Exploratory Data Analysis on Most Polluted Countries

Problem Statement Exploratory Data Analysis on Most Polluted Countries

In this Jupyter project, I conducted a comprehensive Exploratory Data Analysis (EDA) on a dataset named "Most Polluted Countries." The dataset encompasses crucial information such as pollution levels, growth rates, geographical details, and rankings of various countries.

Key Highlights:

Data Overview: I began by loading and exploring the dataset, providing a snapshot of its structure and summary statistics. Additionally, I checked for missing values to ensure data integrity.

Visualizations: Leveraging Python libraries like Matplotlib and Seaborn, I created insightful visualizations to uncover patterns and trends within the dataset. This included histograms to showcase the distribution of pollution levels, box plots to analyze pollution growth rates across regions, and a correlation heatmap to identify relationships between variables.

Answering Questions: I addressed specific questions such as identifying the topmost polluted countries in 2023 and examining the relationship between land area and pollution density.

Insights: Throughout the analysis, I gained valuable insights into the distribution of pollution levels, regional variations in pollution growth rates, and correlations between different factors.

This project not only demonstrates my proficiency in Python for data analysis but also showcases my ability to derive meaningful insights from complex datasets. The visualizations and code snippets provide a clear narrative, making it accessible to both technical and non-technical audiences.

Import Library

In [1]:

```python
import pandas as pd
```

In [2]:

```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import seaborn as sns
```

```
C:\Users\Syed Arif\anaconda3\lib\site-packages\scipy\__init__.py:146: UserWar
ning: A NumPy version >=1.16.5 and <1.23.0 is required for this version of Sc
iPy (detected version 1.25.1
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}"
```

Uploading Csv fle

In [3]:

```
df = pd.read_csv(r"C:\Users\Syed Arif\Desktop\most-polluted-countries.csv")
```

Data Preprocessing

.head()

head is used show to the By default = 5 rows in the dataset

In [4]:

```
df.head()
```

Out[4]:

	pollution_2023	pollution_growth_Rate	country_name	ccn3	country_region	united_nation_Men
0	1428627663	0.00808	India	356	Asia	
1	1425671352	-0.00015	China	156	Asia	
2	339996563	0.00505	United States	840	North America	
3	277534122	0.00738	Indonesia	360	Asia	
4	240485658	0.01976	Pakistan	586	Asia	

.tail()

tail is used to show rows by Descending order

In [5]:
```
df.tail()
```

Out[5]:

	pollution_2023	pollution_growth_Rate	country_name	ccn3	country_region	united_nation_Me
91	704149	0.01292	Macau	446	Asia	
92	654768	0.01107	Luxembourg	442	Europe	
93	535064	0.00333	Malta	470	Europe	
94	412623	0.00644	Bahamas	44	North America	
95	375318	0.00649	Iceland	352	Europe	

.shape

It show the total no of rows & Column in the dataset

In [6]:
```
df.shape
```

Out[6]:
```
(96, 12)
```

.Columns

It show the no of each Column

In [7]:
```
df.columns
```

Out[7]:
```
Index(['pollution_2023', 'pollution_growth_Rate', 'country_name', 'ccn3',
       'country_region', 'united_nation_Member', 'country_land_Area_in_Km',
       'pollution_density_in_km', 'pollution_density_per_Mile',
       'share_borders', 'pollution_Rank',
       'mostPollutedCountries_particlePollution'],
      dtype='object')
```

.dtypes

This Attribute show the data type of each column

```
In [8]: df.dtypes
```

```
Out[8]: pollution_2023                                int64
        pollution_growth_Rate                       float64
        country_name                                 object
        ccn3                                          int64
        country_region                               object
        united_nation_Member                           bool
        country_land_Area_in_Km                     float64
        pollution_density_in_km                     float64
        pollution_density_per_Mile                  float64
        share_borders                                object
        pollution_Rank                                int64
        mostPollutedCountries_particlePollution     float64
        dtype: object
```

.unique()

In a column, It show the unique value of specific column.

```
In [9]: df["country_region"].unique()
```

```
Out[9]: array(['Asia', 'North America', 'Africa', 'South America', 'Europe',
               'Oceania'], dtype=object)
```

.nuique()

It will show the total no of unque value from whole data frame

```
In [10]: df.nunique()
```

```
Out[10]: pollution_2023                             96
         pollution_growth_Rate                      95
         country_name                               96
         ccn3                                       96
         country_region                              6
         united_nation_Member                        2
         country_land_Area_in_Km                    96
         pollution_density_in_km                    96
         pollution_density_per_Mile                 96
         share_borders                              83
         pollution_Rank                             96
         mostPollutedCountries_particlePollution    93
         dtype: int64
```

.describe()

It show the Count, mean , median etc

In [11]:

```
df.describe()
```

Out[11]:

	pollution_2023	pollution_growth_Rate	ccn3	country_land_Area_in_Km	pollution_der
count	9.600000e+01	96.000000	96.000000	9.600000e+01	
mean	7.405002e+07	0.007062	402.822917	1.088409e+06	
std	2.083376e+08	0.013354	251.466687	2.518835e+06	2
min	3.753180e+05	-0.074480	4.000000	3.290000e+01	
25%	5.881984e+06	0.001303	190.250000	6.213750e+04	
50%	1.976120e+07	0.006790	386.000000	2.304400e+05	
75%	5.565119e+07	0.012140	617.000000	7.740505e+05	
max	1.428628e+09	0.049800	860.000000	1.637687e+07	21

.value_counts

It Shows all the unique values with their count

In [12]:

```
df["country_region"].value_counts()
```

Out[12]:

```
Asia             37
Europe           35
Africa            9
North America     7
South America     6
Oceania           2
Name: country_region, dtype: int64
```

.isnull()

It shows the how many null values

In [13]:
```
df.isnull()
```

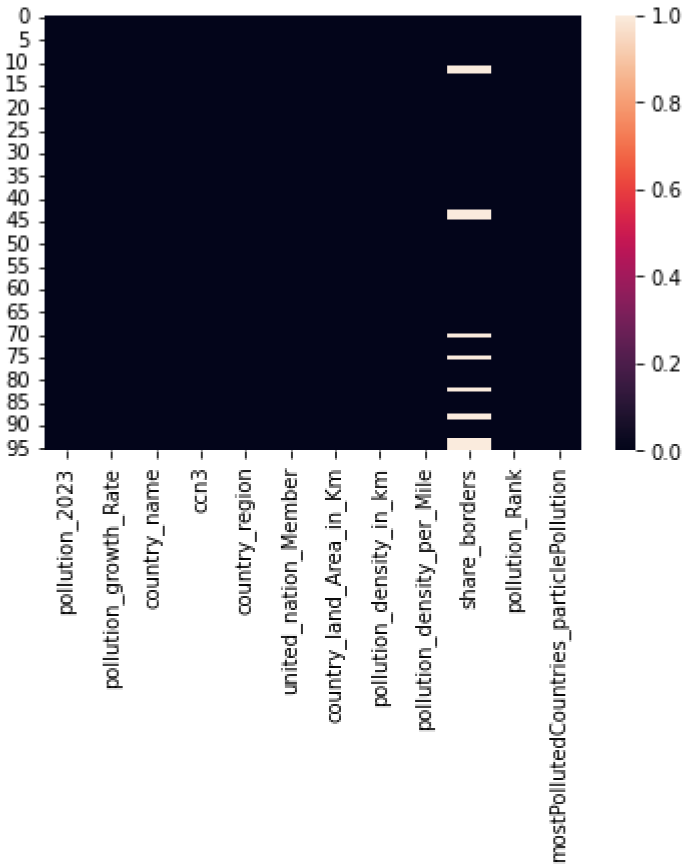
Out[13]:

	pollution_2023	pollution_growth_Rate	country_name	ccn3	country_region	united_nation_Me
0	False	False	False	False	False	
1	False	False	False	False	False	
2	False	False	False	False	False	
3	False	False	False	False	False	
4	False	False	False	False	False	
...	...	...	...	...	...	
91	False	False	False	False	False	
92	False	False	False	False	False	
93	False	False	False	False	False	
94	False	False	False	False	False	
95	False	False	False	False	False	

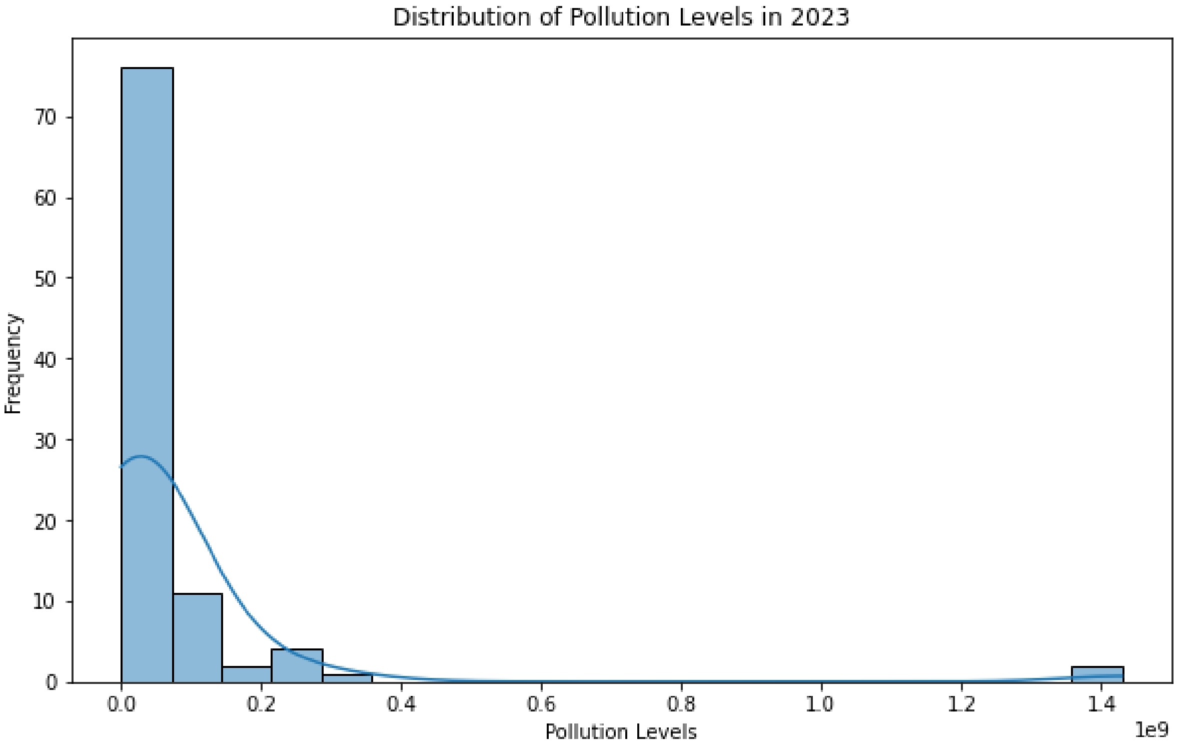
96 rows × 12 columns

In [14]:
```
sns.heatmap(df.isnull())
```

Out[14]:
```
<AxesSubplot:>
```

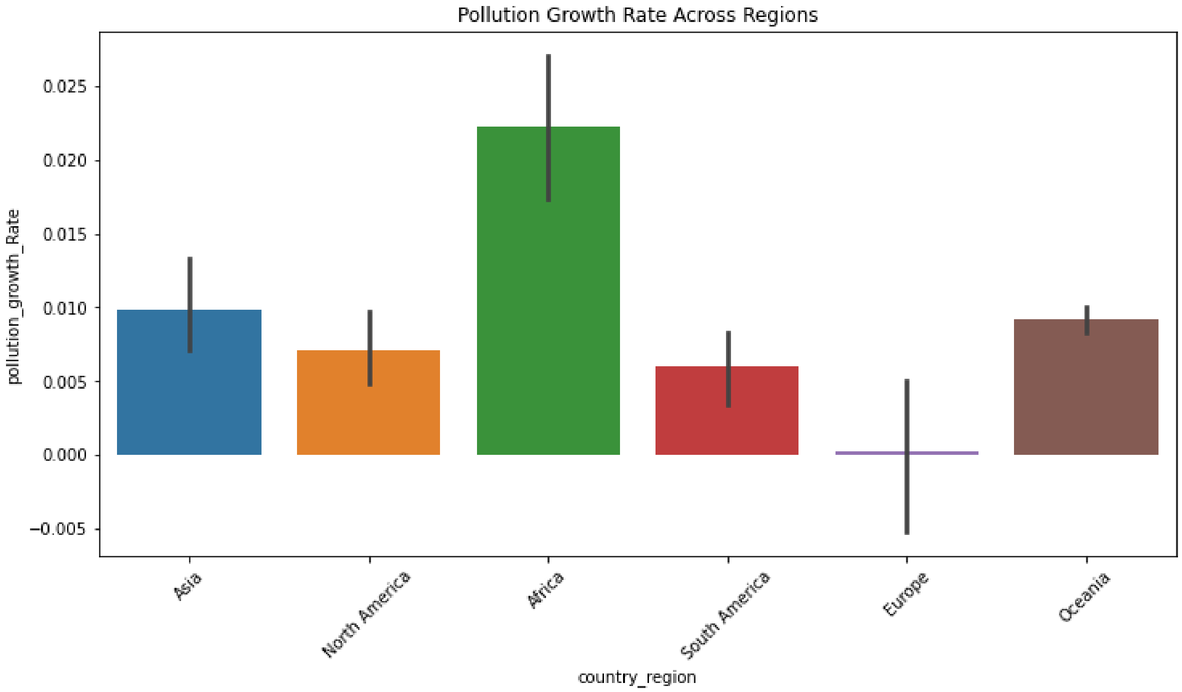


In [15]:
```python
plt.figure(figsize=(10, 6))
sns.histplot(df['pollution_2023'], bins=20, kde=True)
plt.title('Distribution of Pollution Levels in 2023')
plt.xlabel('Pollution Levels')
plt.ylabel('Frequency')
plt.show()
```



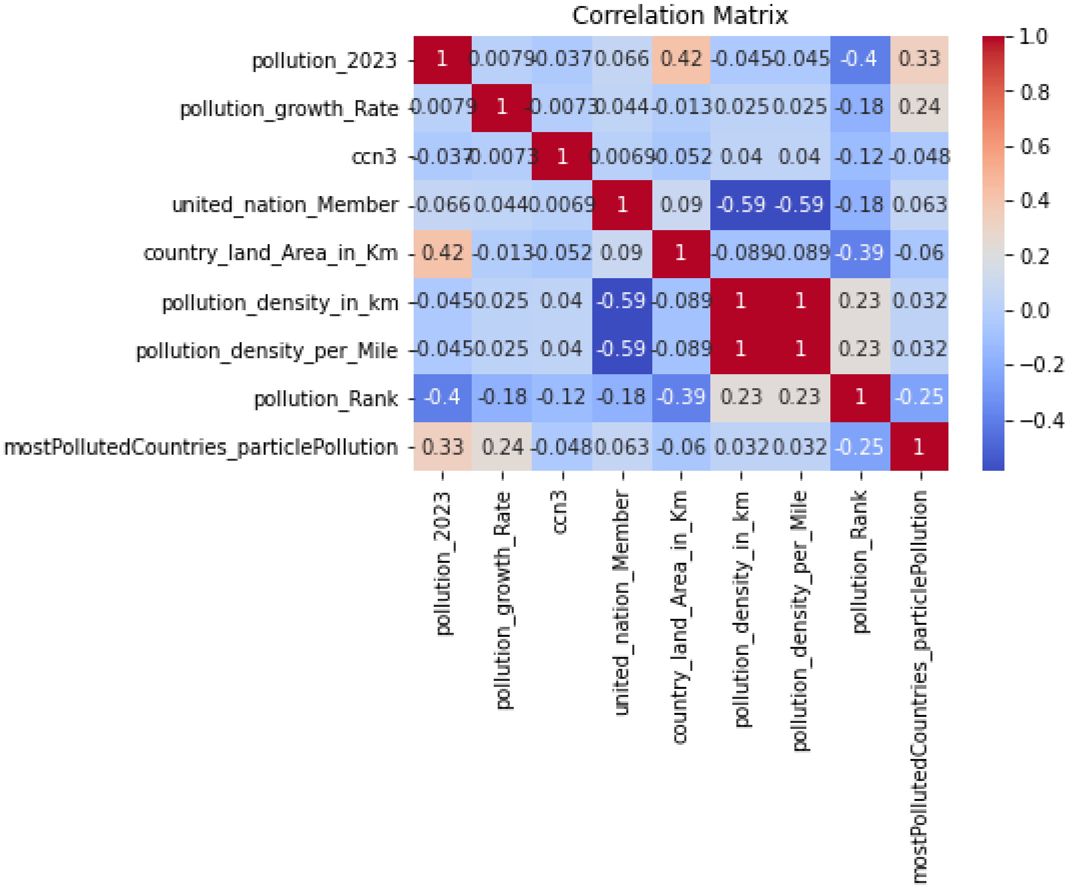
In [17]:

```
plt.figure(figsize=(12, 6))
sns.barplot(x='country_region', y='pollution_growth_Rate', data=df)
plt.title('Pollution Growth Rate Across Regions')
plt.xticks(rotation=45)
plt.show()
```



```
In [20]: # Correlation matrix
         correlation_matrix = df.corr()

         # Visualize the correlation matrix
         sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
         plt.title('Correlation Matrix')
         plt.show()
```



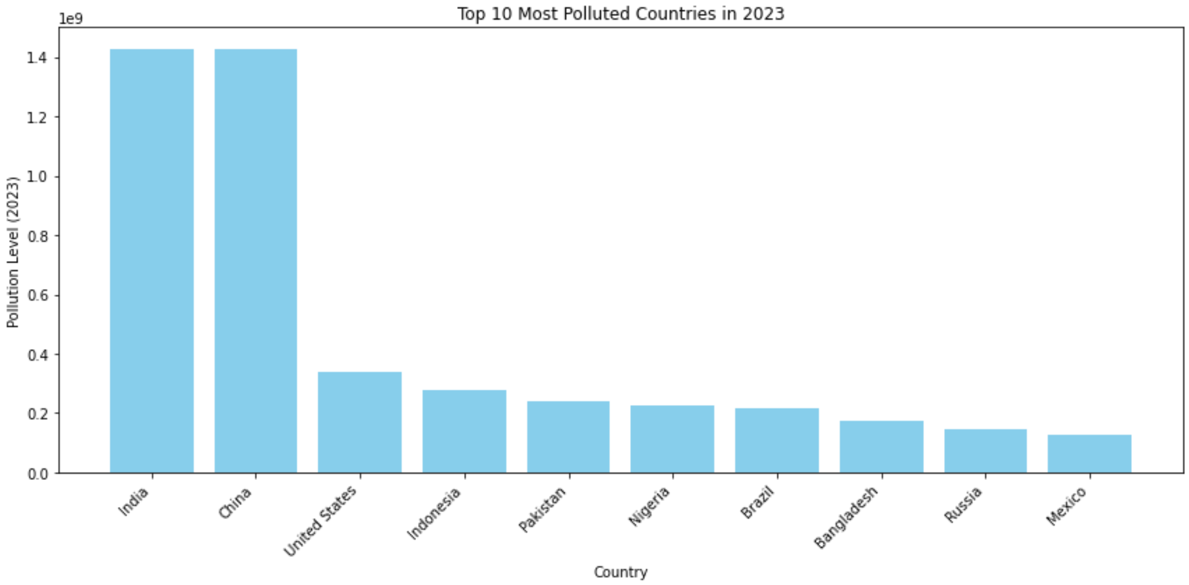
```
In [21]: most_polluted_countries = df.sort_values('pollution_2023', ascending=False)['c
         print("Top 10 Most Polluted Countries in 2023:")
         print(most_polluted_countries)
```

```
Top 10 Most Polluted Countries in 2023:
0            India
1            China
2    United States
3        Indonesia
4         Pakistan
5          Nigeria
6           Brazil
7       Bangladesh
8           Russia
9           Mexico
Name: country_name, dtype: object
```

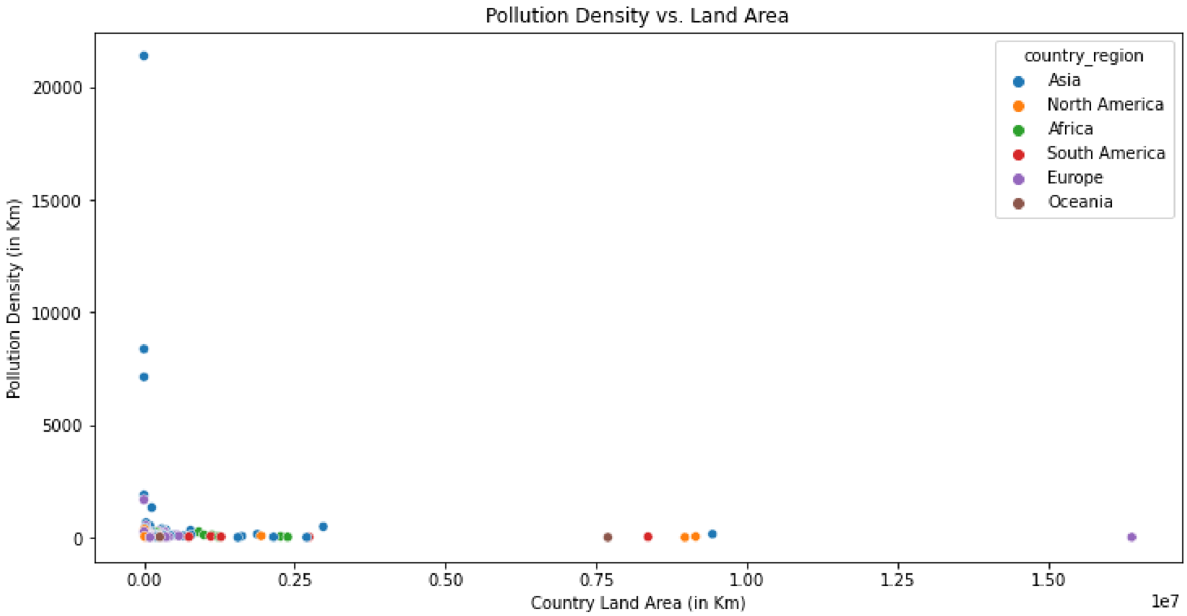
In [30]:

```
most_polluted_countries = df.sort_values('pollution_2023', ascending=False).hea

# Create a bar plot
plt.figure(figsize=(12, 6))
plt.bar(most_polluted_countries['country_name'], most_polluted_countries['pollu
plt.title('Top 10 Most Polluted Countries in 2023')
plt.xlabel('Country')
plt.ylabel('Pollution Level (2023)')
plt.xticks(rotation=45, ha='right')  # Rotate country names for better visibil
plt.tight_layout()
P
# Show the plot
plt.show()
```



In [22]:
```
plt.figure(figsize=(12, 6))
sns.scatterplot(x='country_land_Area_in_Km', y='pollution_density_in_km', hue=
plt.title('Pollution Density vs. Land Area')
plt.xlabel('Country Land Area (in Km)')
plt.ylabel('Pollution Density (in Km)')
plt.show()
```


Pollution Density vs. Land Area
Pollution Density (in Km)
1e7
Country Land Area (in Km)
country_region
● Asia
● North America
● Africa
● South America
● Europe
● Oceania