**COVID-19 CASE ANALYSIS**

**Data Collection:**

Obtain the COVID-19 case data from reliable sources, such as government health departments, the World Health Organization (WHO), or datasets provided by reputable organizations.

**Data Loading:**

Load the data into a data analysis tool like Python with Pandas or R.

**Data Inspection:**

Use Pandas to inspect the data, checking for the structure, data types, and the first few rows.

**Handling Missing Values:**

Identify and handle missing values, which may include removing rows with missing data or imputing values.

**Data Type Conversion:**

Ensure that data types are appropriate for analysis. Convert data types as needed.

**Data Cleaning:**

Clean the data by addressing inconsistencies, outliers, and errors. This might involve removing duplicates, correcting data entries, or filtering the dataset.

**Feature Engineering:**

Create new features or modify existing ones to extract valuable information for analysis.

**Data Aggregation:**

Aggregate data when necessary to summarize information over time, by location, or by other relevant factors.

**Data Transformation:**

Apply transformations, such as scaling, normalization, or log transformations, if required for the analysis.

**Data Visualization:**

Create data visualizations to explore the data and detect patterns or anomalies.

**Data Export:**

Save the pre processed data for analysis in a new file if needed.

**SOURCE CODE**

import pandas as pd

import matplotlib.pyplot as plt

# Read the data into a Pandas DataFrame

data = pd.read\_csv('/content/Covid\_19\_cases4.csv')

# Ensure the date column is in datetime format

data['dateRep'] = pd.to\_datetime(data['dateRep'], format='%d-%m-%Y')

# Sort the data by date

data = data.sort\_values(by='dateRep')

# Plot daily cases and deaths

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

plt.plot(data['dateRep'], data['cases'], label='Daily Cases', color='b')

plt.plot(data['dateRep'], data['deaths'], label='Daily Deaths', color='r')

plt.xlabel('Date')

plt.ylabel('Count')

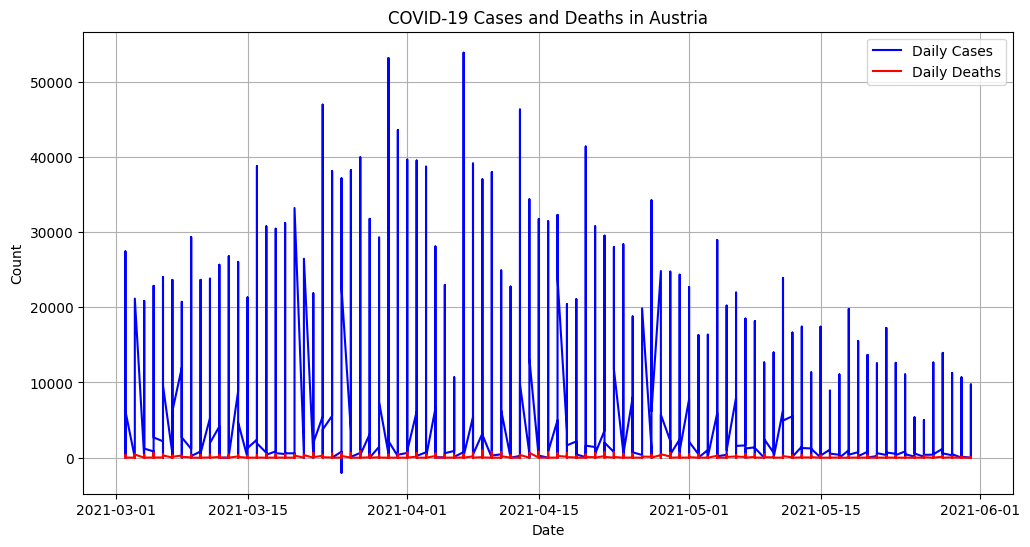
plt.title('COVID-19 Cases and Deaths in Austria')

plt.legend()

plt.grid(True)

plt.show()

**OUTPUT**

****