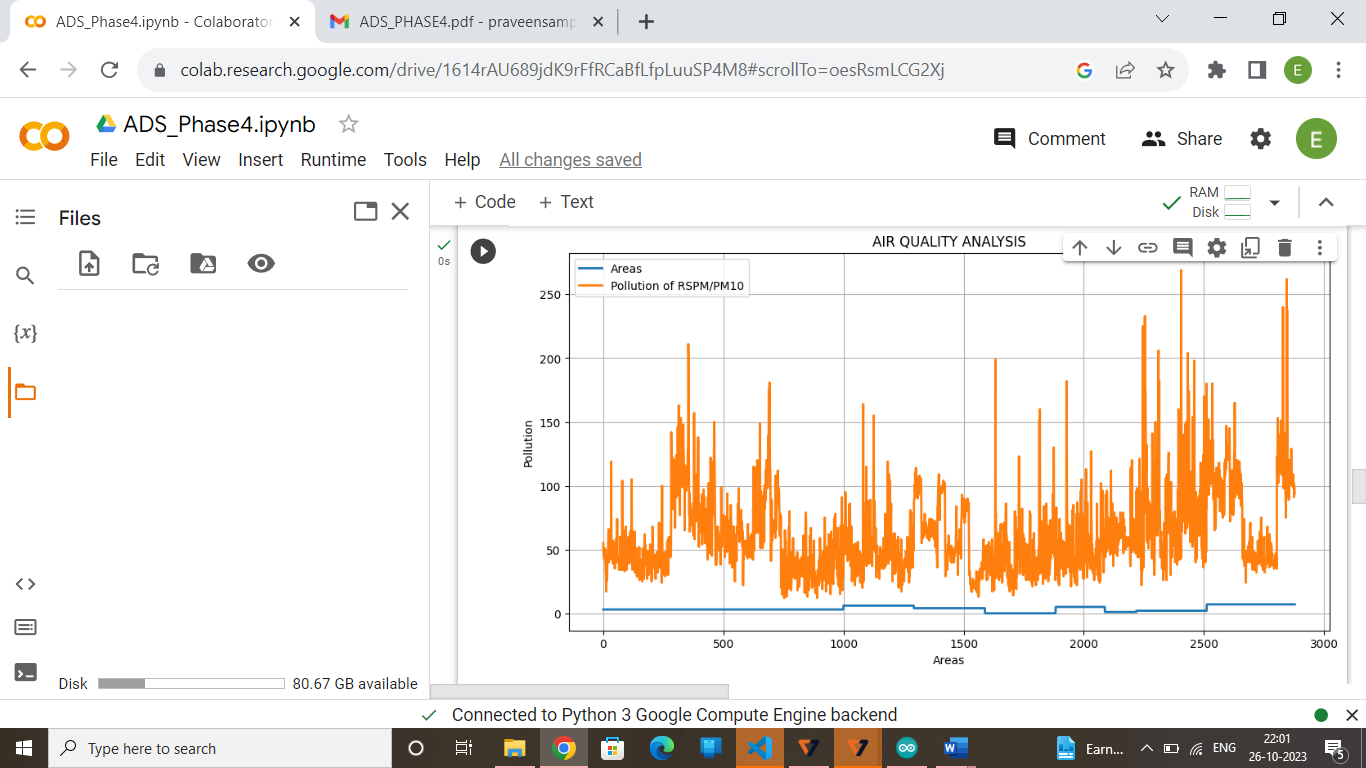
plt.xlabel('Areas')

plt.ylabel('Pollution')

plt.legend()

plt.grid()

plt.show()

****

**Import Libraries:**

from sklearn.linear\_model import Lasso,SGDRegressor,Ridge

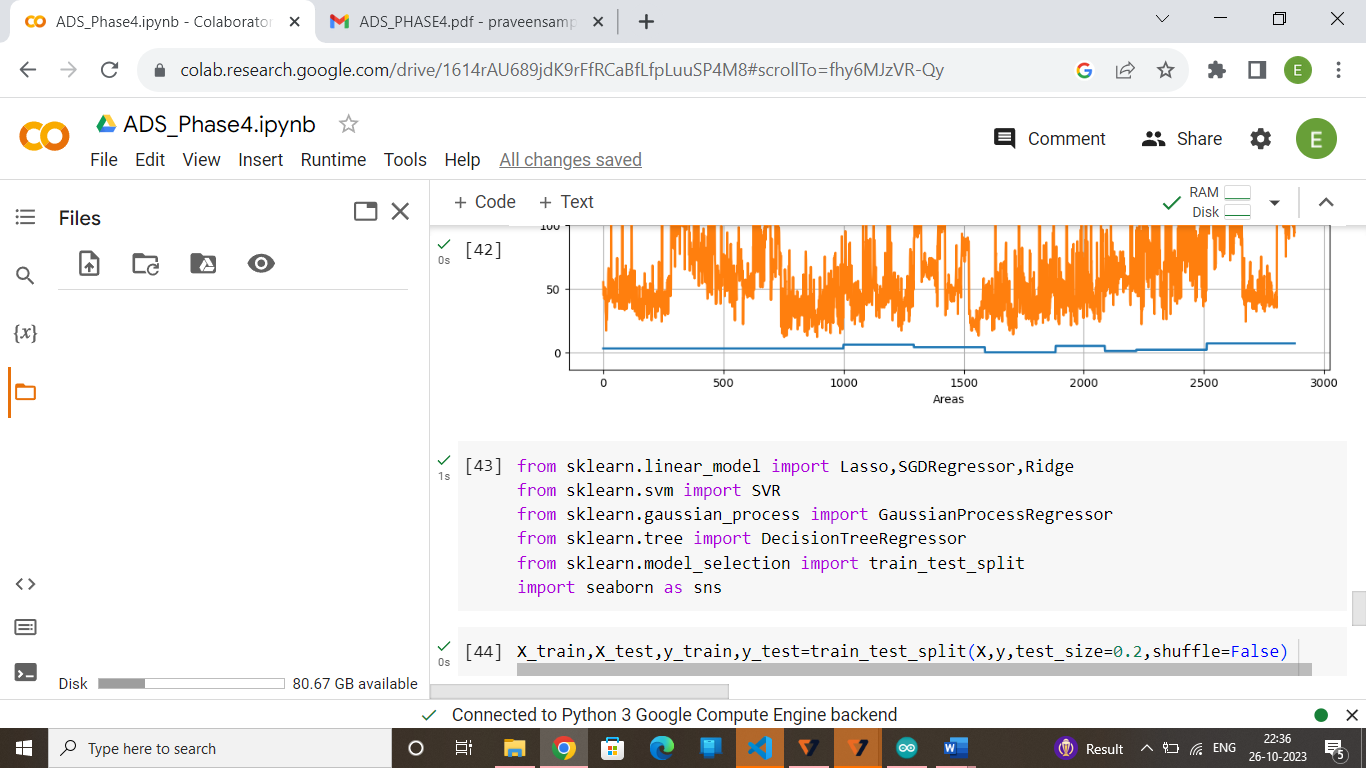
from sklearn.svm import SVR

from sklearn.gaussian\_process import GaussianProcessRegressor

from sklearn.tree import DecisionTreeRegressor

from sklearn.model\_selection import train\_test\_split

import seaborn as sns

****

**Model Training:**

X\_train,X\_test,y\_train,y\_test=train\_test\_split(X,y,test\_size=0.2,shuffle=False)

ms=[]

ma=[]

mse=mean\_squared\_error

mae=mean\_absolute\_error

def model\_training\_and\_score(model):

  model.fit(X\_train,y\_train)

  y\_pred=np.nan\_to\_num(model.predict(X\_test))

  print(mse(y\_test,y\_pred))

  print(mae(y\_test,y\_pred))

  ms.append(mse(y\_test,y\_pred))

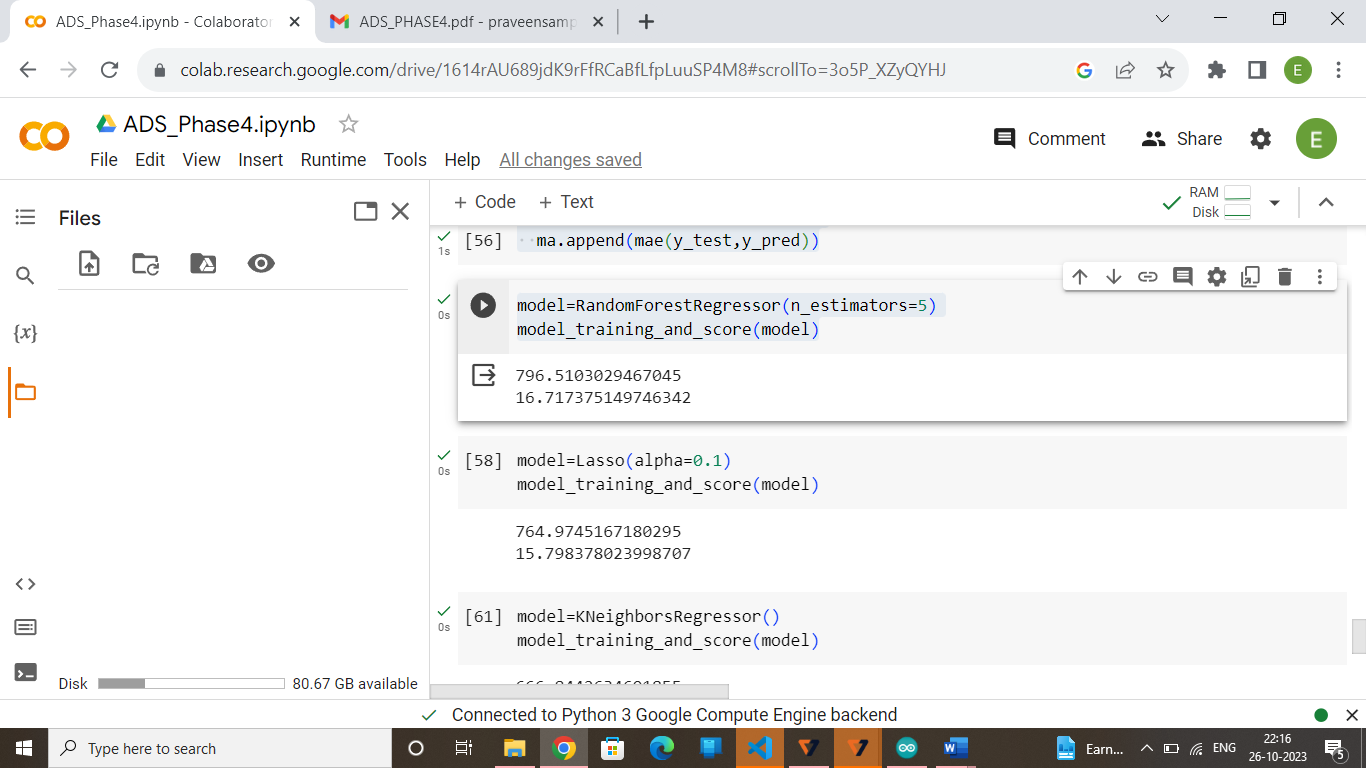
  ma.append(mae(y\_test,y\_pred))

**Model Evaluation:**

1. **Random Forest**

model=RandomForestRegressor(n\_estimators=5)

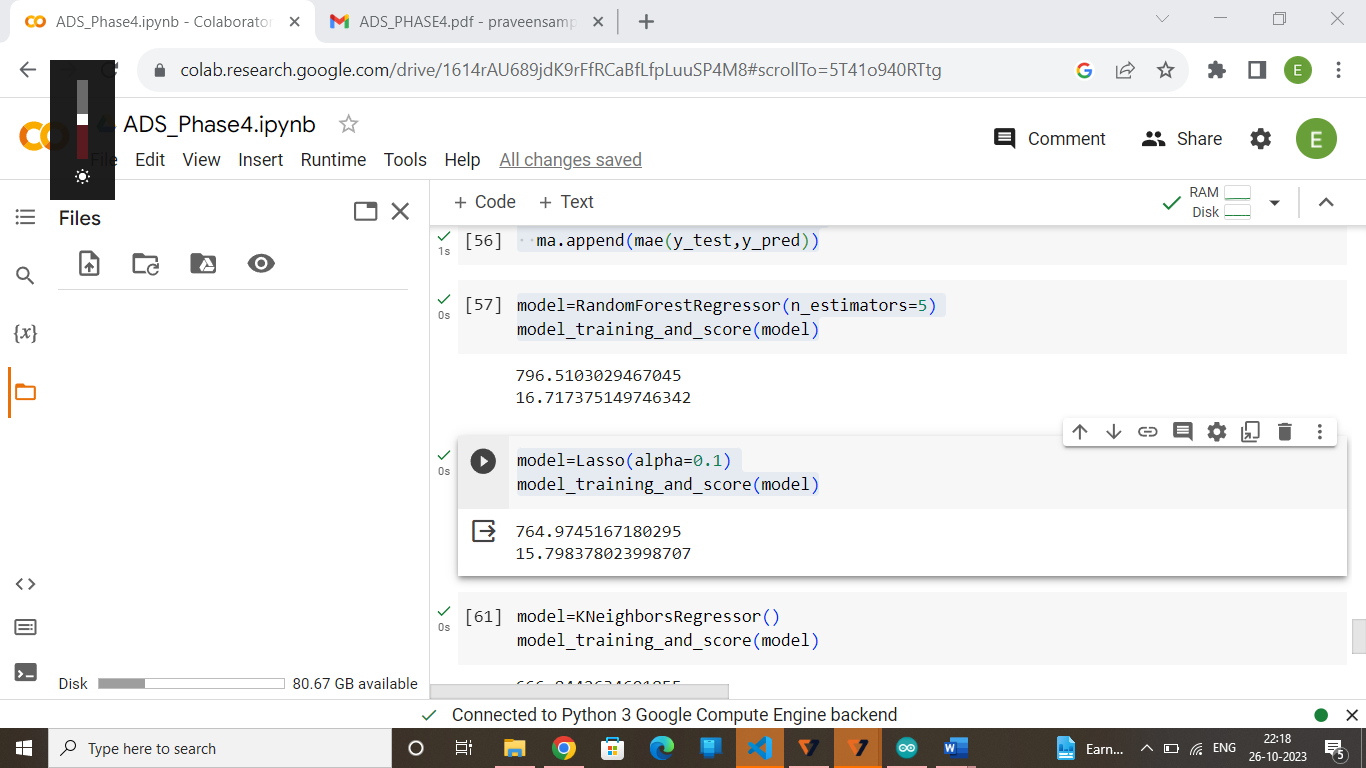
model\_training\_and\_score(model)



**2.Lasso Regrssion**

model=Lasso(alpha=0.1)

model\_training\_and\_score(model)



**3.K Neighbors Regression**

model=KNeighborsRegressor()

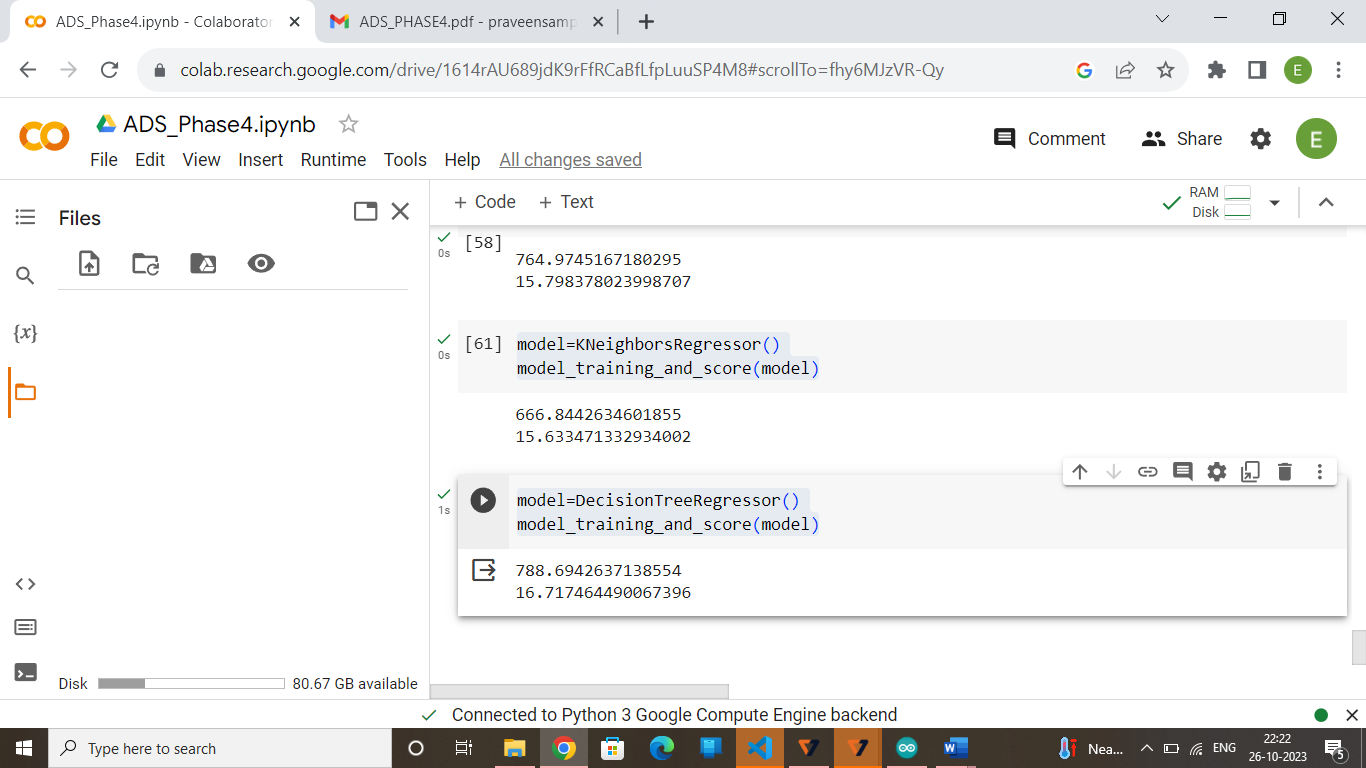
model\_training\_and\_score(model)



**4. Decision Tree Regression**

model=DecisionTreeRegressor()

model\_training\_and\_score(model)



**CONCLUSION:**

In conclusion, this project focuses on analyzing and predicting air quality in Tamil Nadu has yielded valuable insights and outcomes. Through the collection and analysis of historical air quality data, we are able to identify trends, seasonal variations, and the impact of various factors or air quality. Our predictive models, based on machine learning algorithms, demonstrated reasonable accuracy in forecasting air quality levels.