DATA ANALYTICS WITH COGNOS

**PROJECT\_TITLE:**

# AIR QUALITY ANALYSIS

**TEAM MEMBERS**

RASHIKA S POOJA E

SHAHRUKH KHAN B KEERTHANA N PRATHIBA S

# INTRODUCTION

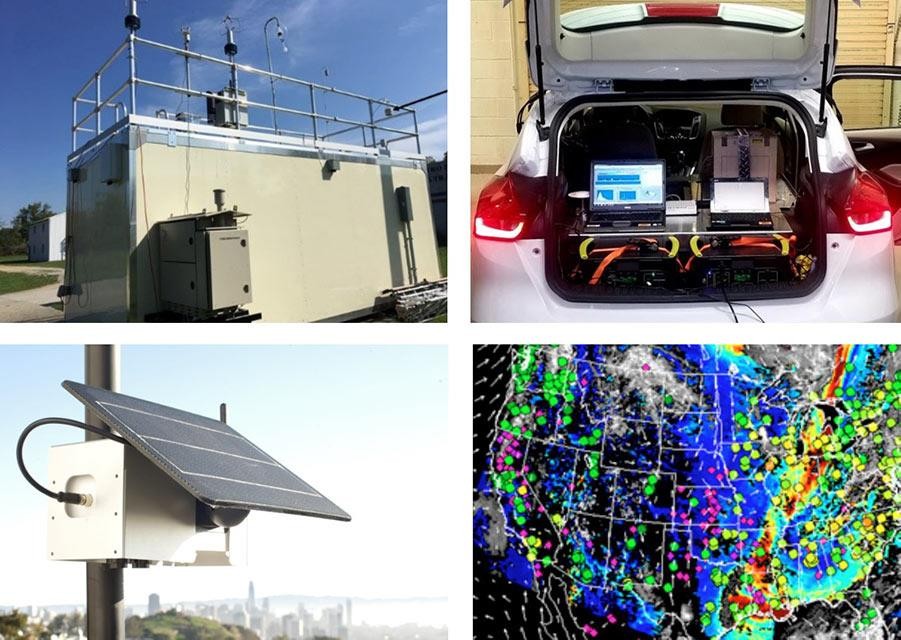
Air quality analysis is a crucial process aimed at assessing and understanding the composition of the atmosphere in a specific area, with a focus on the presence and concentration of various pollutants and particulate matter. It plays a vital role in safeguarding public health, the environment, and overall quality of life.

The quality of the air we breathe is of paramount importance, as it directly impacts human health, ecosystems, and climate. Airborne pollutants, such as particulate matter (PM), nitrogen oxides (NOx), sulfur dioxide (SO2), carbon monoxide (CO), ozone (O3), and volatile organic compounds (VOCs), can have adverse effects on respiratory health, cardiovascular functions, and can contribute to climate change.

Air quality analysis involves the collection of data through monitoring stations equipped with various sensors and instruments. These stations measure the concentration of pollutants and meteorological parameters like temperature, humidity, wind speed, and direction. The data collected is then processed and analyzed to assess compliance with air quality standards set by regulatory agencies.

# PROJECT DESCRIPTION

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



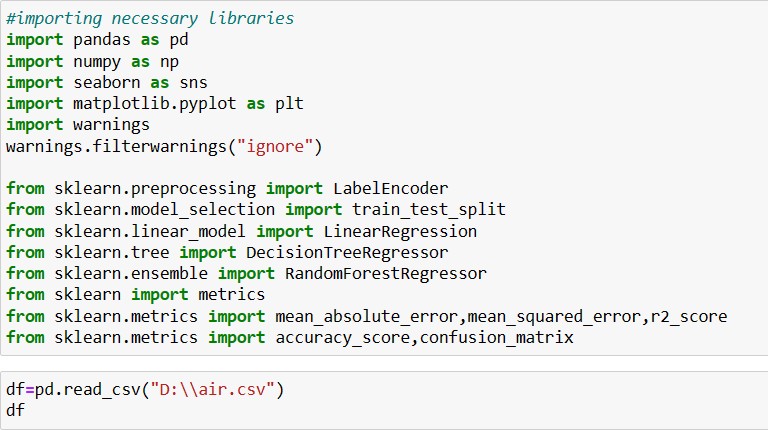
# DATA COLLECTING

Data collection refers to the process of systematically gathering information or observations from various sources or instruments. It is a fundamental step in research, analysis, and decision-making across a wide range of fields, including science, business, healthcare, and social sciences.

**LINK:** [**www.kaggle.air.csv**](http://www.kaggle.air.csv/)

# DATA LOADING

Data loading refers to the process of importing or inputting data into a computer system or software application for further processing, analysis, or storage. It is a crucial step in various data-driven tasks, including data analysis, machine learning, database management, and more.

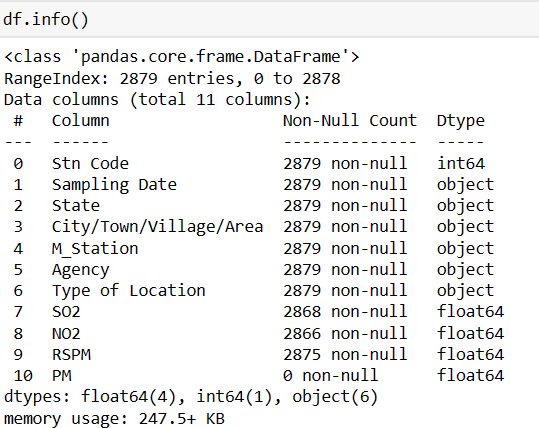


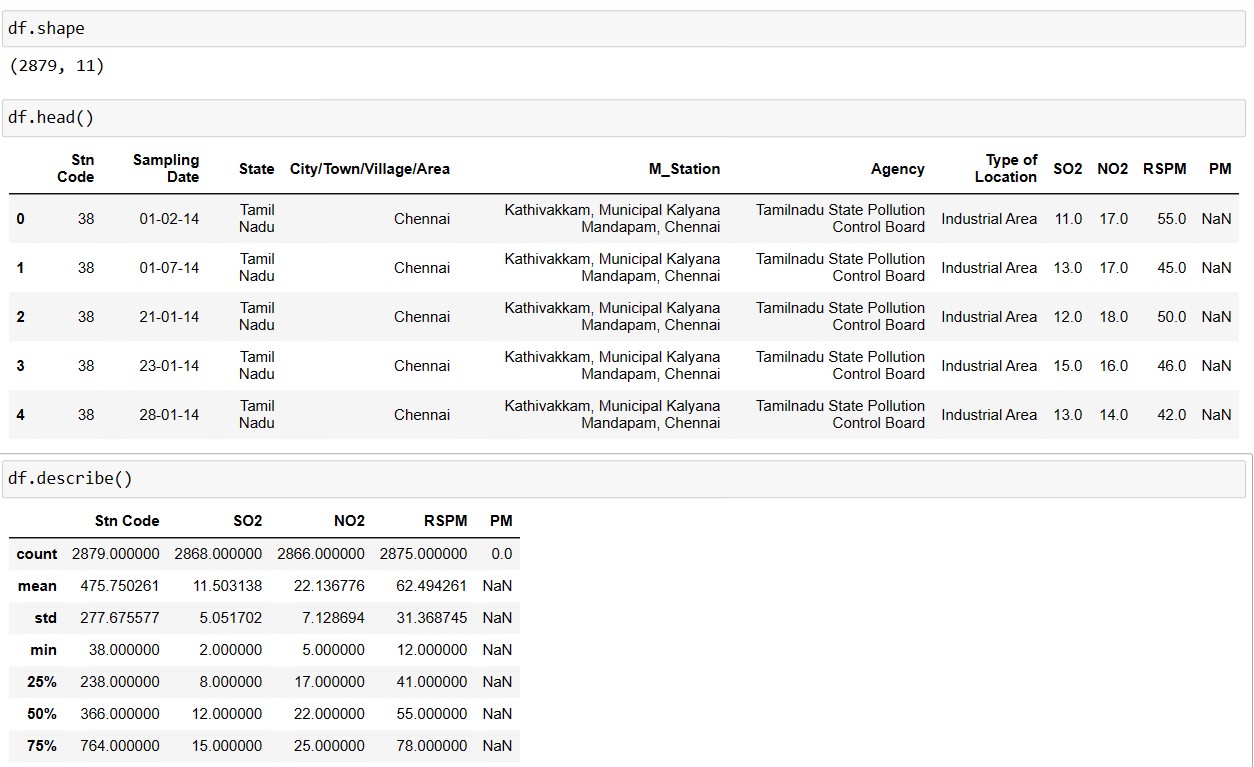
# DATA PRE-PROCESSING

Data pre-proccessing is a crucial step in data analysis and machine learning. It involves preparing and cleaning raw data to make it suitable for further processing. This step is essential because real-world data is often messy, incomplete or contains inconsistencies, which can hinder accurate analysis or modeling.

## UNDERSTANDING THE DATASET

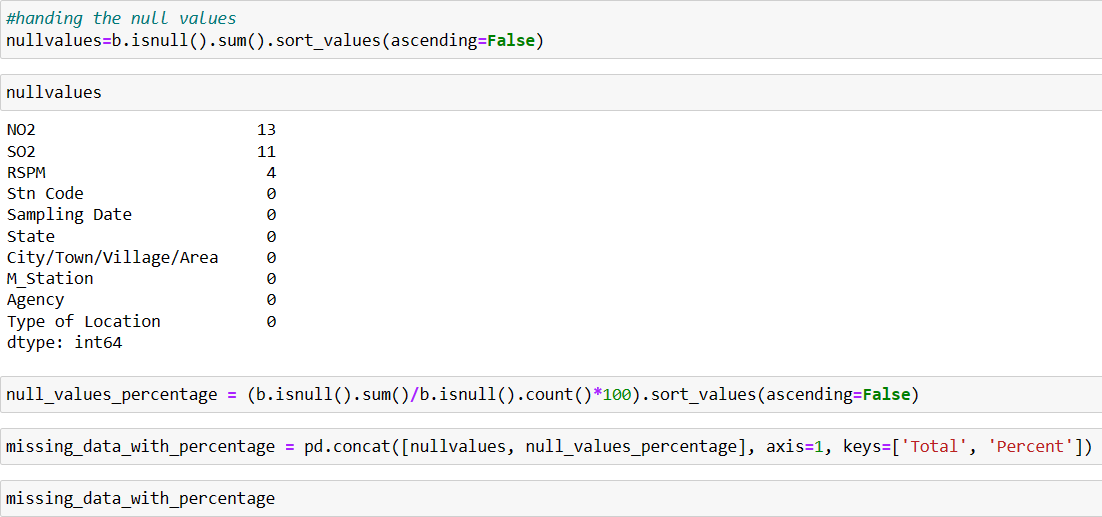
It involves gaining insights into the structure, content and characteristics of the data you are working with.By thoroughly understanding your dataset you lay the foundation for making informed decisions about data preprocessing, feature selection, modeling approaches, and interpreting the results of your analysis.





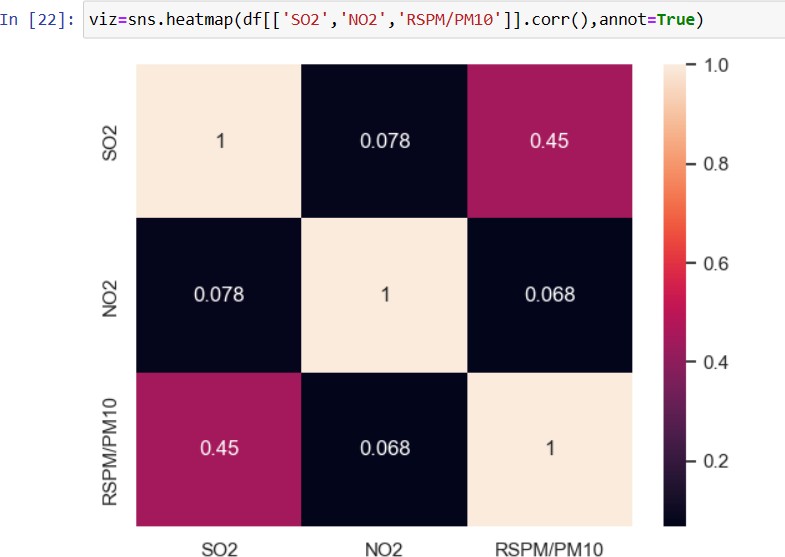
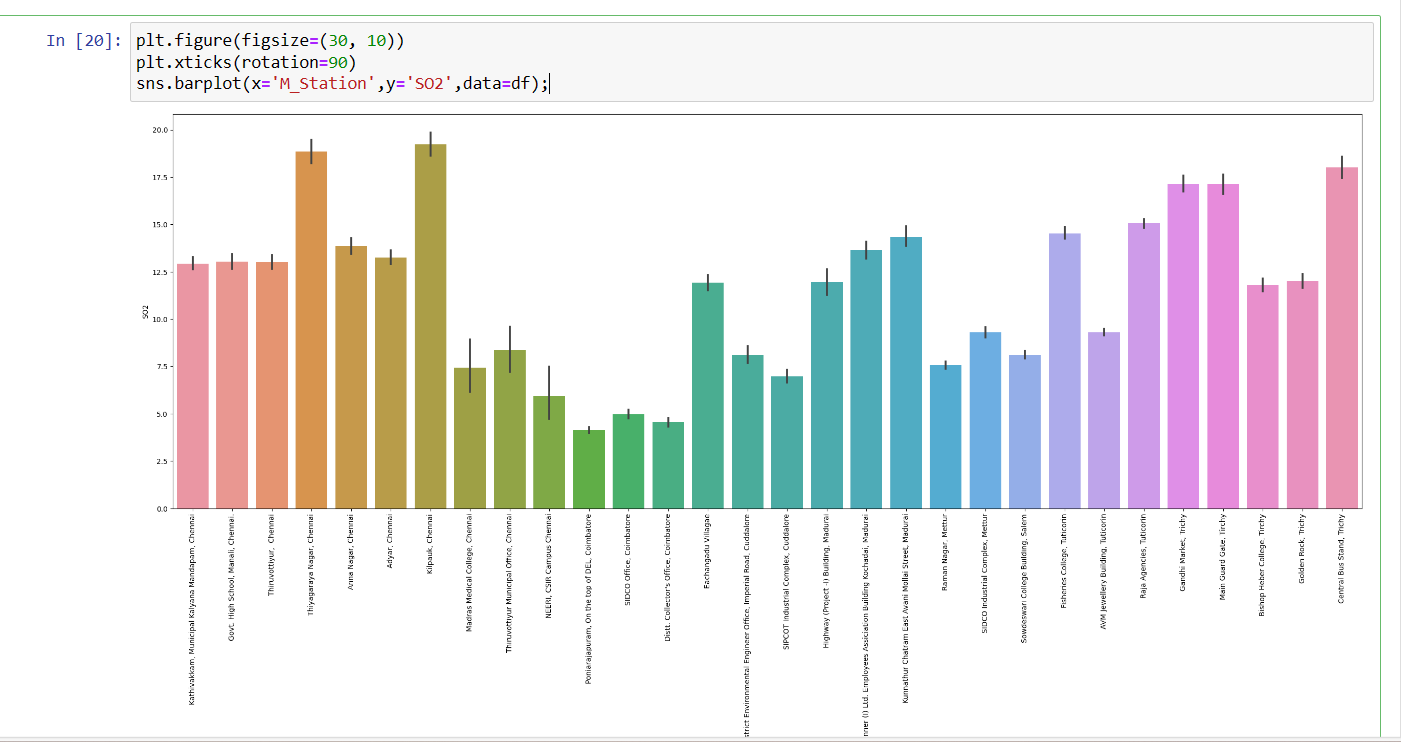
## HANDLING THE MISSING VALUES

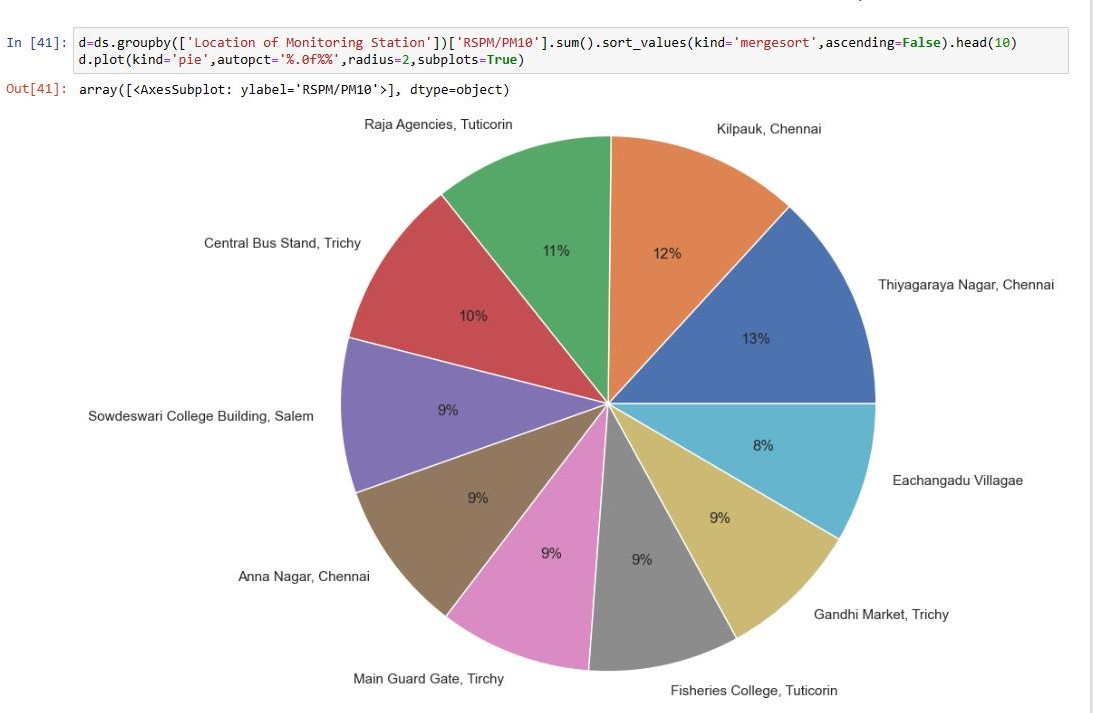
Null values can be handled using some default functions in the pandas.It is very complex to handle the dataset with the null values so it must be very important to clean the dataset before processing.

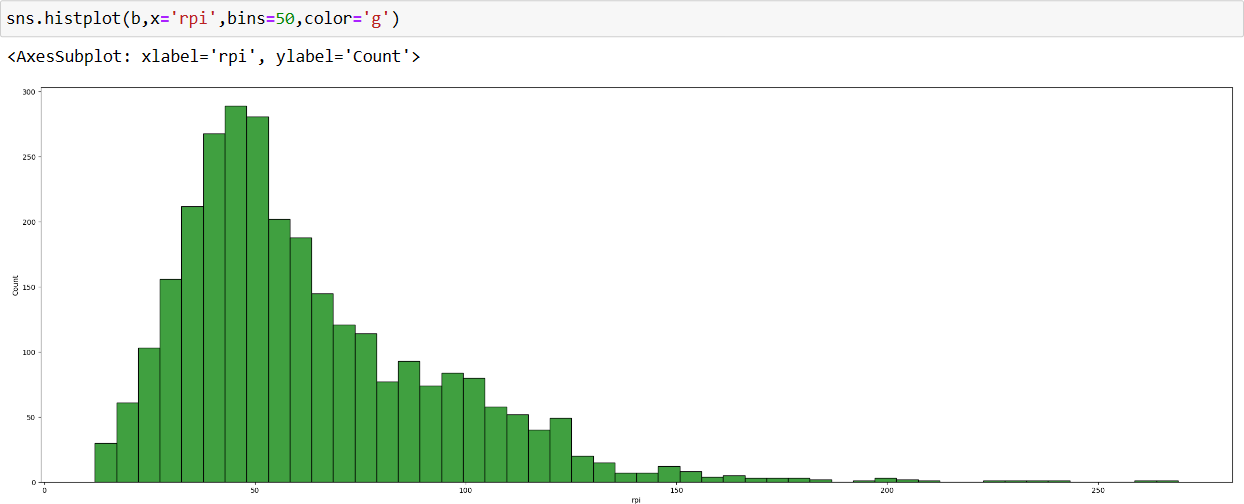


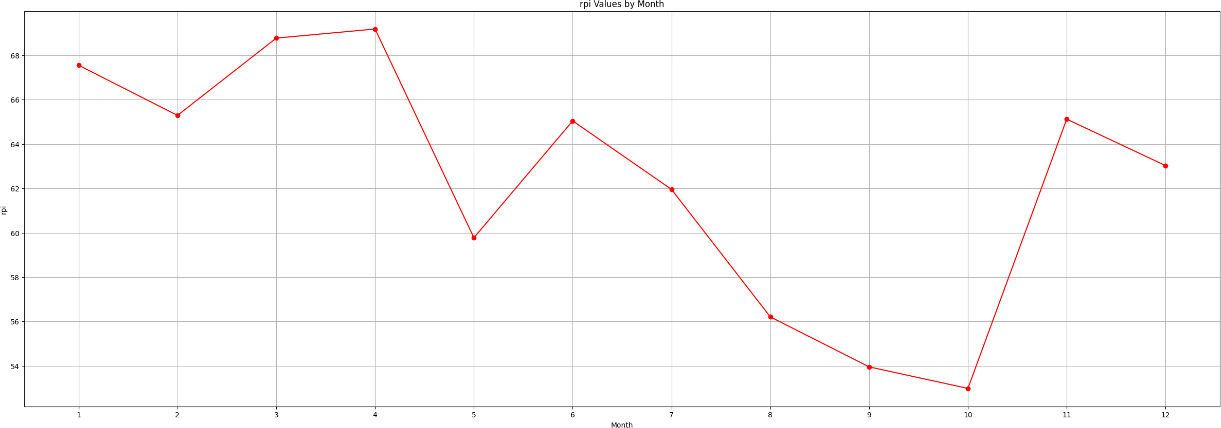
# DATA VISUALIZATION

Data visualization is the process of representing data in graphical or visual formats to help convey information, identify patterns, and gain insights from the data. It is a powerful tool in data analysis and communication, as it allows complex information to be presented in a more understandable and digestible form.



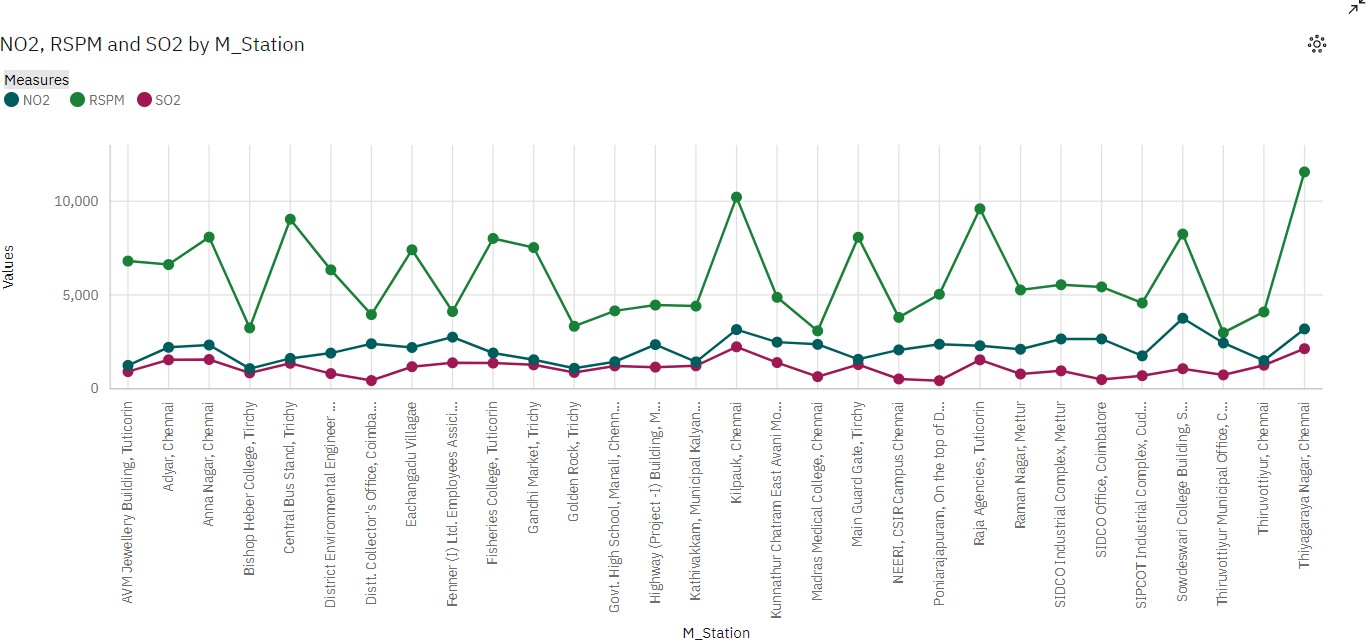
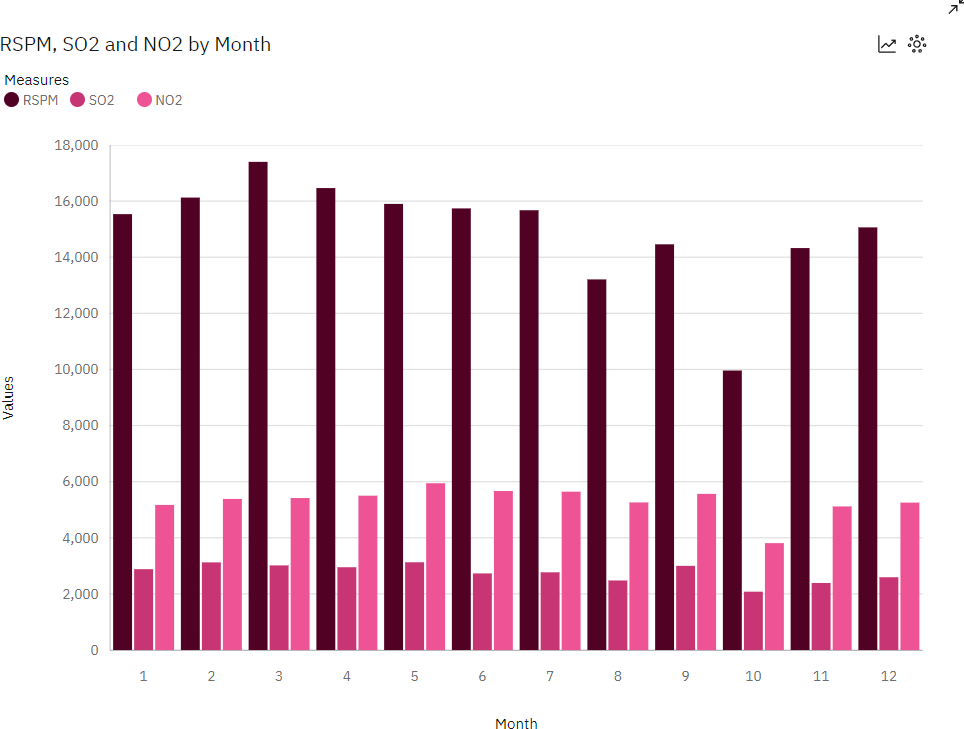


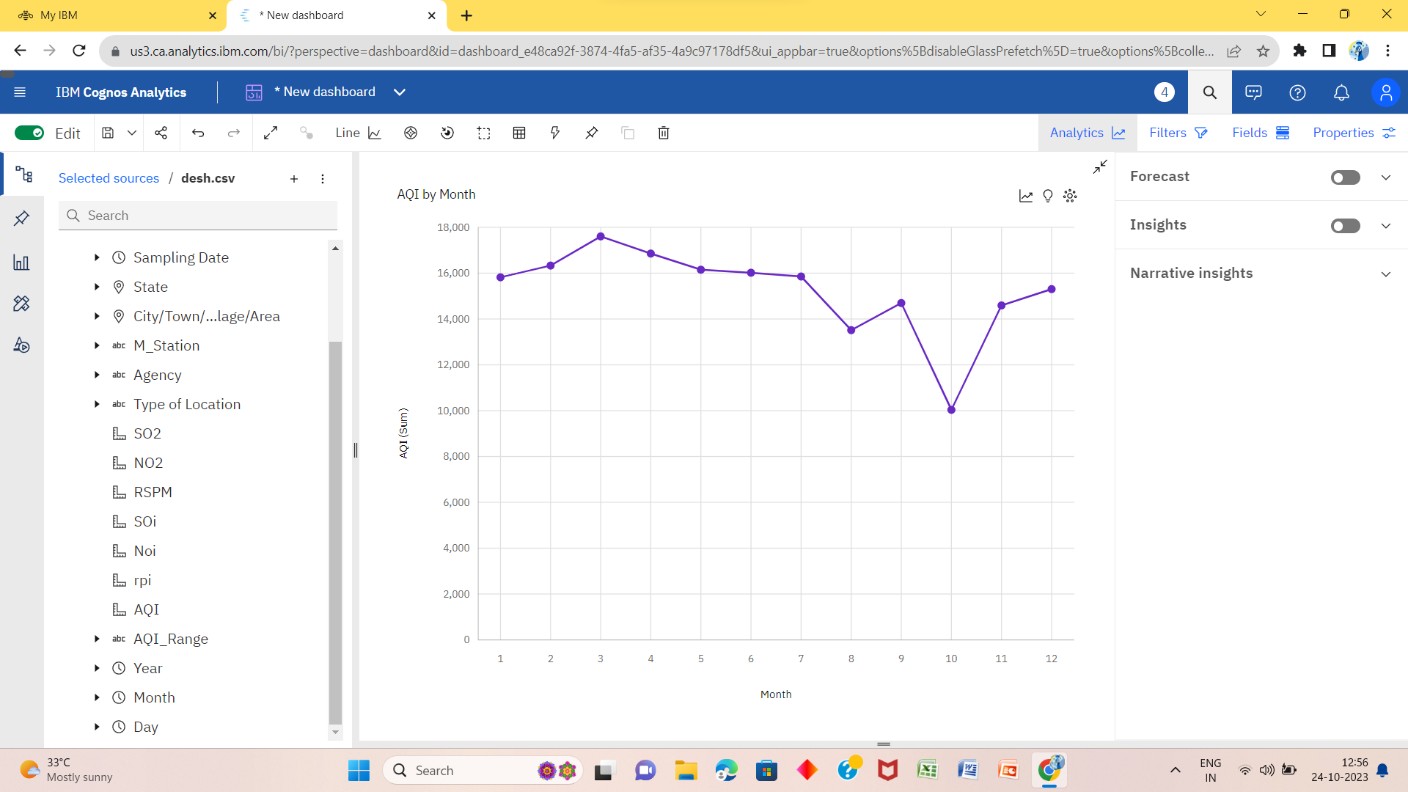


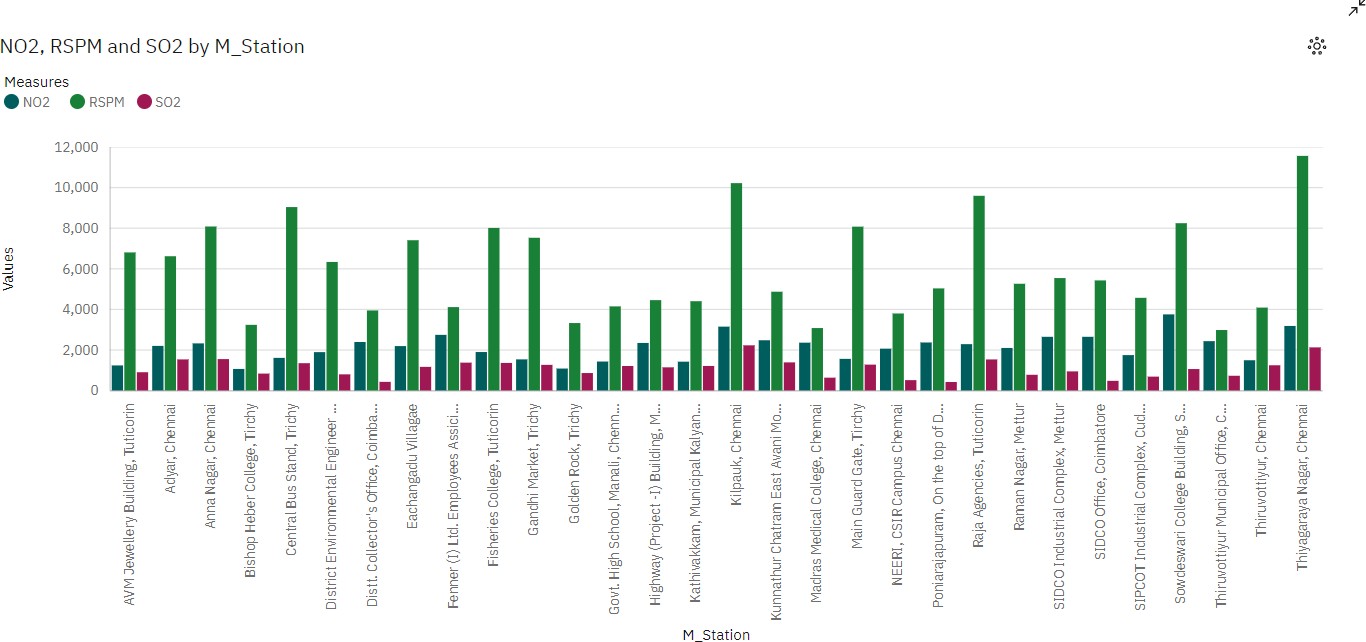


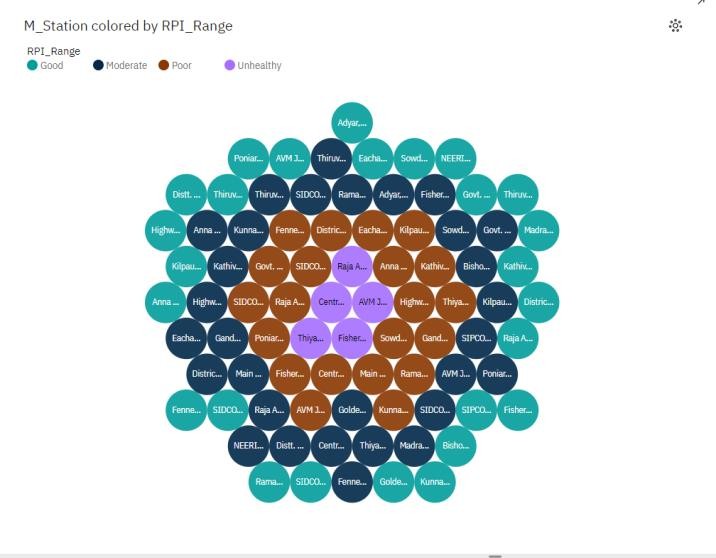
# IBM VISUALIZATION

Visualizations communicate comparisons, relationships, and trends. They emphasize and clarify numbers. You can import custom visualizations in your IBM Cognos Analytics dashboard.IBM Cognos Analytics provides analytic insights that help you to detect and validate important relationships and meaningful differences based on the data that is presented by the visualization.







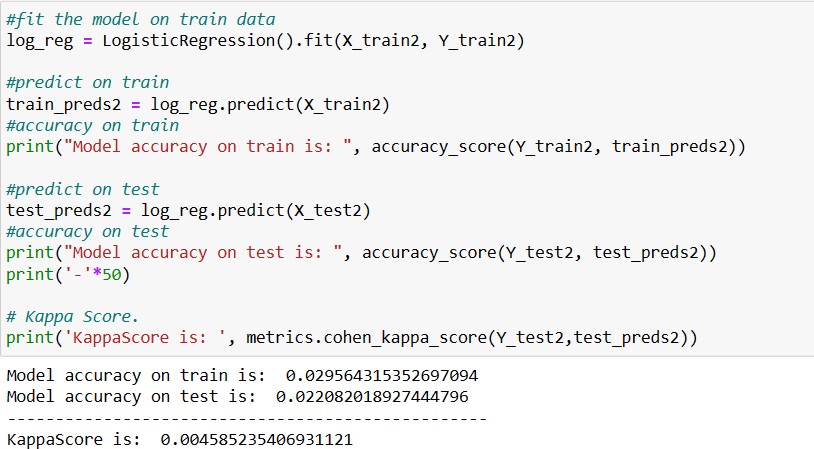


# MACHINE LEARNING ALGORITHMS

Machine learning algorithms are computational models that enable computers to learn from data and make predictions or decisions without being explicitly programmed for each specific task. There are several types of machine learning algorithms, each designed for different types of tasks and data.

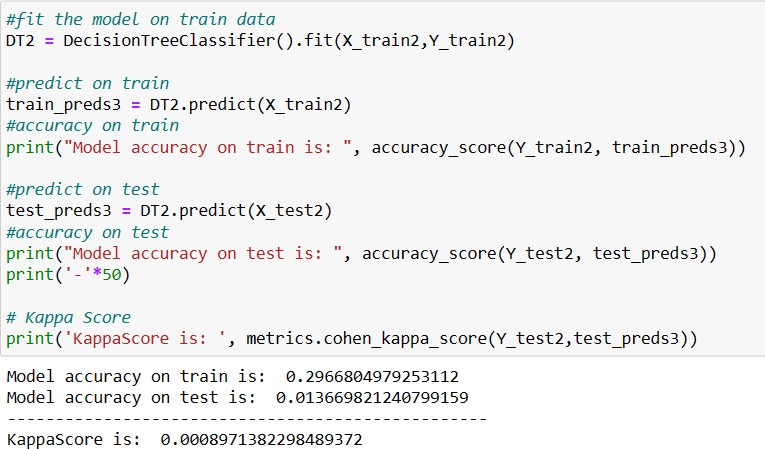
**LOGISTIC REGRESSION**

Logistic regression is a supervised machine learning algorithm mainly used for classification tasks where the goal is to predict the probability that an instance of belonging to a given class or not. It is a kind of statistical algorithm, which analyze the relationship between a set of independent variables and the dependent binary variables.



# DECISION TREE CLASSIFIER

Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. Decision trees are used for both classification and regression tasks. They make decisions by asking a series of questions based on input features and ultimately lead to a predicted outcome.

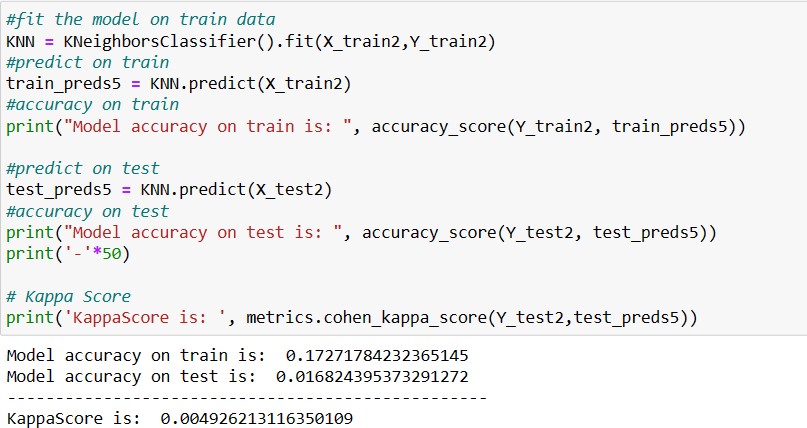


It has the kappa score of 0.0008971.

# K-NEAREST NEIGHBORS

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.

K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.K-NN algorithm stores all the available data and classifies a new data point based on the similarity.

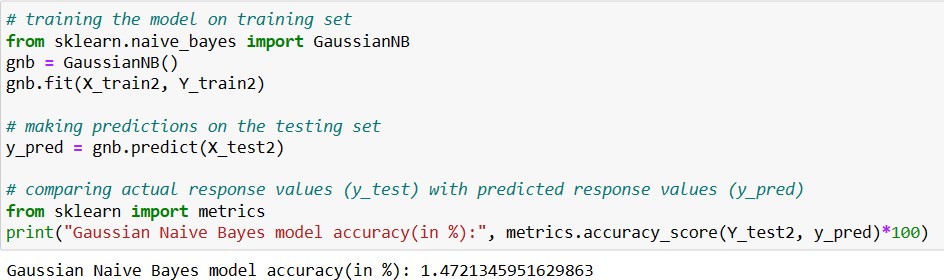


It has the kappa score of 0.0004926. Using the k\_means we can able to predict the accuracy.

# NAÏVE BAYES CLASSIFIER

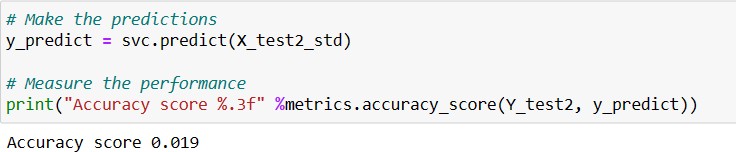
Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.

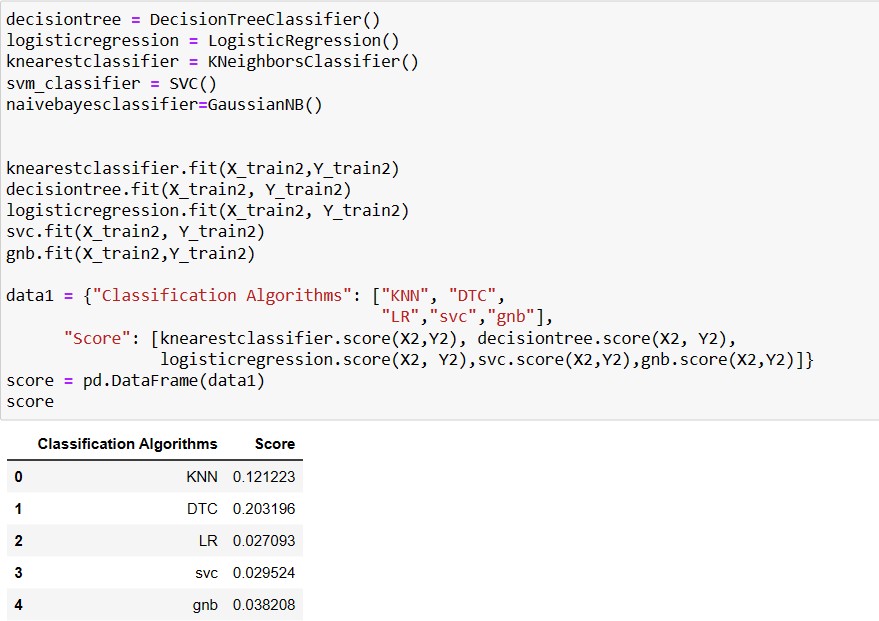


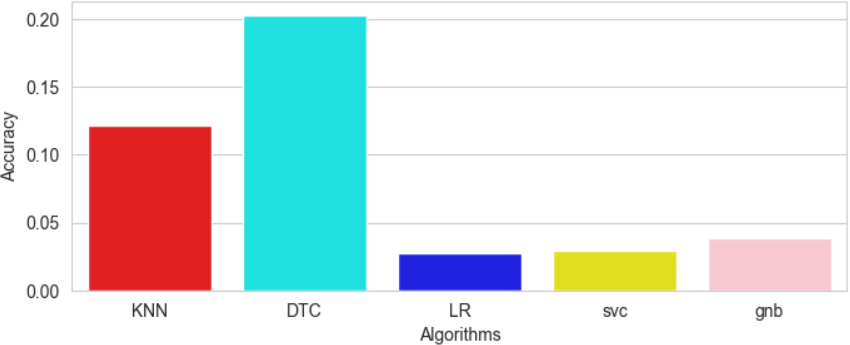
# SUPPORT VECTOR MACHINE

Support Vector Machine (SVM) is a powerful machine learning algorithm used for linear or nonlinear classification, regression, and even outlier detection tasks. SVMs can be used for a variety of tasks, such as text classification, image classification, spam detection, handwriting identification, gene expression analysis, face detection, and anomaly detection.



# MODEL COMPARISONS





From this we can take the decision tree classifier model to predict the rspm values. Because it gives more accuracy than the other.

# PYTHON CODING TO PREDICT THE RSPM VALUES BASED ON SO2 AND NO2 VALUES

import pandas as pd import joblib

from sklearn.model\_selection import train\_test\_split from sklearn.tree import DecisionTreeClassifier import warnings

warnings.filterwarnings("ignore")

# Load your dataset (CSV file or any other format)

# Assuming your dataset has columns 'SO2', 'NO2', and 'RSPM' # Read your dataset

b = pd.read\_csv('D:\\desh.csv')

# Extract features (SO2 and NO2) and target (RSPM) X = b[['SOi', 'Noi']]

Y = b['rpi']

# Split the data into training and testing sets

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y, test\_size=0.2, random\_state=42)

# Create a decision tree classifier model and train it model=DecisionTreeClassifier()

model.fit(X\_train,Y\_train) # Save the model to a file

joblib.dump(model,'rspm\_prediction\_model.pkl') # Load the trained model model=joblib.load('rspm\_prediction\_model.pkl') def predict\_rspm(so2, no2):

# Make a prediction using the loaded model prediction = model.predict([[so2, no2]]) return prediction[0]

if name == ' main ': # Get input from the user try:

so2 = float(input("Enter SO2 value: ")) no2 = float(input("Enter NO2 value: "))

# Predict RSPM value

predicted\_rspm = predict\_rspm(so2, no2) print("Predicted RSPM value:", predicted\_rspm)

except ValueError:

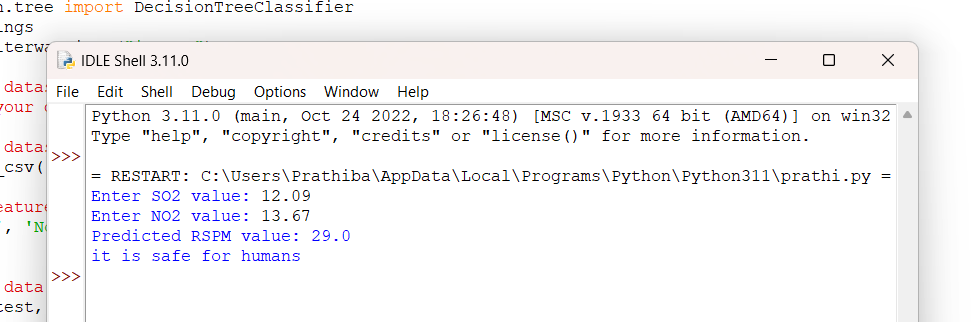
print("Invalid input. Please enter numeric values for SO2 and NO2.")

if(predicted\_rspm<50):

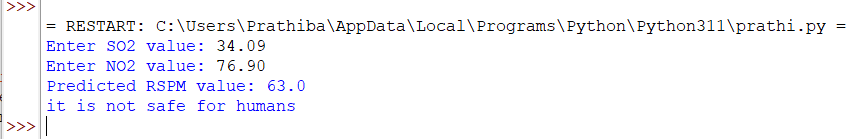
print("it is safe for humans") else:

print("it is not safe for humans")

# OUTPUT



The given values gives us the rspm values is safe for humans.



The given values gives us the rspm value is unsafe for humans.

# CONCLUSION



The rspm value was predicted based on s02 and no2 using the decision tree classifier model which can also shows that it was healthy or unhealthy to the humans.