**Project Title: Predicting CO₂ Emissions Using Machine Learning  
Team ID: SWTID1749709340  
Date: 4 July 2025**

**1. Project Overview**

The goal of this project was to develop a machine learning model capable of predicting CO₂ emissions (in kilotons) based on country and year inputs. The project involved data preprocessing, exploratory data analysis (EDA), model training and evaluation, deployment via a web interface, and visualization of key trends in global CO₂ emissions.

**2. Dataset Description**

The dataset was obtained from the World Bank Indicators dataset, containing over 5 million rows covering multiple countries, years, and development indicators. We filtered it to focus exclusively on the **CO₂ emissions (kt)** indicator.

**Key features used:**

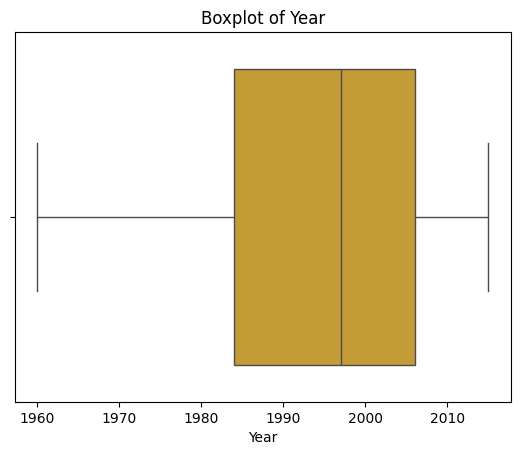
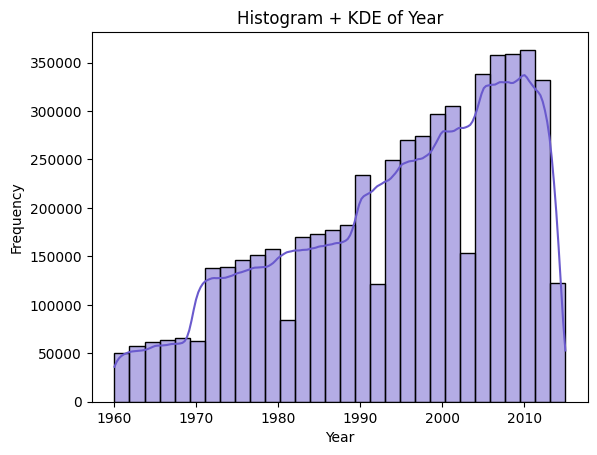
* CountryName
* CountryCode
* IndicatorName
* IndicatorCode
* Year
* Value (Target variable)

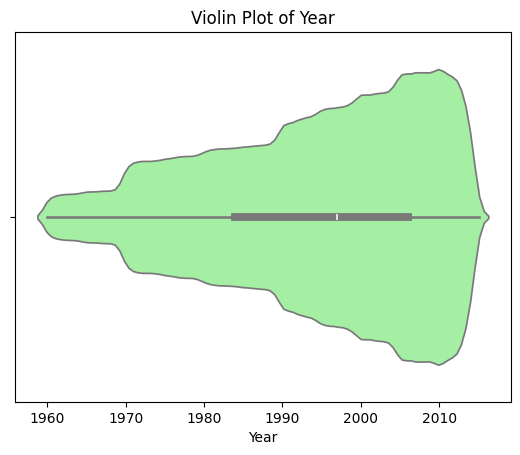
**3. Exploratory Data Analysis (EDA)**

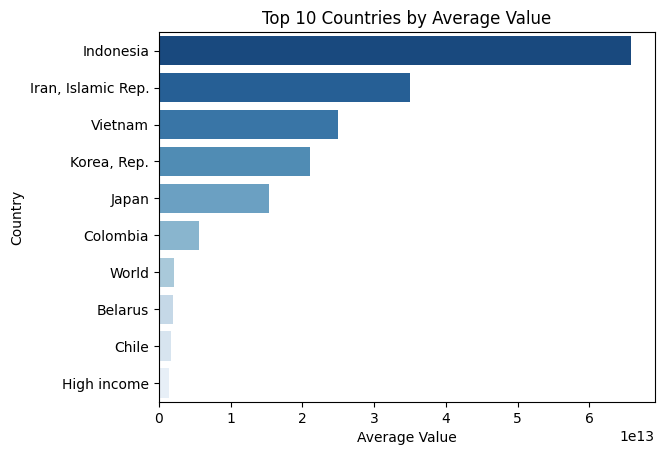
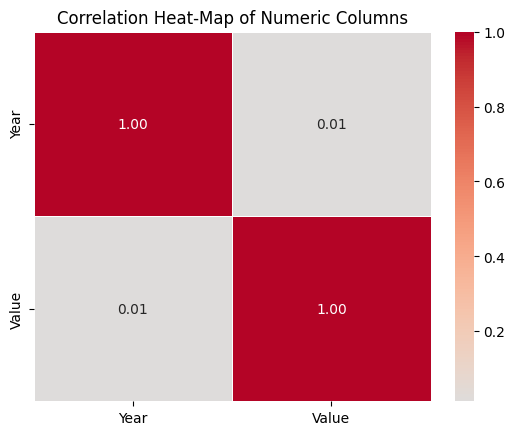
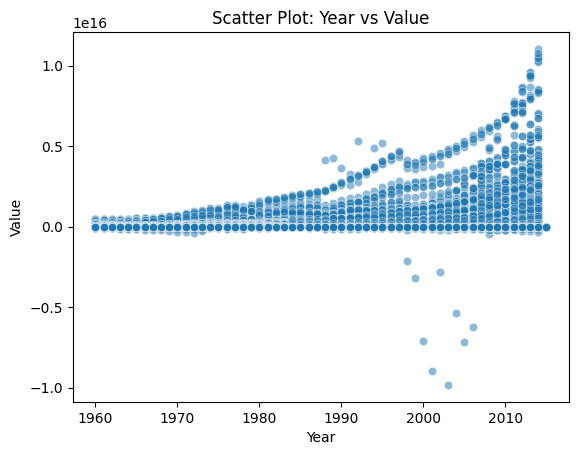
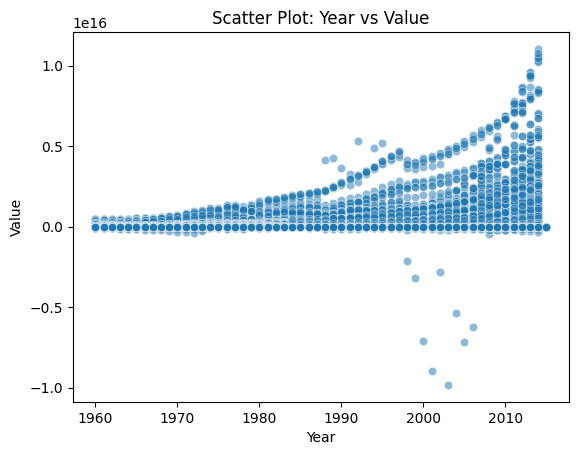
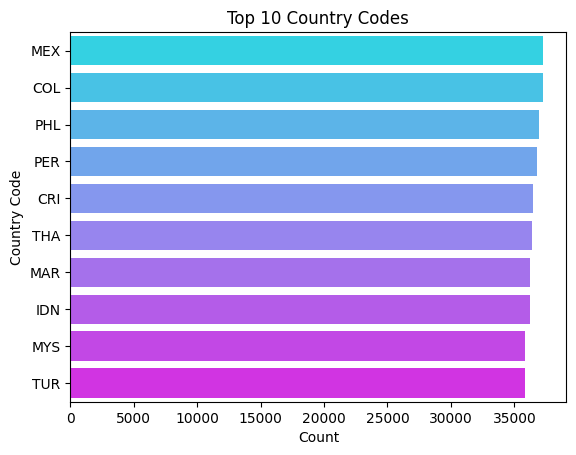
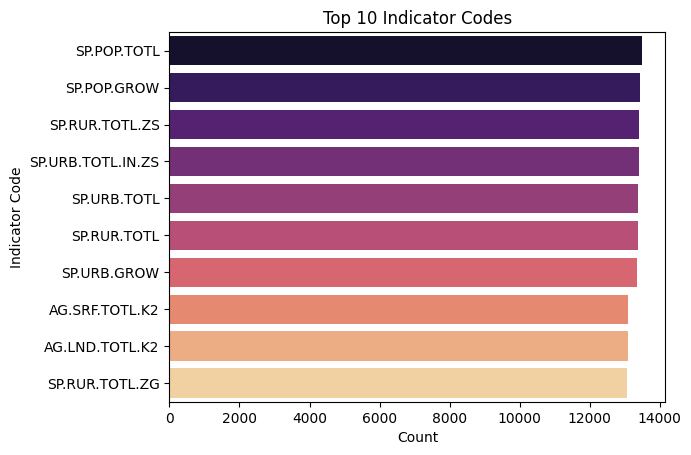
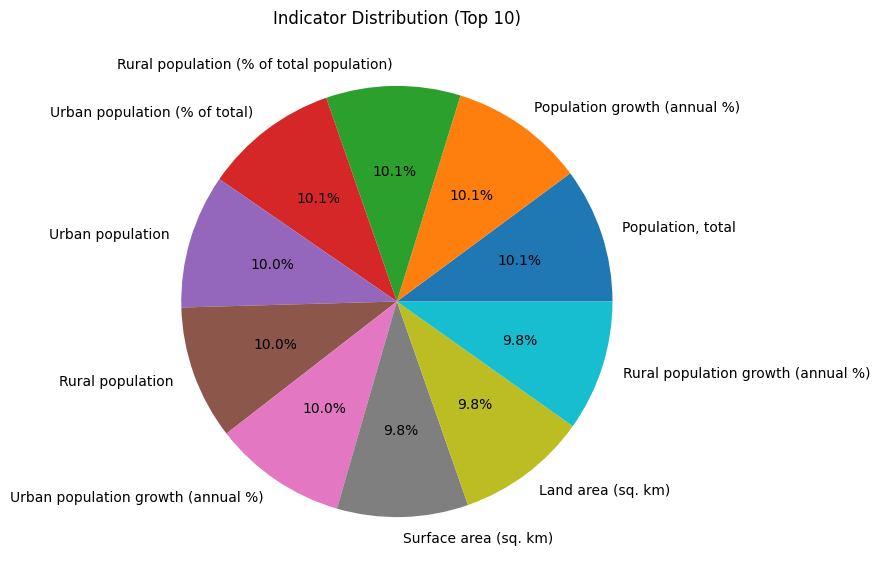
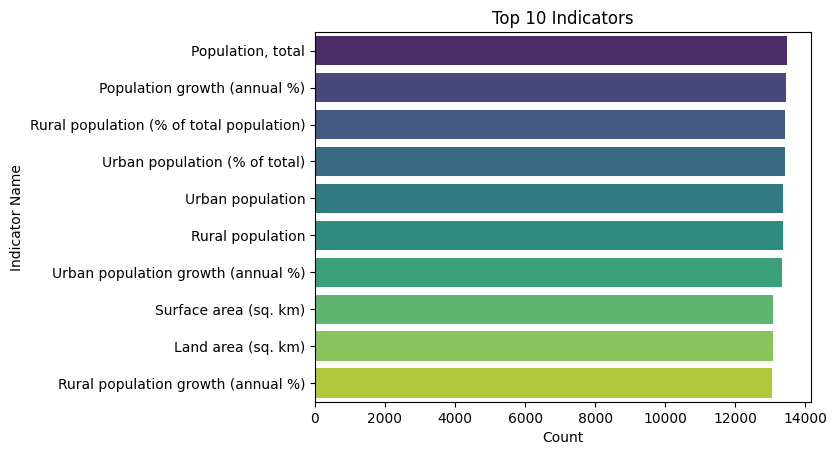
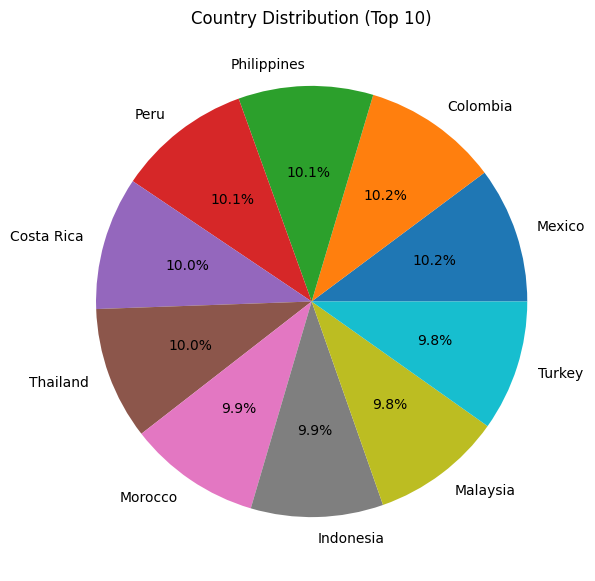
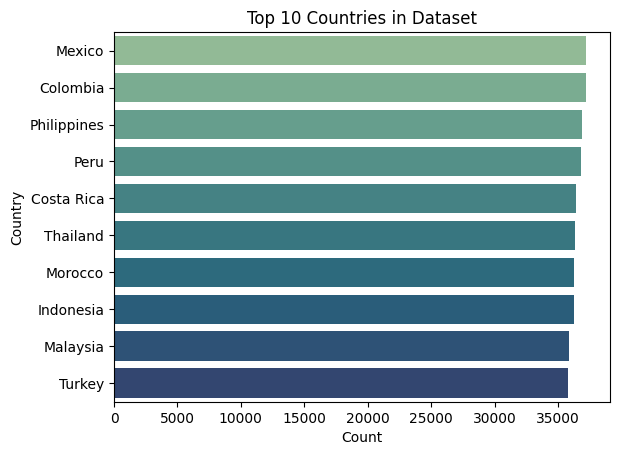
EDA was performed using **Matplotlib** and **Seaborn** to understand trends, patterns, and distributions in CO₂ emissions. The following visualizations were created:

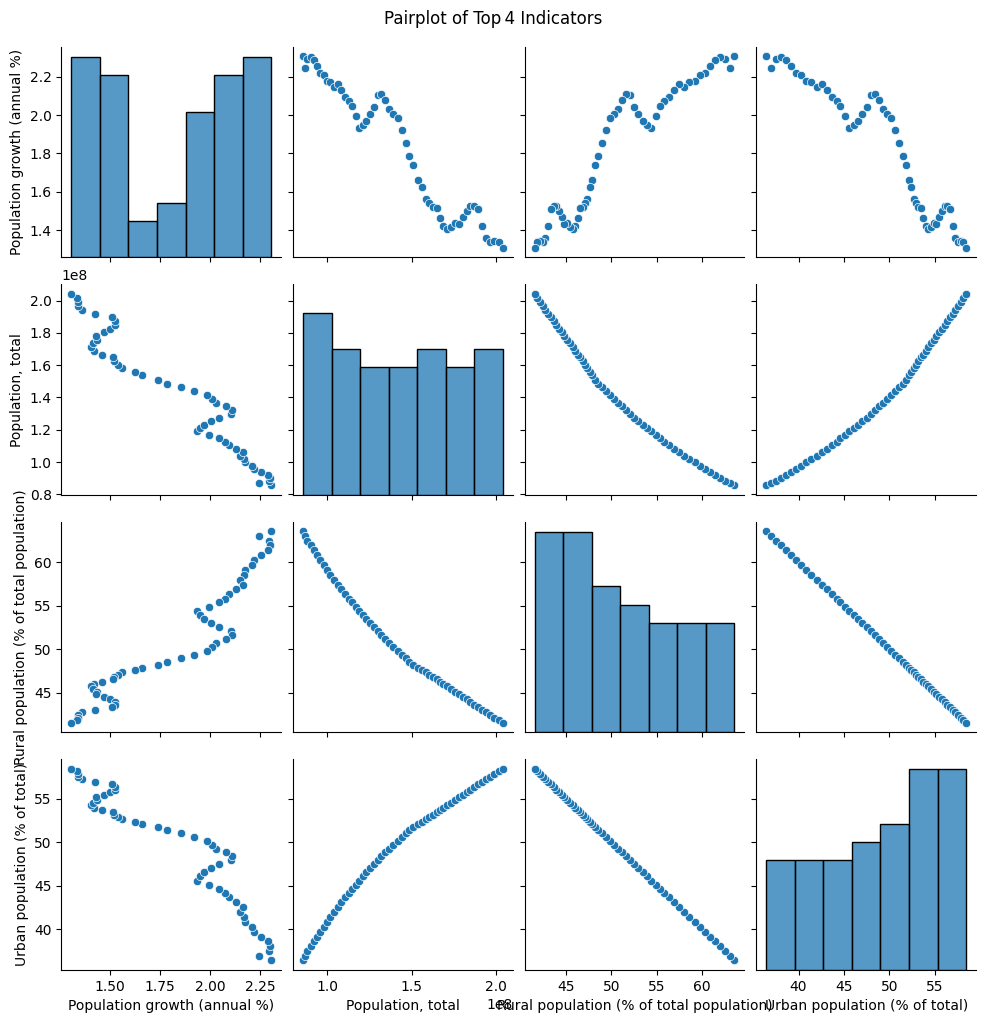
* Bar charts showing top countries by average CO₂ emissions
* Time-series plots for specific countries like India
* Histograms, KDE plots, and violin plots to show distribution
* Correlation heatmaps and pair plots for numeric analysis
* Pie charts to categorize countries based on emission levels

These visuals revealed clear upward trends in emissions for industrial nations and wide variability across countries.









**4. Data Preprocessing**

The preprocessing steps included:

* Dropping missing or irrelevant entries
* Encoding categorical variables (CountryName, IndicatorCode, etc.) using **One-Hot Encoding**
* Scaling numerical features like Year using **StandardScaler**
* Using **ColumnTransformer** to combine preprocessing steps in a clean pipeline

**5. Model Deployment**

We used **Pickle** to save the trained Random Forest model and the label encoder. The deployment was implemented as a **Flask web app** with:

* HTML frontend (user inputs country and year)
* Prediction endpoint
* **Ngrok** tunneling to serve the site publicly via temporary HTTPS URL (for demonstration)

This app allowed real-time prediction of CO₂ emissions for any country-year pair.

**6. Tools & Technologies Used**

* **Python (Pandas, scikit-learn, XGBoost)**
* **Matplotlib, Seaborn** for visualization
* **Flask + HTML** for deployment
* **Ngrok** for hosting the web app
* **Jupyter Notebook / Google Colab** for development
* **Pickle** for model persistence

**7. Conclusion & Learnings**

This project demonstrated how data science and machine learning can be used to model and visualize environmental trends like CO₂ emissions. By building a deployable prediction tool, we showcased the end-to-end ML pipeline from data ingestion to real-world application. Key takeaways included:

* Importance of EDA and preprocessing
* Trade-offs between model performance and interpretability
* Real-world deployment using minimal tools (Flask + Ngrok)