**KGiSL Institute Of Technology**

**NAAN MUDHALVAN**

***Project Title*** *:*

## Air Quality Analysis In TN

***Team Members :***

1. Nimisha.p.s

2. Mirthika.s

3. Eben F Gabriel

4. Vigneshwaran.p

***Problem Definition:***

## Air Quality Analysis In TN

***Objectives:***

***Steps:***

IN GOOGLE COLAB NOTEBOOK:

• Mount the GoogleDrive

• Loading the Dataset to GoogleDrive

• Processing and cleansing the dataset

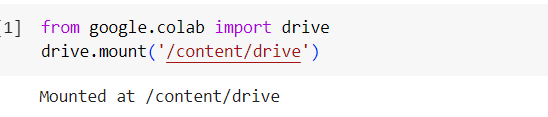
• Accuracy

***Code:***

from google.colab import drive

# This will prompt you to click a link and generate an authentication code drive.mount('/content/gdrive')

***Output:***



***#LOADING THE DATASET:***

***Code:***

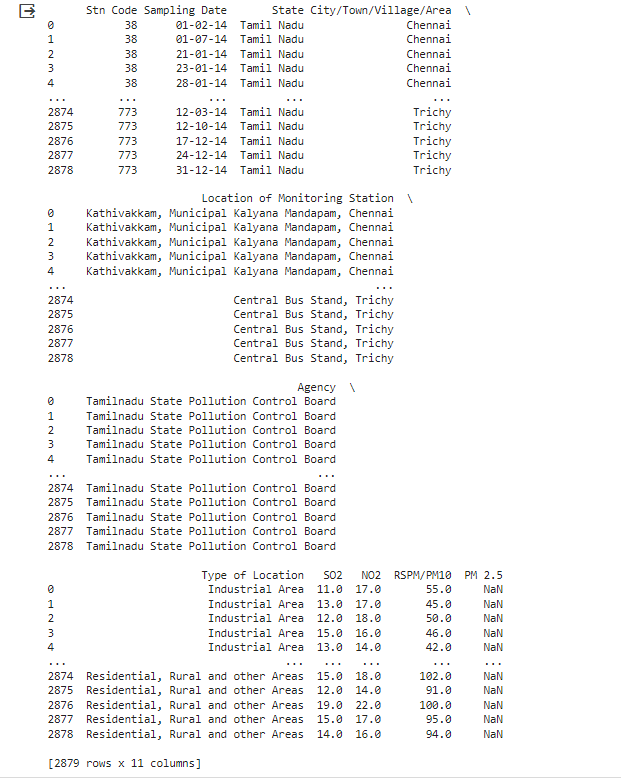
import pandas as pd

# Load your dataset

phase\_3 = '/content/gdrive/MyDrive/Colab Notebooks/statsfinal.csv'

data = pd.read\_csv(phase\_3)

***Output:***

******

***#PROCESSING AND CLEANSING THE DATASET:***

***Code:***

import pandas as pd

import matplotlib.pyplot as plt

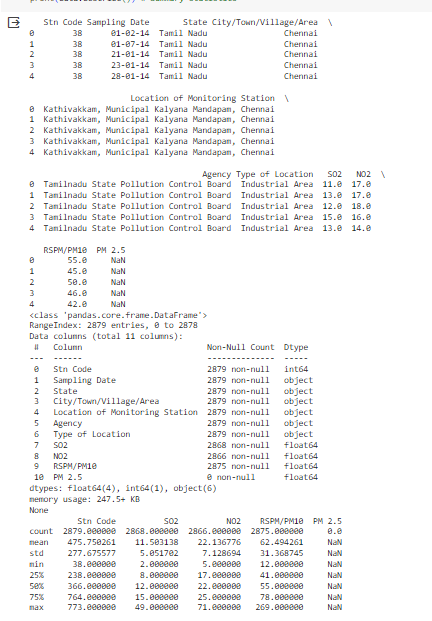
import seaborn as sns

print(data.head()) # Display the first few rows of the dataset

print(data.info()) # Display data types and non-null counts

print(data.describe()) # Summary statistics

***Output:***

******

***#CLEANSING THE DATA:***

***Code:***

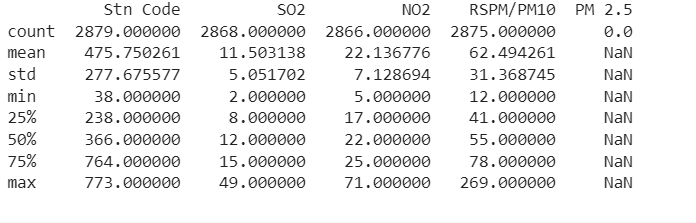
import pandas as pd

data = data.drop\_duplicates()

data.to\_csv("cleaned\_dataset.csv", index=False)

print(data.describe())

***Output:***



***#ANALYSIS ON DATASET:***

***Code:***

plt.figure(figsize=(10, 6))

sns.barplot(x='Q-P1', y='S-P1', data=data)

plt.title("Sales by Product Category")

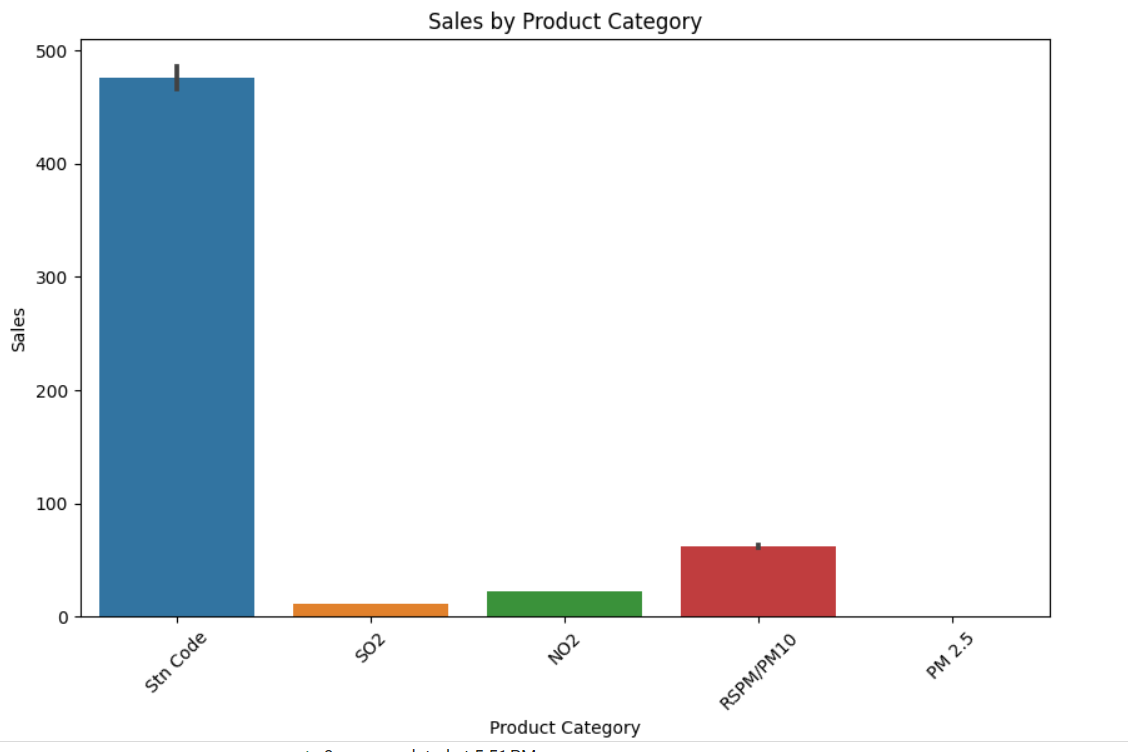
plt.xlabel("Product Category")

plt.ylabel("Sales")

plt.xticks(rotation=45)

plt.show()

***Output:***



***#LOGICAL REGRESSION:***

***Code:***

from sklearn.datasets import make\_classification

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

from sklearn.metrics import accuracy\_score

from sklearn.linear\_model import LogisticRegression

nb\_samples = 1000

x, y = make\_classification(n\_samples=nb\_samples, n\_features=2,

n\_informative=2, n\_redundant=0, n\_clusters\_per\_class=1)

xtrain, xtest, ytrain, ytest = train\_test\_split(x, y, test\_size=0.2, random\_state=42)

model = LogisticRegression()

model.fit(xtrain, ytrain)

***Output:***



***#ACCURACY:***

***Code:***

print(accuracy\_score(ytest, model.predict(xtest)))

***Output:***

