## **Program for Time Series Data Cleaning, Loading, Handling, and Pre-processing Techniques**

**1. Importing Libraries**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

pandas: Handles the dataset as a DataFrame for structured analysis

numpy: Supports numerical computations.

matplotlib and seaborn: Create visualizations to analyze trends and patterns.

**2. Loading the Dataset**

df = pd.read\_csv("D:/221501124/OzoneHole\_Data.csv")

Loads the ozone dataset from a CSV file using pandas.

The file path (D:/221501124/OzoneHole\_Data.csv) must point to a valid file containing ozone data with columns such as Year, Hole Area, and Minimum Ozone.

**3. Viewing Dataset Structure and Summary Statistics**

print(df.info())

print(df.describe())

**df.info()**:

Displays the structure of the dataset, including column names, data types, and non-null counts.

Confirms that all 42 rows contain non-null values for Year, Hole Area, and Minimum Ozone.

**df.describe()**:

Year: Range is from 1979 to 2021.

Hole Area: Mean of ~18.44 with a standard deviation of ~7.41.

Minimum Ozone: Mean of ~131.14, indicating average ozone concentration levels.

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

sns.lineplot(x='Year', y='Hole Area', data=df, label='Hole Area')

sns.lineplot(x='Year', y='Minimum Ozone', data=df, label='Minimum Ozone')

plt.title('Ozone Hole Area and Minimum Ozone Levels Over Time')

plt.xlabel('Year')

plt.ylabel('Value')

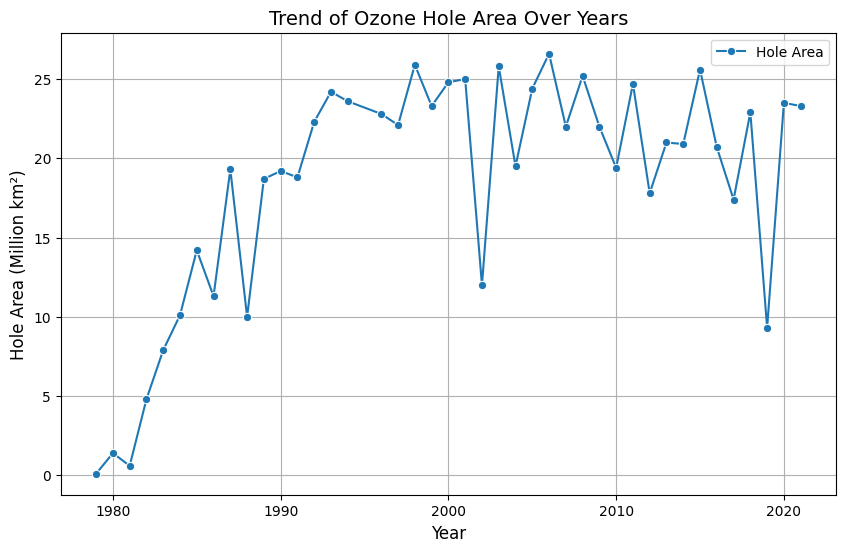
plt.legend()

plt.show()

**4.Visualizing Time-Series Trends**

Visualizes trends for Hole Area and Minimum Ozone over the years

* Hole Area likely shows an increasing trend during certain periods, indicating ozone depletion.
* Minimum Ozone levels might exhibit a declining trend, reflecting reduced ozone concentration.



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

sns.lineplot(x="Year", y="Minimum Ozone", data=df, marker='o', color='orange', label='Minimum Ozone')

plt.title("Trend of Minimum Ozone Levels Over Years", fontsize=14)

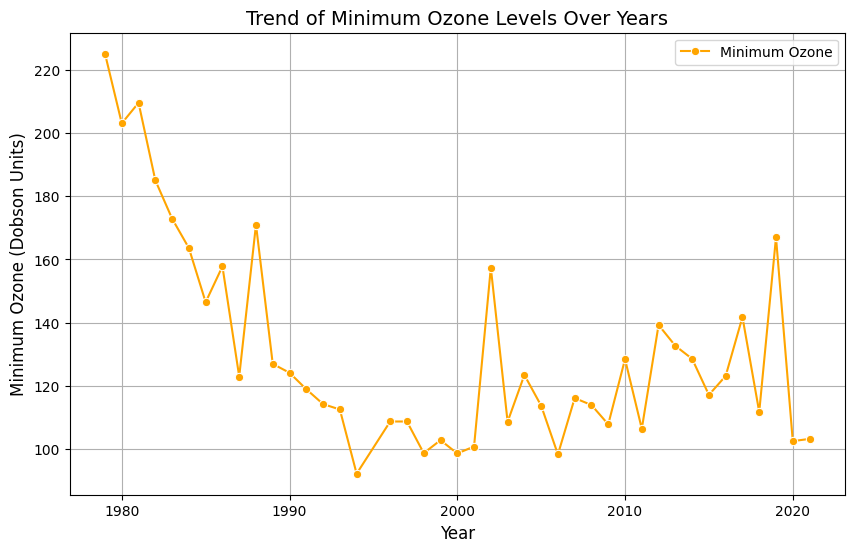
plt.xlabel("Year", fontsize=12)

plt.ylabel("Minimum Ozone (Dobson Units)", fontsize=12)

plt.legend()

plt.grid(True)

plt.show()



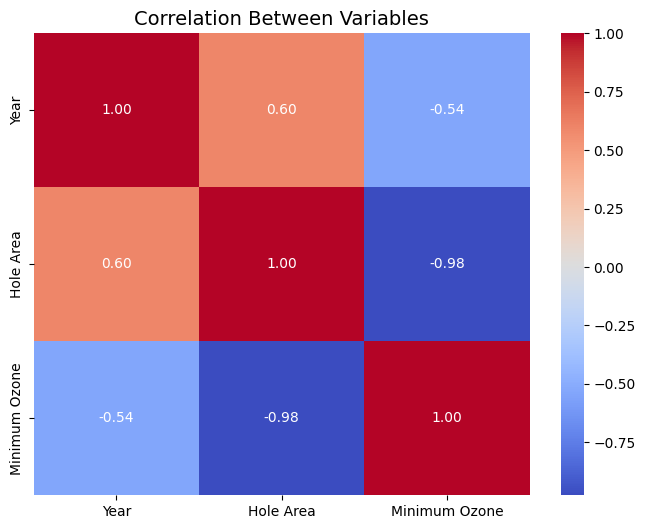
**5.Exploring Correlation Between Variables**

plt.figure(figsize=(8, 6))

correlation\_matrix = df.corr()

sns.heatmap(correlation\_matrix, annot=True, cmap="coolwarm", fmt=".2f")

plt.title("Correlation Between Variables", fontsize=14)



**6.Result**

Thus successfully implemented a program related to ozone depletion using machine learning