Data Collection and Preprocessing Phase

Date	13 June 2024
Team ID	SWTID1749709340
Project Title	Predicting Co2 Emission by countries Using Machine Learning
Maximum Marks	6 Marks

Data Exploration and Preprocessing Template

Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

Section	**Description**
Data Overview	Basic statistics, dimensions, and structure of the data.
Univariate Analysis	Exploration of individual variables (mean, median, mode, etc.).
Bivariate Analysis	Relationships between two variables (correlation, scatter plots).
Multivariate Analysis	Patterns and relationships involving multiple variables.
Outliers and Anomalies	Identification and treatment of outliers.
Data Preprocessing Code Screenshots	
Loading Data	Code to load the dataset into the preferred environment (e.g., Python, R).

Handling Missing Data	Code for identifying and handling missing values.

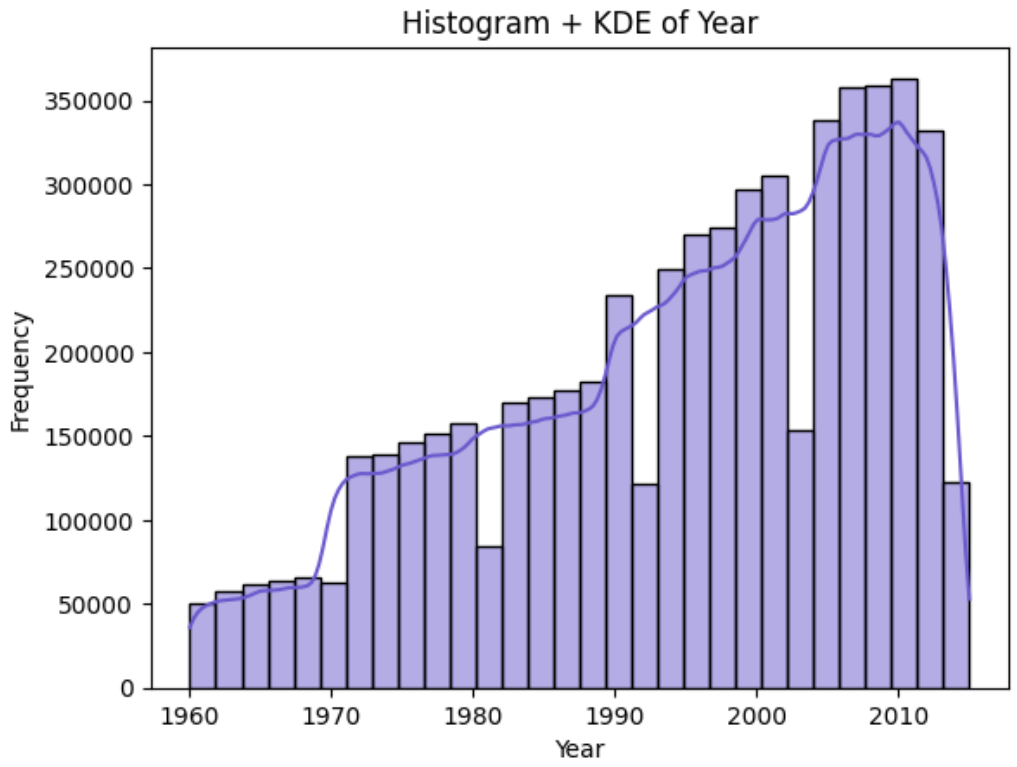
DATA OVERVIEW:

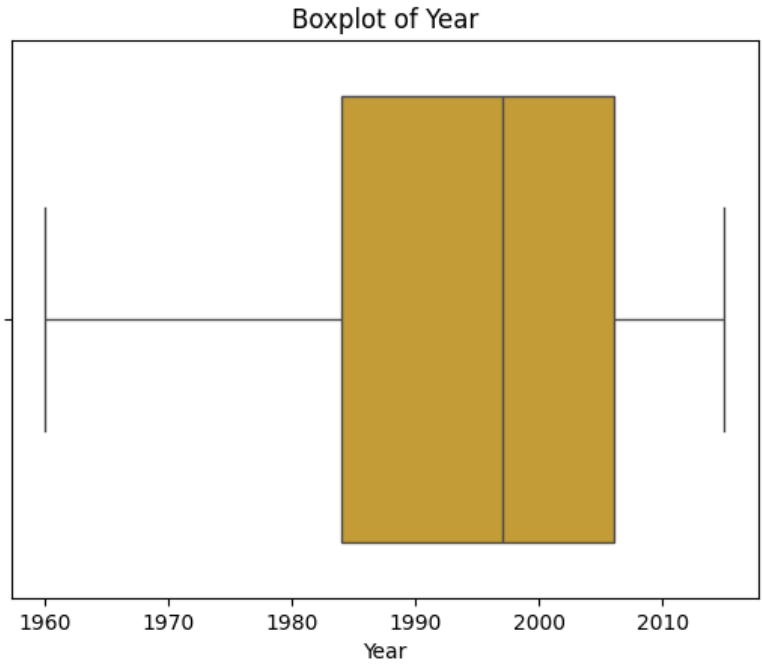
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5656458 entries, 0 to 5656457
Data columns (total 6 columns):
 #   Column         Dtype
---  ------         -----
 0   CountryName    object
 1   CountryCode    object
 2   IndicatorName  object
 3   IndicatorCode  object
 4   Year           int64
 5   Value          float64
dtypes: float64(1), int64(1), object(4)
memory usage: 258.9+ MB

Duplicates: 0

Missing values per column:
 Series([], dtype: int64)
```

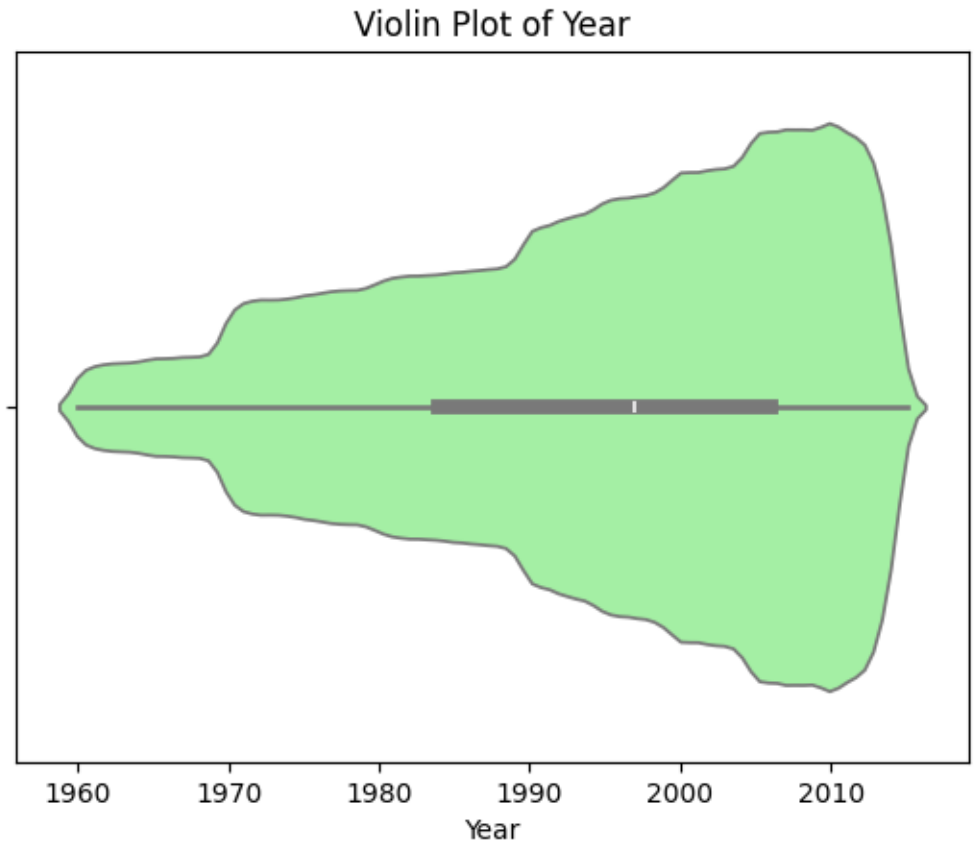
	Year	Value
count	5.656458e+06	5.656458e+06
mean	1.994464e+03	1.070501e+12
std	1.387895e+01	4.842469e+13
min	1.960000e+03	-9.824821e+15
25%	1.984000e+03	5.566242e+00
50%	1.997000e+03	6.357450e+01
75%	2.006000e+03	1.346722e+07
max	2.015000e+03	1.103367e+16

UNIVARIATE ANALYSIS:

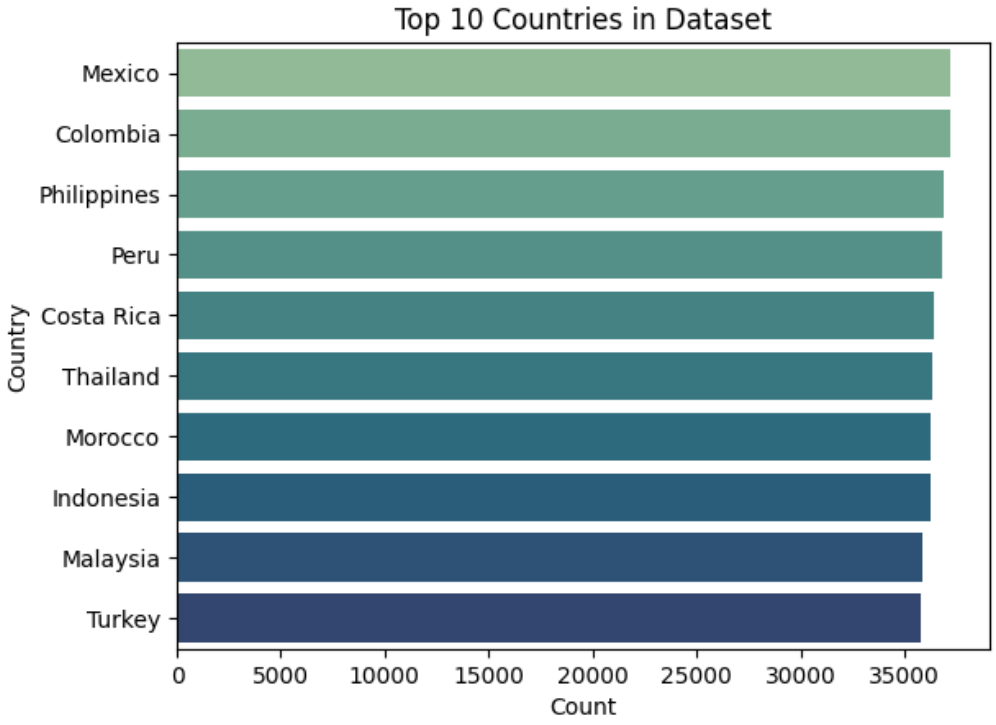
FOR NUMERICAL DATA:

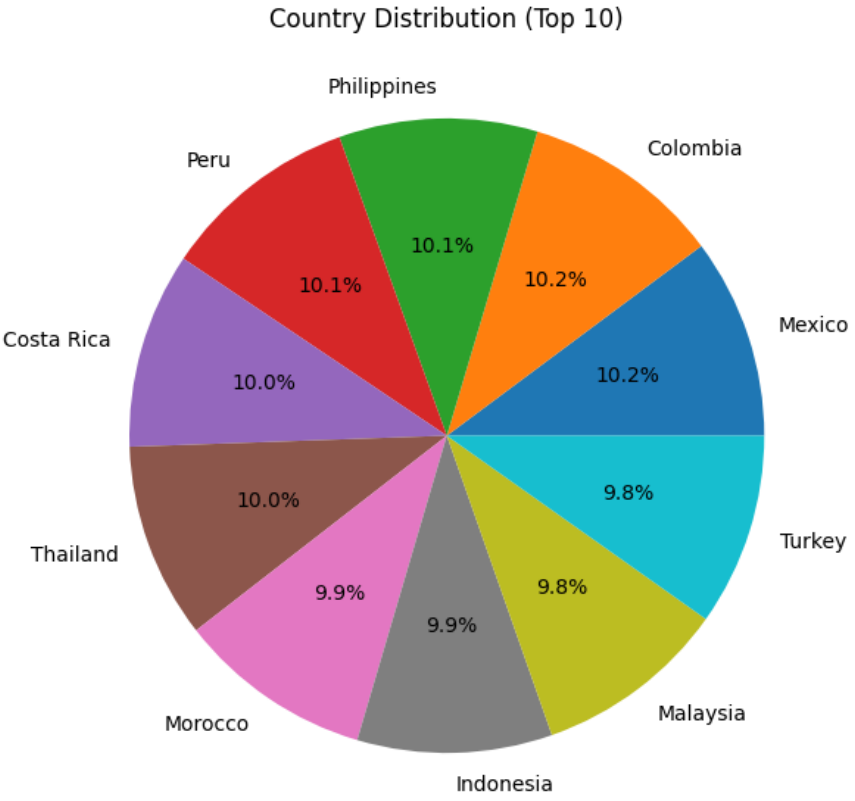


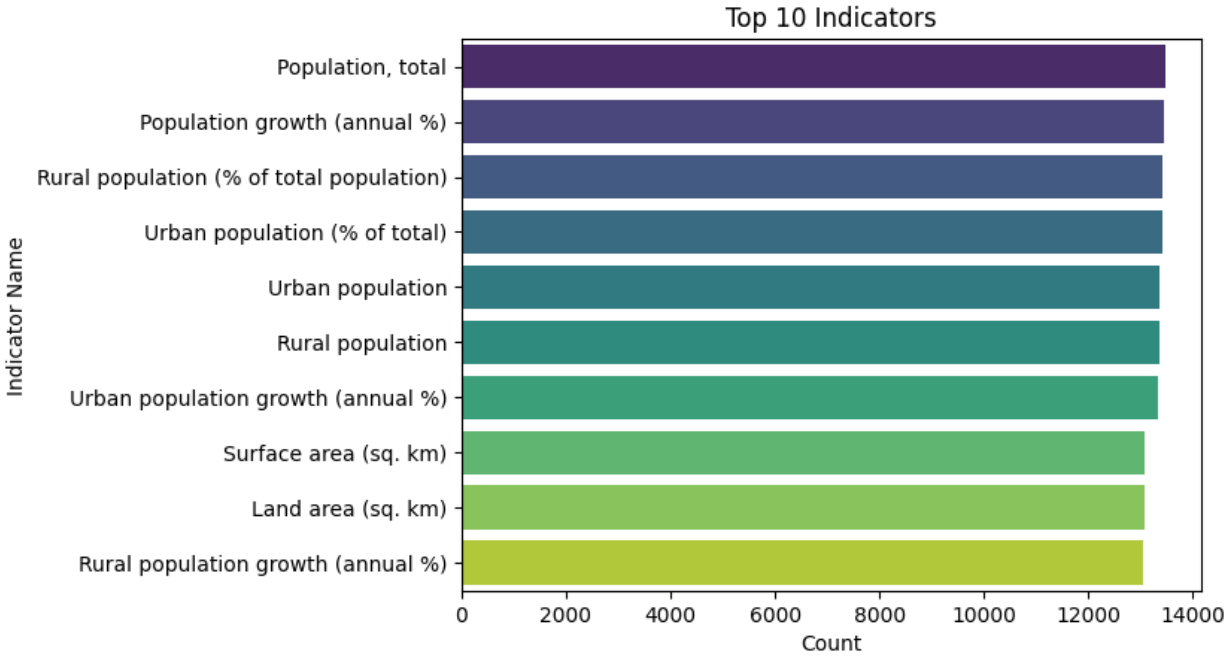


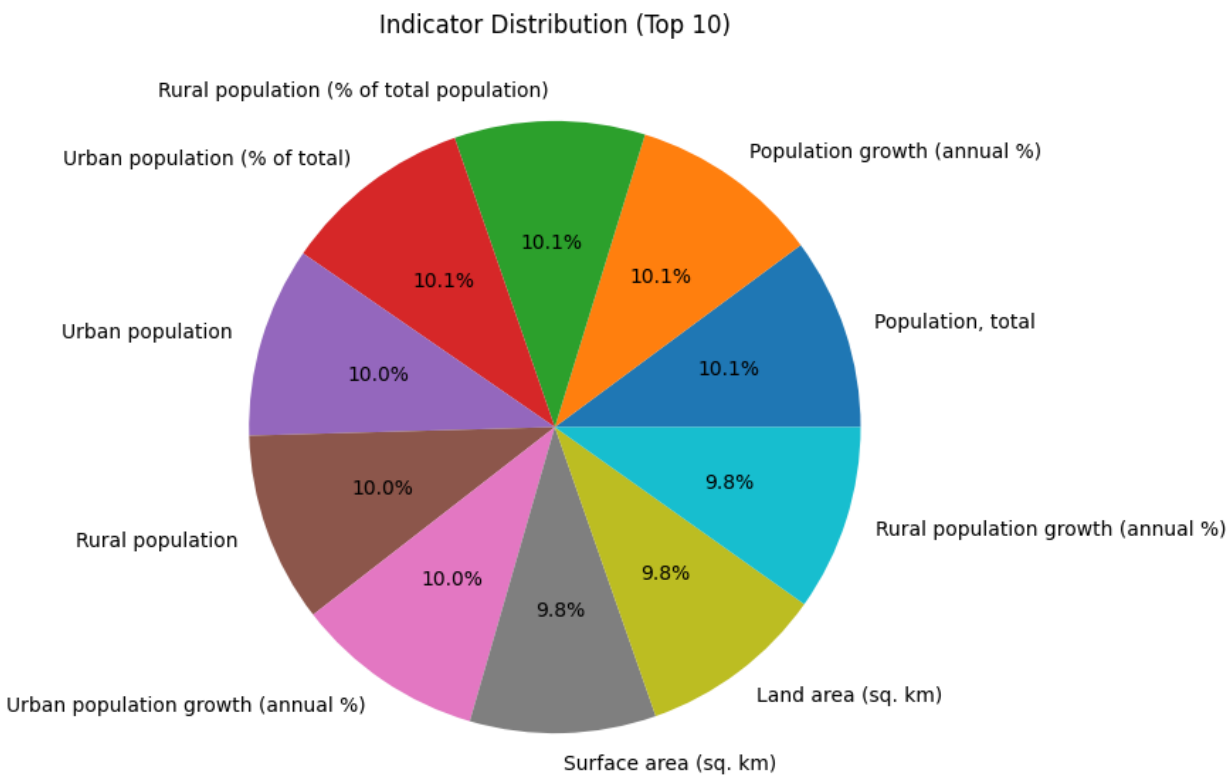


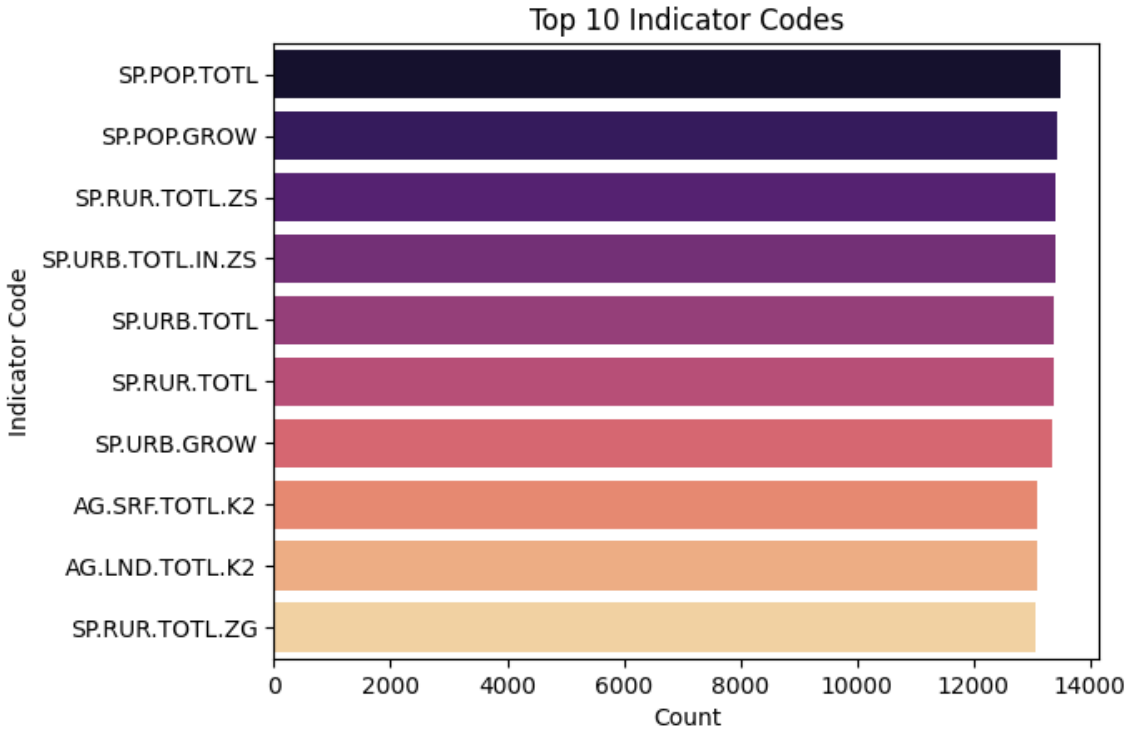
FOR CATEGORICAL DATA:

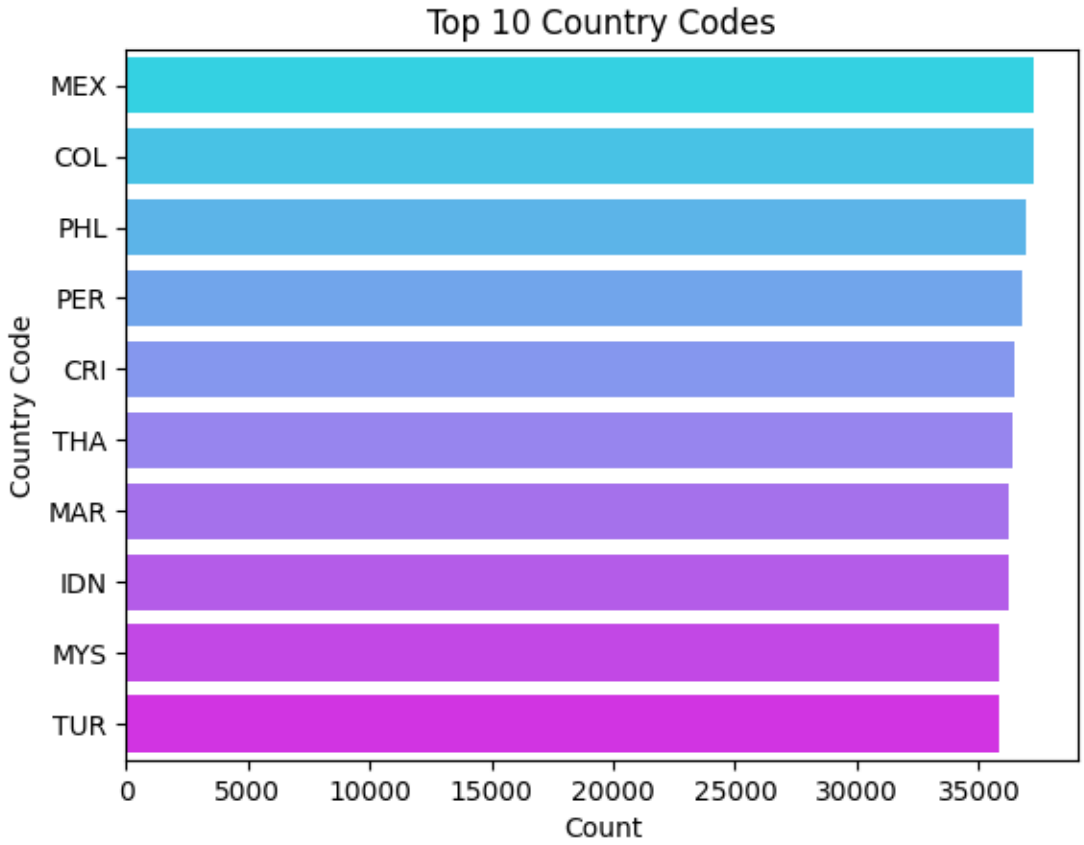



Country Distribution (Top 10)
Philippines
Colombia
Peru
Mexico
Costa Rica
Thailand
Turkey
Morocco
Malaysia
Indonesia
10.1%
10.2%
10.1%
10.2%
10.0%
10.0%
9.8%
9.9%
9.8%
9.9%

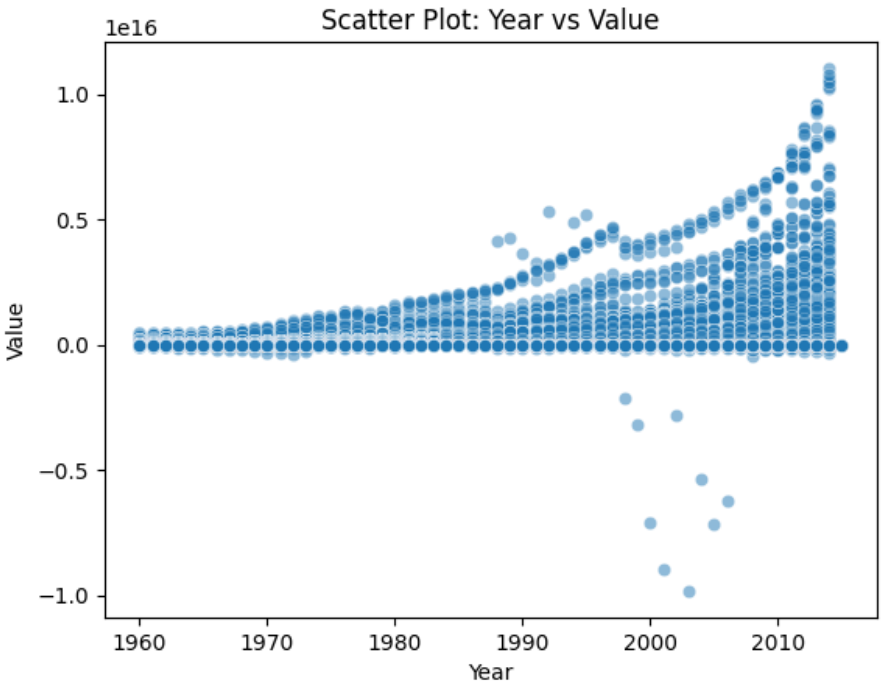

Top 10 Indicators
Indicator Name
Population, total
Population growth (annual %)
Rural population (% of total population)
Urban population (% of total)
Urban population
Rural population
Urban population growth (annual %)
Surface area (sq. km)
Land area (sq. km)
Rural population growth (annual %)
0
2000
4000
6000
8000
10000
12000
14000
Count

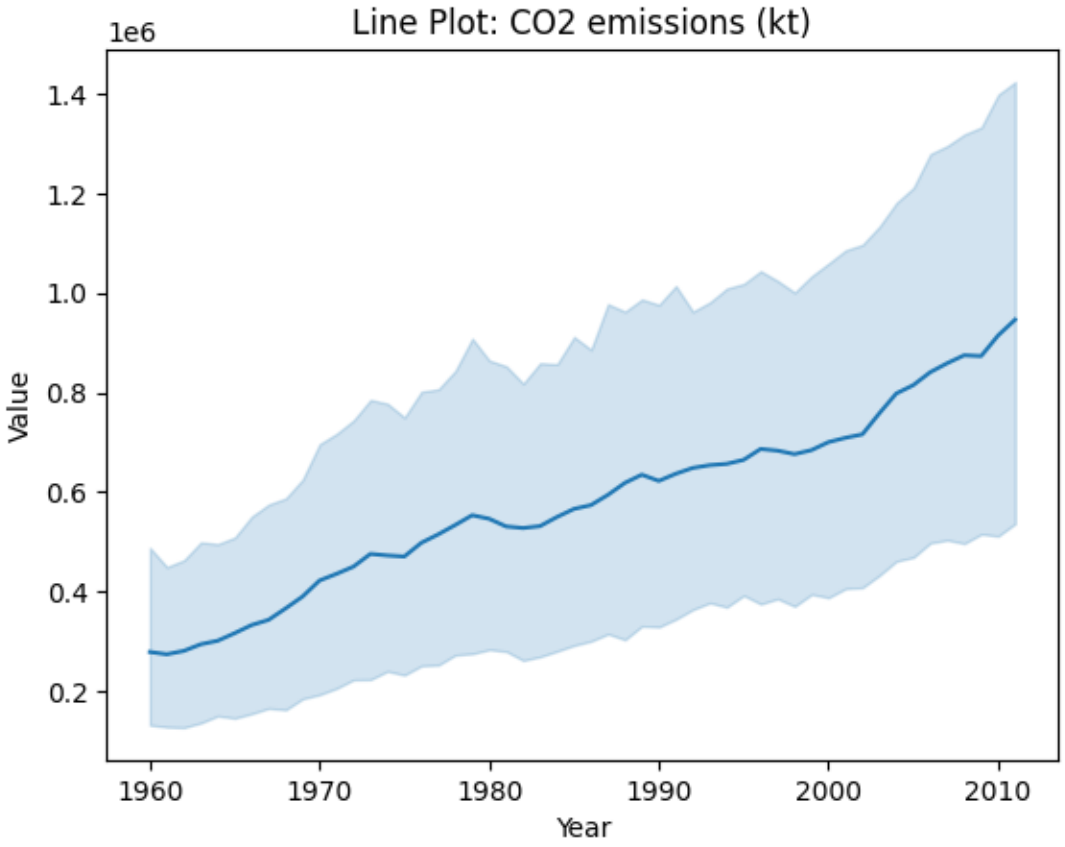

Indicator Distribution (Top 10)
Rural population (% of total population)
Urban population (% of total)
Population growth (annual %)
Urban population
Population, total
Rural population
Rural population growth (annual %)
Urban population growth (annual %)
Land area (sq. km)
Surface area (sq. km)
10.1%
10.1%
10.1%
10.1%
10.0%
10.0%
10.0%
9.8%
9.8%
9.8%

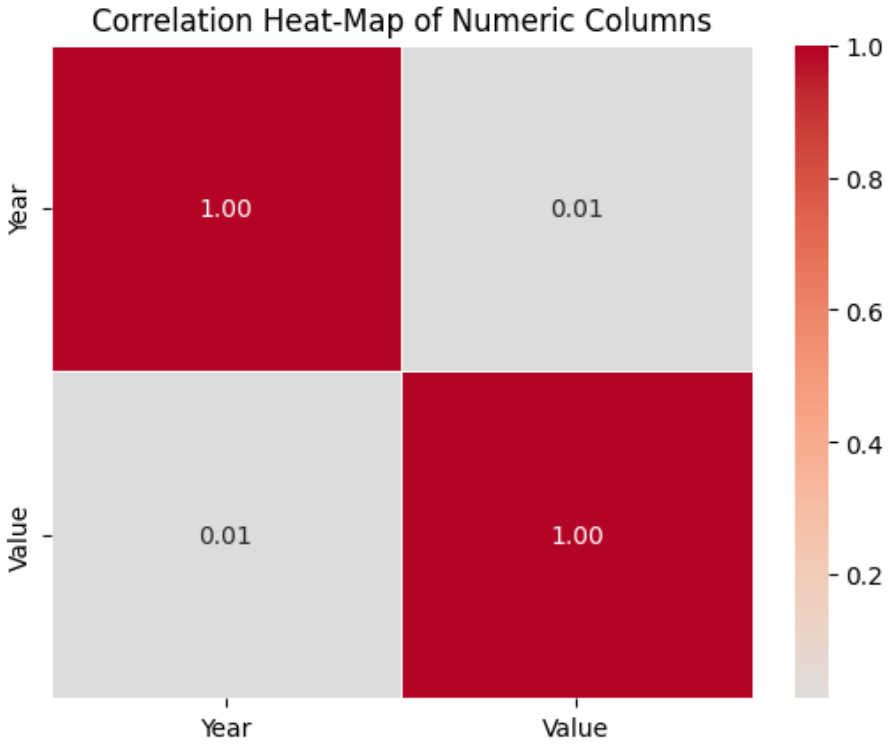

Top 10 Indicator Codes
Indicator Code
SP.POP.TOTL
SP.POP.GROW
SP.RUR.TOTL.ZS
SP.URB.TOTL.IN.ZS
SP.URB.TOTL
SP.RUR.TOTL
SP.URB.GROW
AG.SRF.TOTL.K2
AG.LND.TOTL.K2
SP.RUR.TOTL.ZG
0
2000
4000
6000
8000
10000
12000
14000
Count

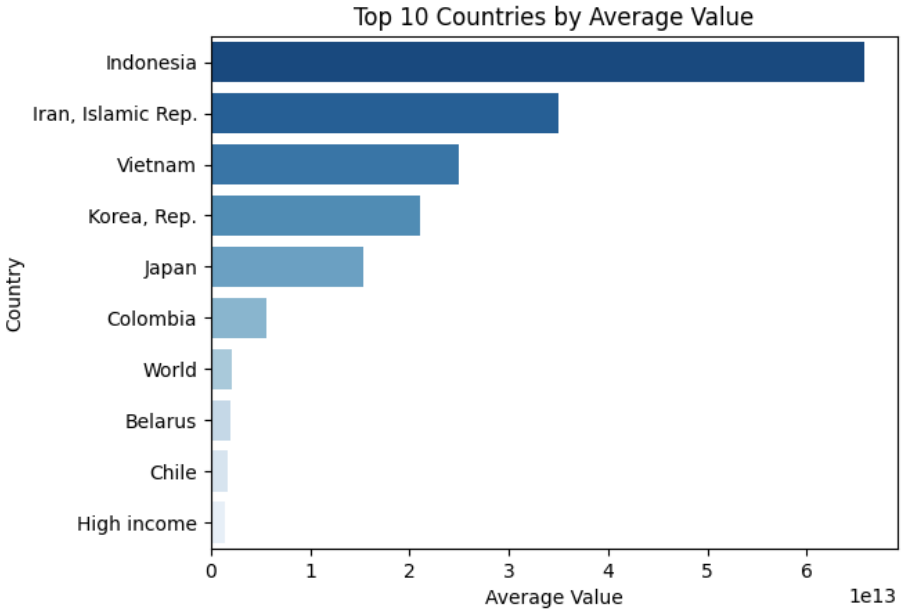


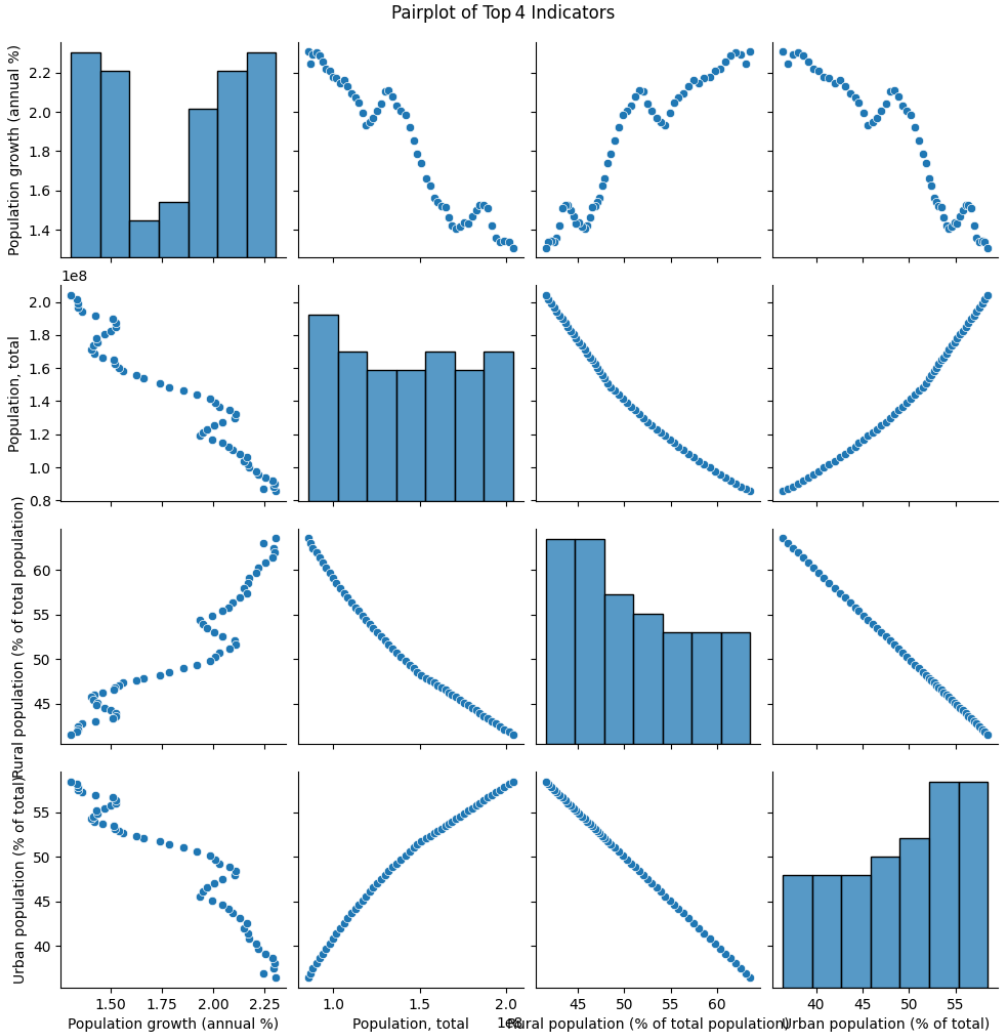
BIVARIATE ANALYSIS:




Line Plot: CO2 emissions (kt)
1e6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
Value
1960
1970
1980
1990
2000
2010
Year


Correlation Heat-Map of Numeric Columns
Year
Value
1.00
0.01
0.01
1.00
Year
Value
1.0
0.8
0.6
0.4
0.2


Top 10 Countries by Average Value
Indonesia
Iran, Islamic Rep.
Vietnam
Korea, Rep.
Japan
Colombia
World
Belarus
Chile
High income
Country
0
1
2
3
4
5
6
Average Value
1e13


Pairplot of Top 4 Indicators
Population growth (annual %)
Population, total
1e8
Rural population (% of total population)
Urban population (% of total)
Population growth (annual %)
Population, total
Rural population (% of total population)
Urban population (% of total)

LOADING DATA:

```
[23] import kagglehub
     kaggle_path = kagglehub.dataset_download('kaggle/world-development-indicators')
     csv_path = os.path.join(kaggle_path, 'Indicators.csv')
     df = pd.read_csv(csv_path)

[24] print(f"Shape: {df.shape}")
     df.head()
```

```
Shape: (5656458, 6)
```

	CountryName	CountryCode	IndicatorName	IndicatorCode	Year	Value
0	Arab World	ARB	Adolescent fertility rate (births per 1,000 wo...	SP.ADO.TFRT	1960	1.335609e+02
1	Arab World	ARB	Age dependency ratio (% of working-age populat...	SP.POP.DPND	1960	8.779760e+01
2	Arab World	ARB	Age dependency ratio, old (% of working-age po...	SP.POP.DPND.OL	1960	6.634579e+00
3	Arab World	ARB	Age dependency ratio, young (% of working-age ...	SP.POP.DPND.YG	1960	8.102333e+01
4	Arab World	ARB	Arms exports (SIPRI trend indicator values)	MS.MIL.XPRT.KD	1960	3.000000e+06

HANDLING MISSING DATA:

```
missing_counts = df.isnull().sum()
missing_percent = (missing_counts / len(df)) * 100
print("Missing percentage:\n",missing_percent)
```

```
Missing percentage:
 CountryName      0.0
CountryCode       0.0
IndicatorName     0.0
IndicatorCode     0.0
Year              0.0
Value             0.0
dtype: float64
```