##**Final Project Plan – Winter 2025/2026**

Objective

Analyze and visualize global development trends using the World Bank World Development Indicators (WDI) dataset. Focus on socio-economic, governance, environmental, and population metrics to uncover insights about global development patterns and country-level performance.

Key idea: Tell a story with the data – e.g., how governance, economic, and environmental factors correlate with human development.

Setup & Libraries

```
# Basic data manipulation
import pandas as pd
import numpy as np

# Visualization
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px

# For maps
import geopandas as gpd
import folium
```

Load Dataset

```
import pandas as pd
from google.colab import files

# Load CSV into DataFrame
df = pd.read_csv("/content/world_bank_development_indicators.csv")

# Quick look
df.head()
df.info()
df.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17272 entries, 0 to 17271
Data columns (total 50 columns):
 #   Column                                               Non-Null Count  Dtype
---  ------                                               --------------  -----
 0   country                                              17272 non-null  object
 1   date                                                 17272 non-null  object
```

```
 2   agricultural_land%                               14714 non-null
float64
 3   forest_land%                                     8176 non-null
float64
 4   land_area                                        14930 non-null
float64
 5   avg_precipitation                                10086 non-null
float64
 6   trade_in_services%                               9195 non-null
float64
 7   control_of_corruption_estimate                   4783 non-null
float64
 8   control_of_corruption_std                        4783 non-null
float64
 9   access_to_electricity%                           7348 non-null
float64
 10  renewvable_energy_consumption%                   8076 non-null
float64
 11  electric_power_consumption                       7790 non-null
float64
 12  CO2_emisions                                     7408 non-null
float64
 13  other_greenhouse_emisions                        7408 non-null
float64
 14  population_density                               14901 non-null
float64
 15  inflation_annual%                                10788 non-null
float64
 16  real_interest_rate                               4416 non-null
float64
 17  risk_premium_on_lending                          2370 non-null
float64
 18  research_and_development_expenditure%            2889 non-null
float64
 19  central_goverment_debt%                          2080 non-null
float64
 20  tax_revenue%                                     5125 non-null
float64
 21  expense%                                         4769 non-null
float64
 22  goverment_effectiveness_estimate                 4759 non-null
float64
 23  goverment_effectiveness_std                      4759 non-null
float64
 24  human_capital_index                              601 non-null
float64
 25  doing_business                                   189 non-null
float64
 26  time_to_get_operation_license                    371 non-null
```

```
float64
 27  statistical_performance_indicators        1237 non-null
float64
 28  individuals_using_internet%               8044 non-null
float64
 29  logistic_performance_index                1407 non-null
float64
 30  military_expenditure%                     10122 non-null
float64
 31  GDP_current_US                            13198 non-null
float64
 32  political_stability_estimate              4820 non-null
float64
 33  political_stability_std                   4820 non-null
float64
 34  rule_of_law_estimate                      4873 non-null
float64
 35  rule_of_law_std                           4873 non-null
float64
 36  regulatory_quality_estimate               4761 non-null
float64
 37  regulatory_quality_std                    4761 non-null
float64
 38  government_expenditure_on_education%      6107 non-null
float64
 39  government_health_expenditure%            4938 non-null
float64
 40  multidimensional_poverty_headcount_ratio%  455 non-null
float64
 41  gini_index                                2108 non-null
float64
 42  birth_rate                                16037 non-null
float64
 43  death_rate                                16019 non-null
float64
 44  life_expectancy_at_birth                  15866 non-null
float64
 45  population                                16665 non-null
float64
 46  rural_population                          16539 non-null
float64
 47  voice_and_accountability_estimate         4850 non-null
float64
 48  voice_and_accountability_std              4850 non-null
float64
 49  intentional_homicides                     4209 non-null
float64
dtypes: float64(48), object(2)
memory usage: 6.6+ MB
```

{"type":"dataframe"}

Data Cleaning & Preparation

```
# Combine country + year as index
df['country_year'] = df['country'] + '_' + df['date'].astype(str)
df.set_index('country_year', inplace=True)

# Convert percentage columns to numeric (if needed)
percent_cols = [col for col in df.columns if '%' in col]
for col in percent_cols:
    df[col] = pd.to_numeric(df[col], errors='coerce')

# Fill missing numeric values (example: forward fill by country)
df = df.groupby('country').apply(lambda x: x.fillna(method='ffill'))
```

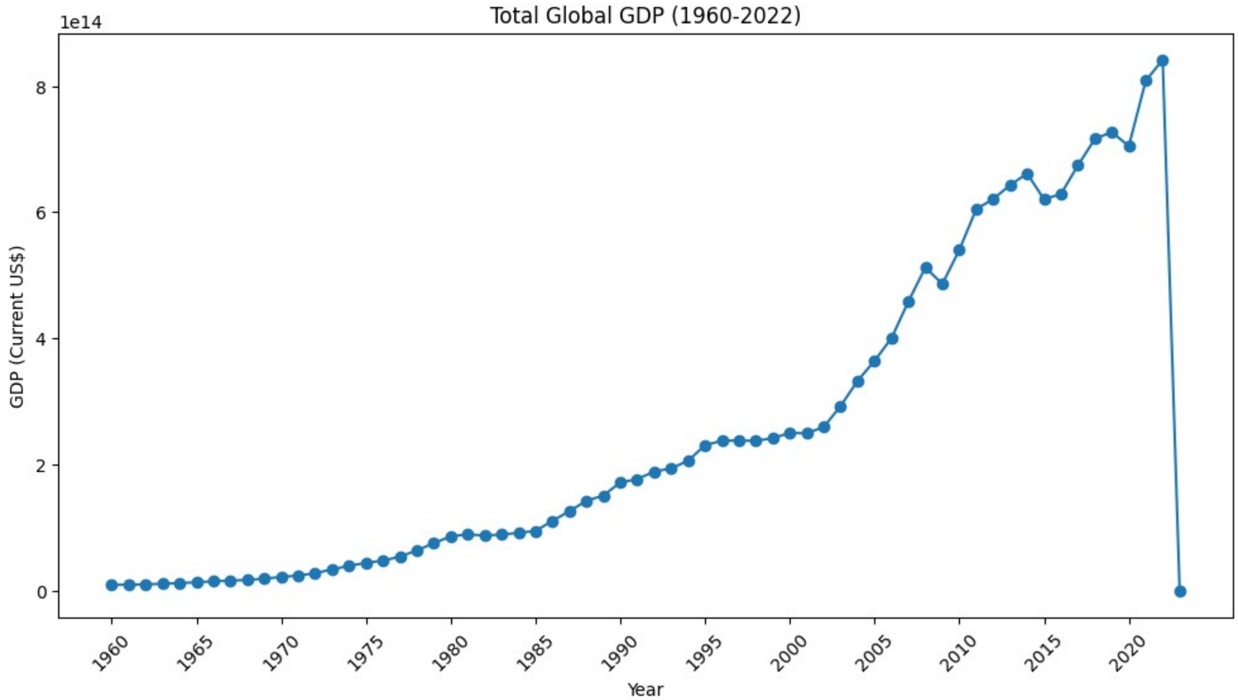
```
/tmp/ipython-input-3026376907.py:11: FutureWarning: DataFrame.fillna
with 'method' is deprecated and will raise in a future version. Use
obj.ffill() or obj.bfill() instead.
  df = df.groupby('country').apply(lambda x: x.fillna(method='ffill'))
/tmp/ipython-input-3026376907.py:11: DeprecationWarning:
DataFrameGroupBy.apply operated on the grouping columns. This behavior
is deprecated, and in a future version of pandas the grouping columns
will be excluded from the operation. Either pass
`include_groups=False` to exclude the groupings or explicitly select
the grouping columns after groupby to silence this warning.
  df = df.groupby('country').apply(lambda x: x.fillna(method='ffill'))
```

Exploratory Analysis & Visualizations (15 Questions)

1) **Global Economic Growth:** How has the total global GDP evolved from 1960 to 2022?

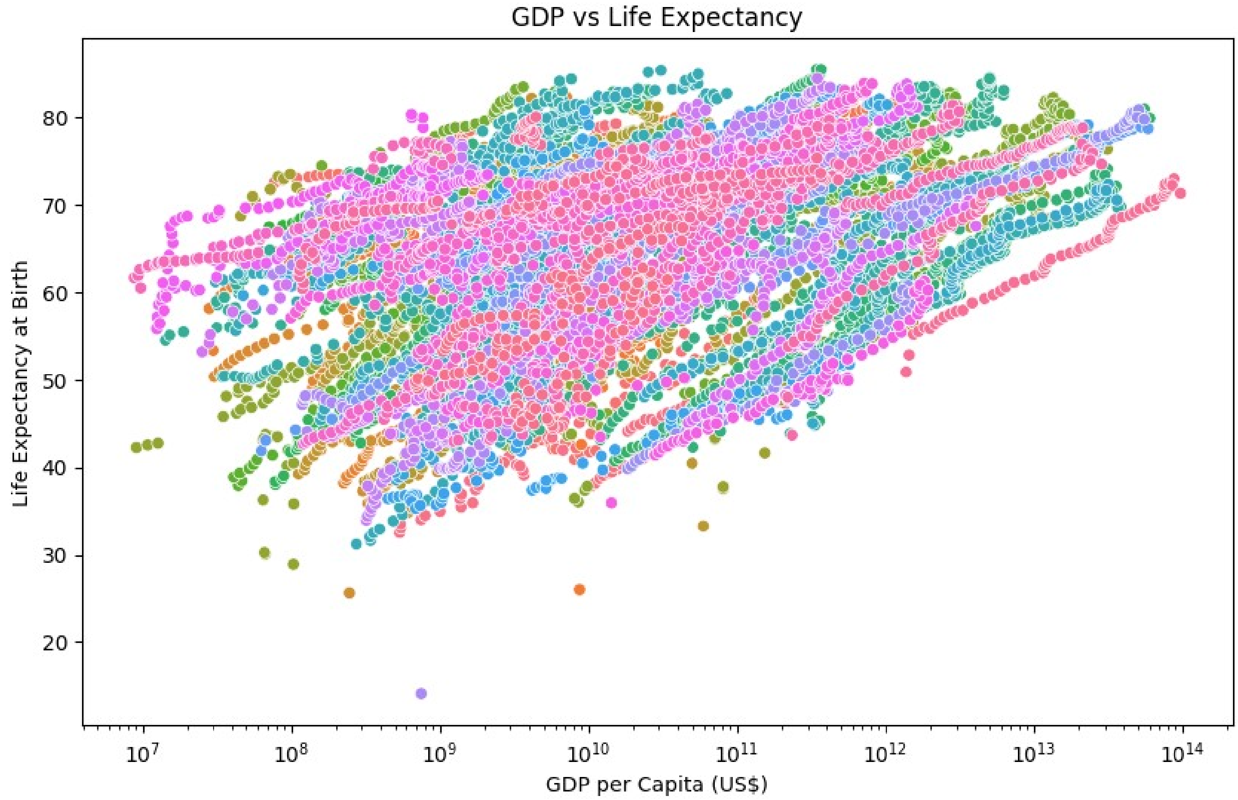
```
import matplotlib.pyplot as plt
import pandas as pd

# Ensure 'date' is numeric (extract year from date strings)
df['date'] = pd.to_datetime(df['date']).dt.year

# Aggregate GDP
gdp_over_time = df.groupby('date')
['GDP_current_US'].sum().reset_index()

plt.figure(figsize=(12,6))
plt.plot(gdp_over_time['date'], gdp_over_time['GDP_current_US'],
marker='o')
plt.title('Total Global GDP (1960-2022)')
plt.xlabel('Year')
plt.ylabel('GDP (Current US$)')

# Show only every 5th year
```

```
plt.xticks(gdp_over_time['date'][::5], rotation=45)

plt.show()
```



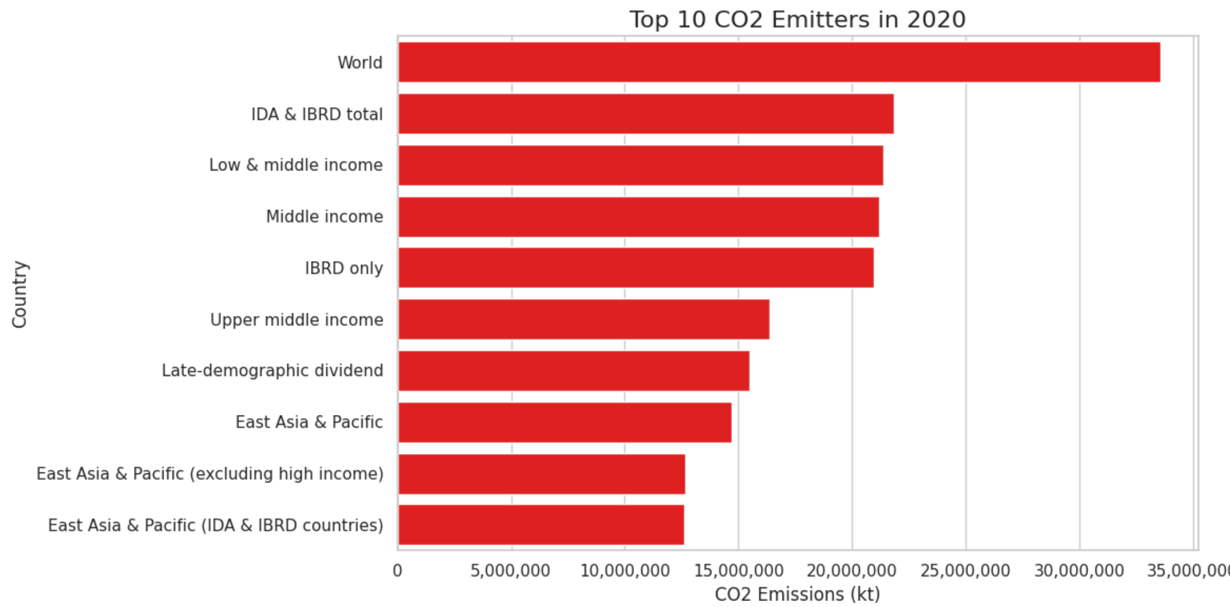
2) **Wealth vs. Health:** What is the relationship between GDP per capita and Life Expectancy?

```
plt.figure(figsize=(10,6))
sns.scatterplot(data=df, x='GDP_current_US',
y='life_expectancy_at_birth', hue='country', legend=False)
plt.xscale('log')
plt.title('GDP vs Life Expectancy')
plt.xlabel('GDP per Capita (US$)')
plt.ylabel('Life Expectancy at Birth')
plt.show()
```



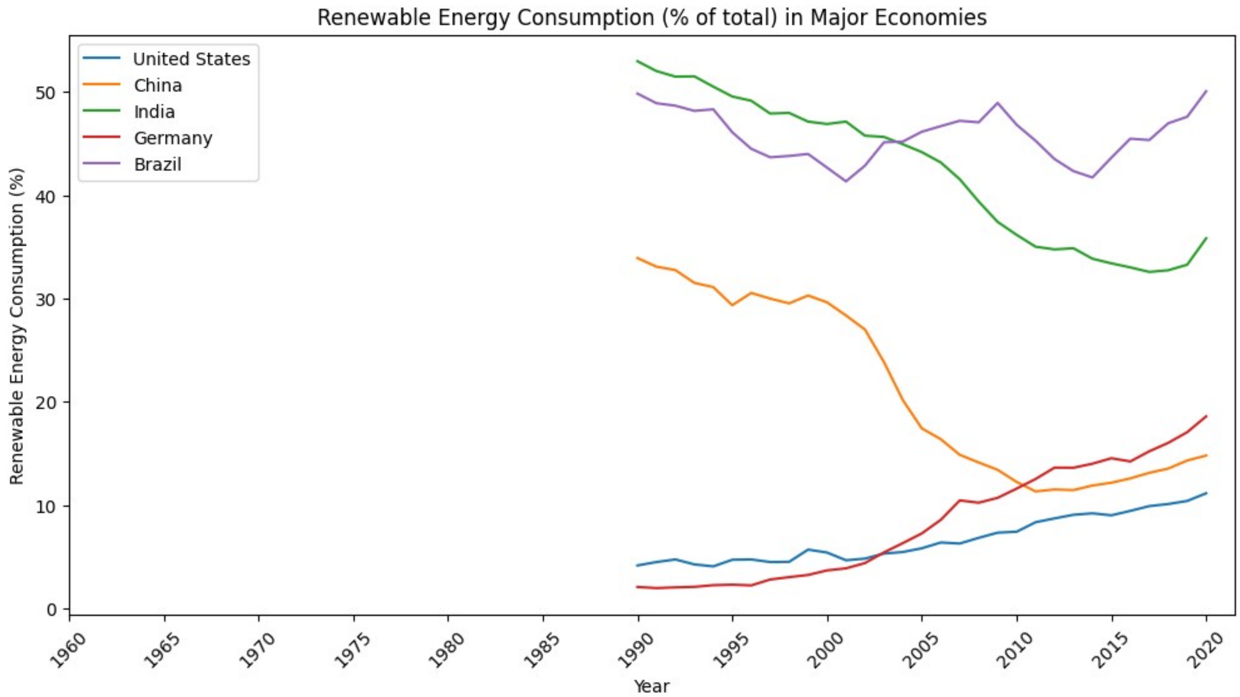
3) **Climate Impact:** Who are the top 10 CO2 emitters in the most recent year?

```
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib.ticker import StrMethodFormatter

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

# Corrected: Use 'date' column instead of 'year'
latest_co2_year = df[df['CO2_emisions'].notna()]['date'].max()
top_emitters = df[df['date'] == latest_co2_year].nlargest(10,
'CO2_emisions')

ax = sns.barplot(data=top_emitters, y='country', x='CO2_emisions',
color='red')

plt.title(f'Top 10 CO2 Emitters in {latest_co2_year}', fontsize=16)
plt.xlabel('CO2 Emissions (kt)', fontsize=12)
plt.ylabel('Country', fontsize=12)
ax.xaxis.set_major_formatter(StrMethodFormatter('{x:,.0f}'))

plt.tight_layout()
plt.show()
```



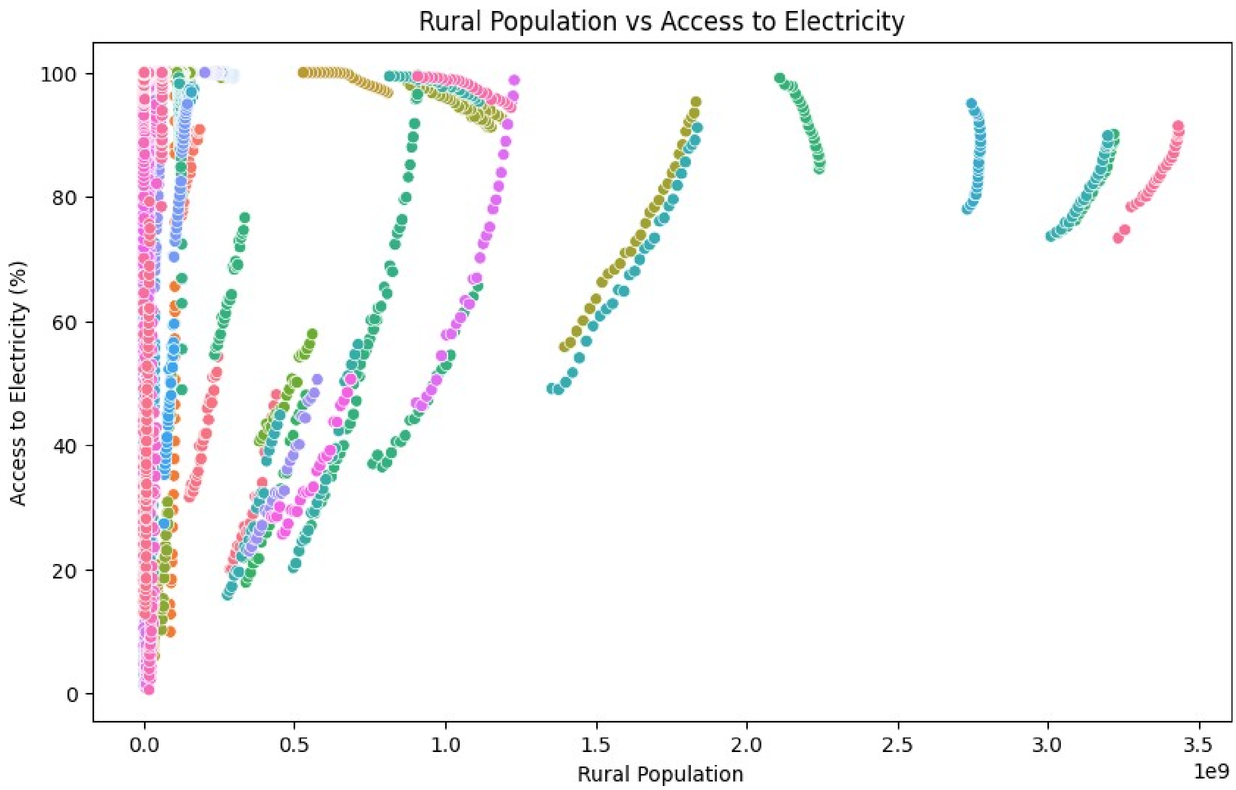
4) **Energy Transition:** How has renewable energy consumption evolved for major economies?

```
major_economies = ['United States', 'China', 'India', 'Germany',
'Brazil']

plt.figure(figsize=(12,6))
for country in major_economies:
    subset = df[df['country']==country]
    plt.plot(subset['date'], subset['renewvable_energy_consumption%'],
label=country)

plt.title('Renewable Energy Consumption (% of total) in Major
Economies')
plt.xlabel('Year')
plt.ylabel('Renewable Energy Consumption (%)')
plt.xticks(subset['date'][::5], rotation=45)  # Every 5 years
plt.legend()
plt.show()
```



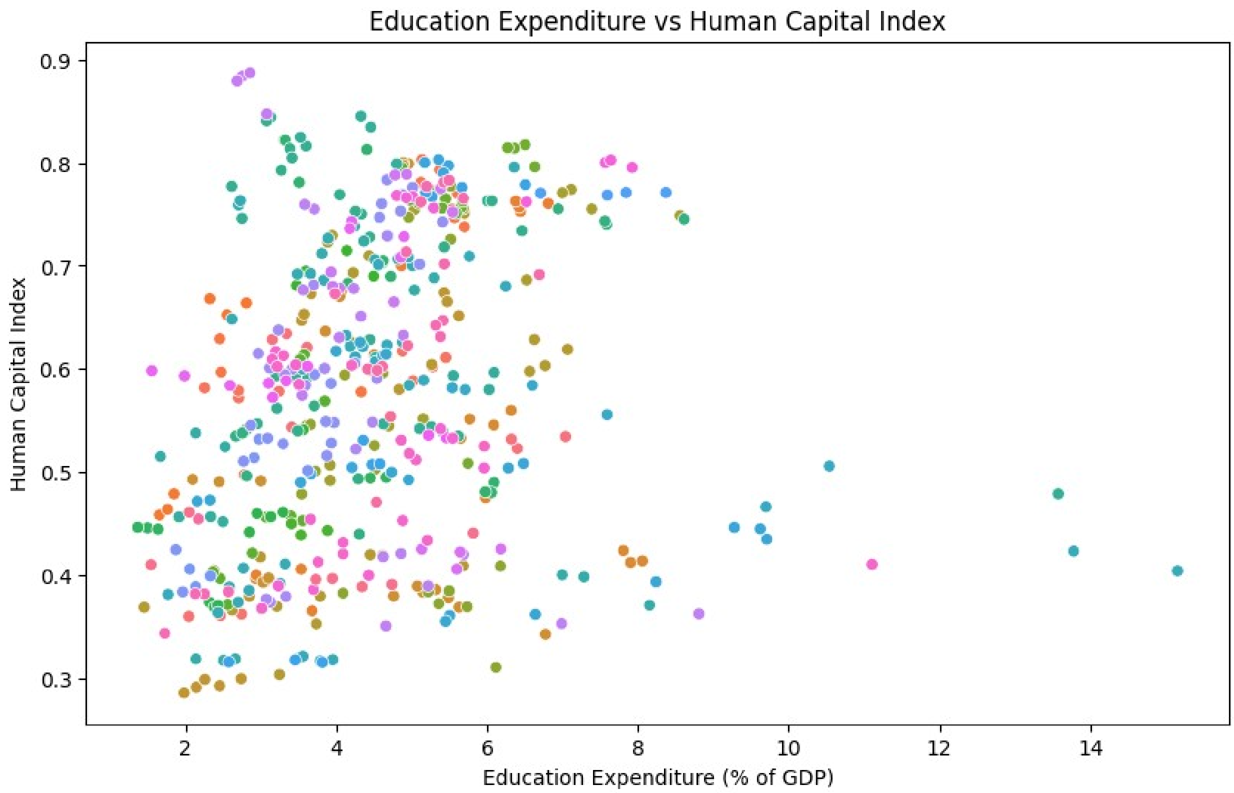
5) **Infrastructure Gap:** How does rurality correlate with access to electricity?

```
plt.figure(figsize=(10,6))
sns.scatterplot(data=df, x='rural_population',
y='access_to_electricity%', hue='country', legend=False)
plt.title('Rural Population vs Access to Electricity')
plt.xlabel('Rural Population')
plt.ylabel('Access to Electricity (%)')
plt.show()
```



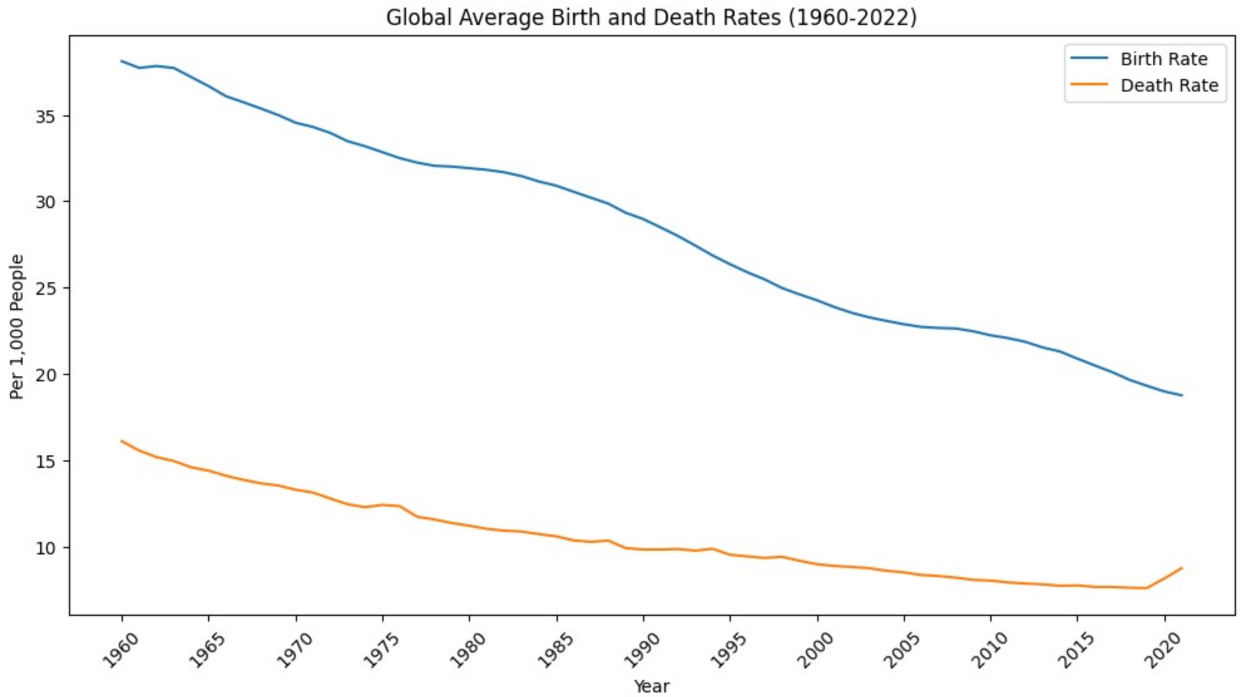
6) **Education Investment:** Does higher education spending correlate with the Human Capital Index?

```
plt.figure(figsize=(10,6))
sns.scatterplot(data=df, x='government_expenditure_on_education%',
y='human_capital_index', hue='country', legend=False)
plt.title('Education Expenditure vs Human Capital Index')
plt.xlabel('Education Expenditure (% of GDP)')
plt.ylabel('Human Capital Index')
plt.show()
```



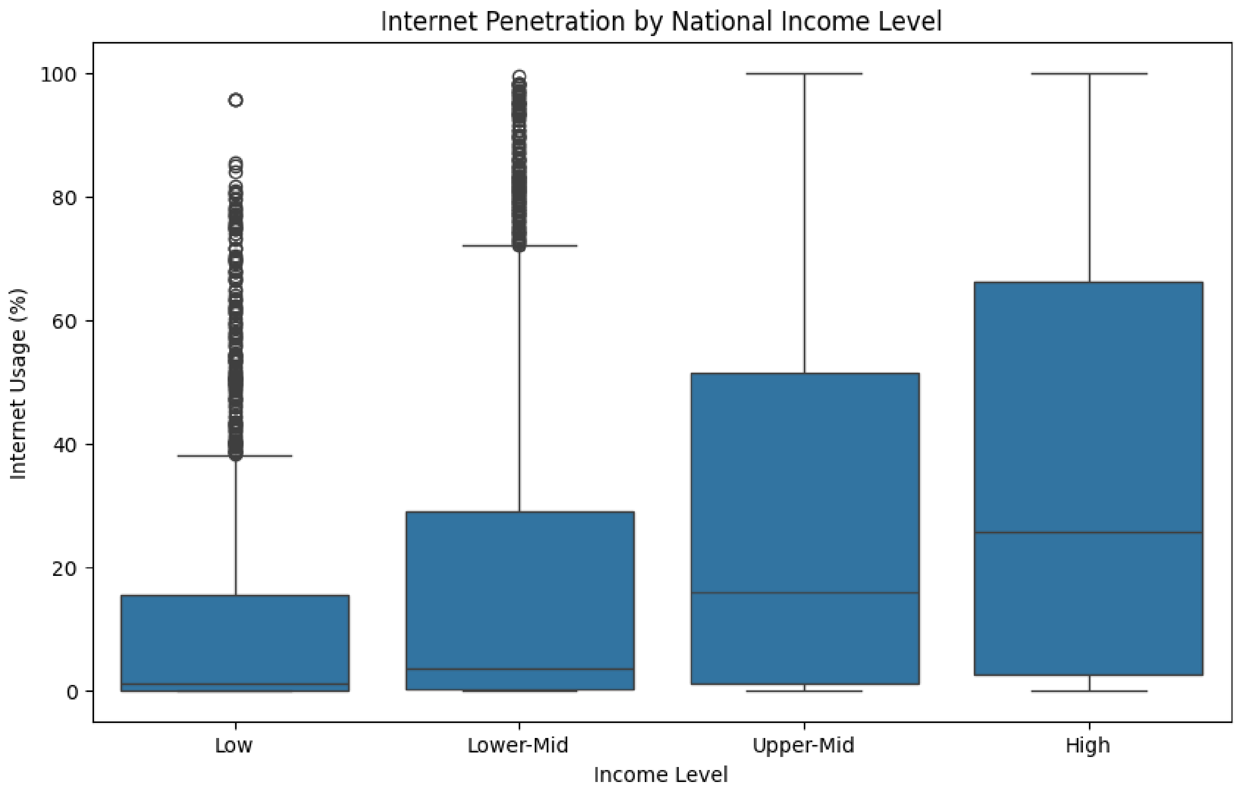
7) **Demographic Transition:** How have global average birth and death rates changed since 1960?

```
birth_death = df.groupby('date')[['birth_rate',
'death_rate']].mean().reset_index()

plt.figure(figsize=(12,6))
plt.plot(birth_death['date'], birth_death['birth_rate'], label='Birth
Rate')
plt.plot(birth_death['date'], birth_death['death_rate'], label='Death
Rate')
plt.title('Global Average Birth and Death Rates (1960-2022)')
plt.xlabel('Year')
plt.ylabel('Per 1,000 People')
plt.xticks(birth_death['date'][::5], rotation=45)  # Every 5 years
plt.legend()
plt.show()
```



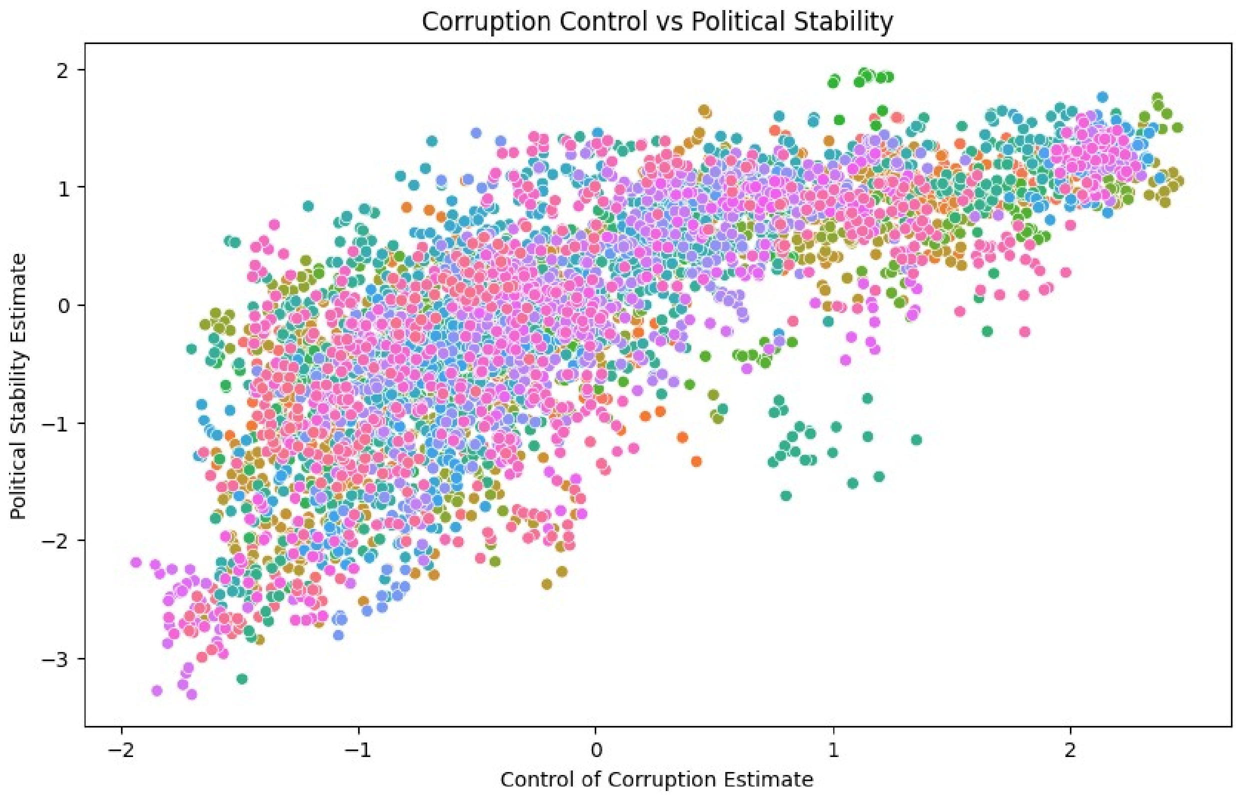
8) **The Digital Divide:** How does internet penetration differ by national income levels?

```
df['income_quartile'] = pd.qcut(df['GDP_current_US'], 4,
labels=['Low', 'Lower-Mid', 'Upper-Mid', 'High'])

plt.figure(figsize=(10,6))
sns.boxplot(data=df, x='income_quartile',
y='individuals_using_internet%')
plt.title('Internet Penetration by National Income Level')
plt.xlabel('Income Level')
plt.ylabel('Internet Usage (%)')
plt.show()
```



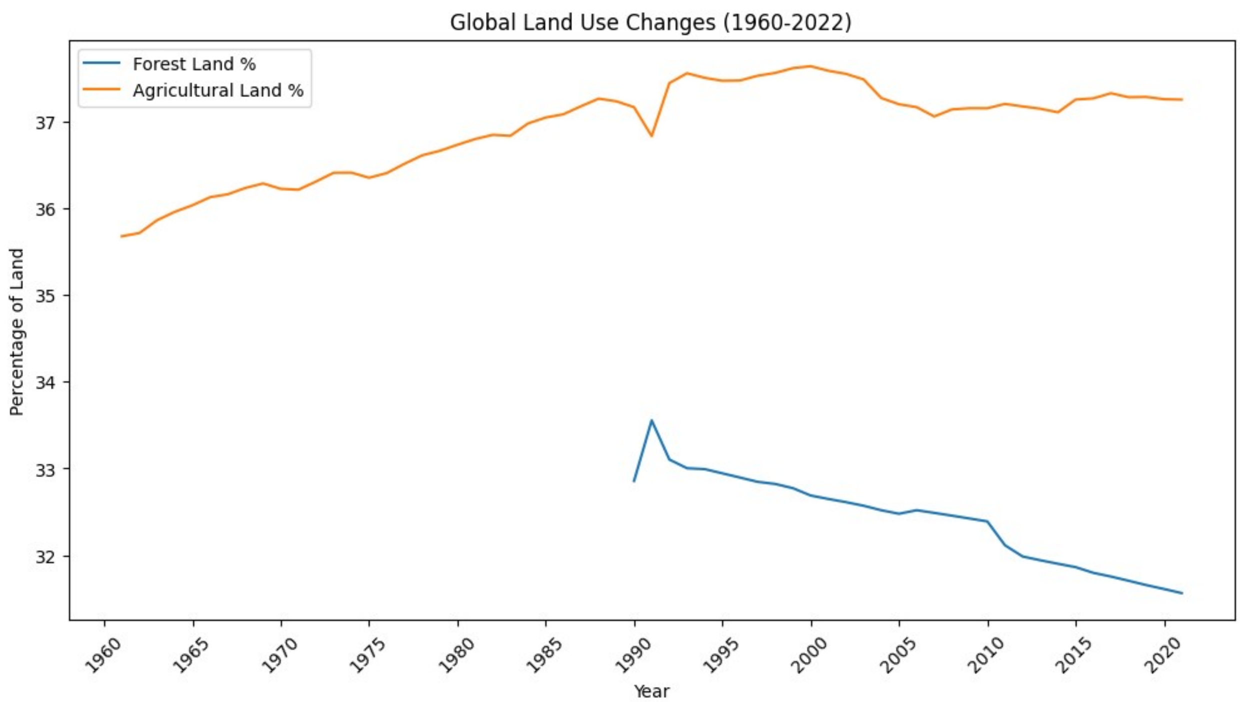
9) **Governance and Stability:** Is there a link between corruption control and political stability?

```
plt.figure(figsize=(10,6))
sns.scatterplot(data=df, x='control_of_corruption_estimate',
y='political_stability_estimate', hue='country', legend=False)
plt.title('Corruption Control vs Political Stability')
plt.xlabel('Control of Corruption Estimate')
plt.ylabel('Political Stability Estimate')
plt.show()
```



10) **Land Use Change:** How have global forest and agricultural land shares shifted over time?

```
land_use = df.groupby('date')[['forest_land%', 'agricultural_land
%']].mean().reset_index()

plt.figure(figsize=(12,6))
plt.plot(land_use['date'], land_use['forest_land%'], label='Forest
Land %')
plt.plot(land_use['date'], land_use['agricultural_land%'],
label='Agricultural Land %')
plt.title('Global Land Use Changes (1960-2022)')
plt.xlabel('Year')
plt.ylabel('Percentage of Land')
plt.xticks(land_use['date'][::5], rotation=45)
plt.legend()
plt.show()
```

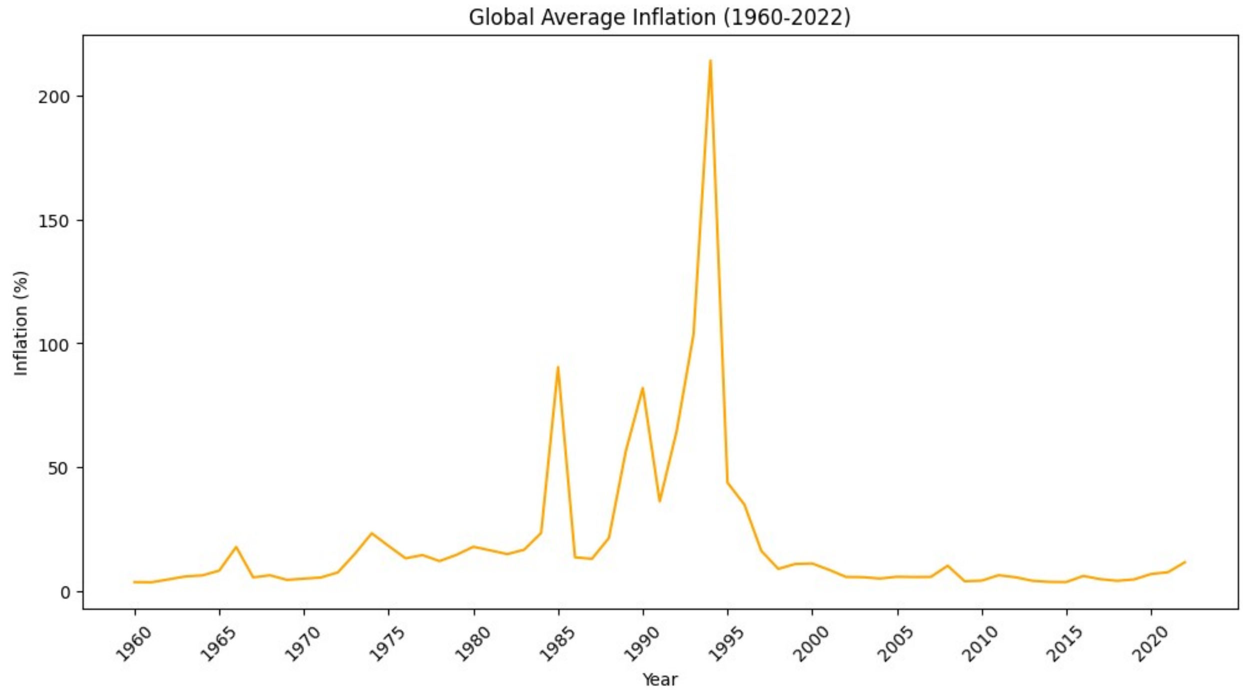


11) **Military Spending:** What is the long-term trend of global military expenditure as a % of GDP?

```
military = df.groupby('date')['military_expenditure
%'].mean().reset_index()

plt.figure(figsize=(12,6))
plt.plot(military['date'], military['military_expenditure%'],
marker='o')
plt.title('Global Military Expenditure (% of GDP) Over Time')
plt.xlabel('Year')
plt.ylabel('Military Expenditure (% of GDP)')
plt.xticks(military['date'][::5], rotation=45)
plt.show()
```

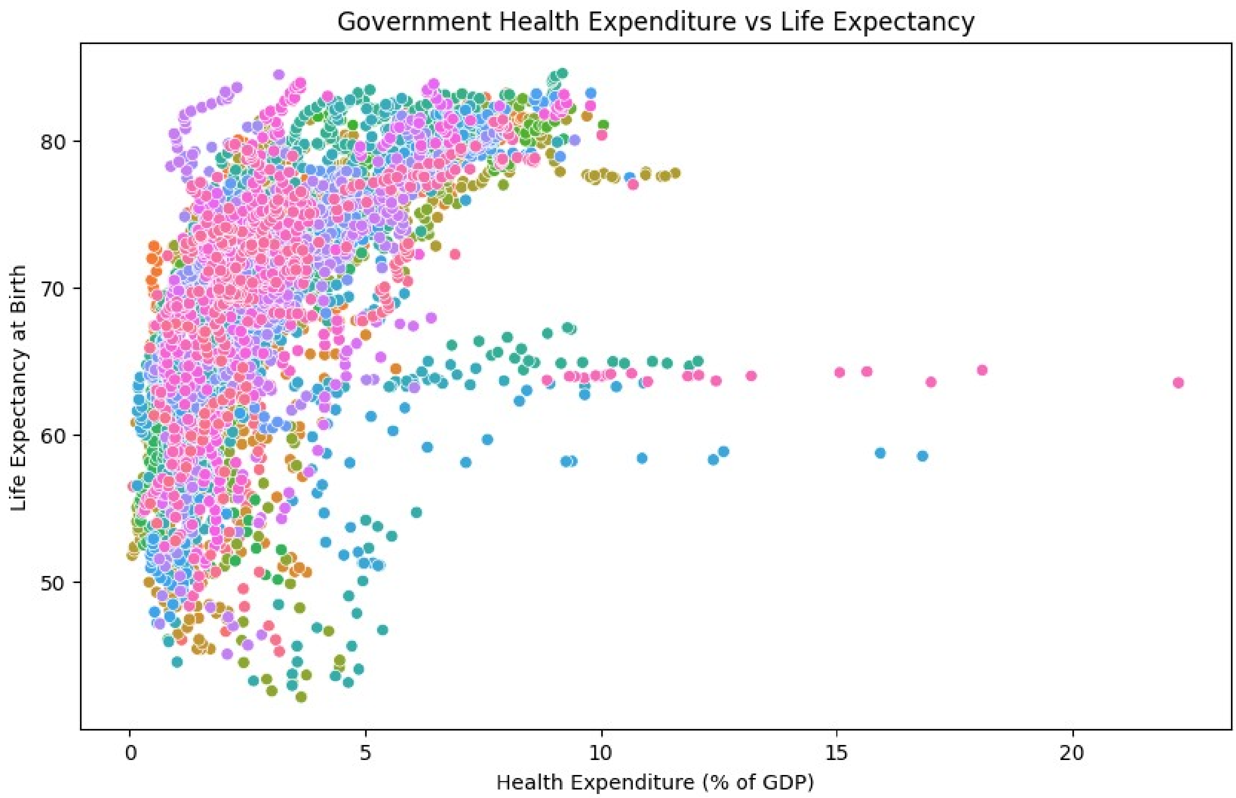
12) **Economic Volatility:** How has global average inflation fluctuated over the decades?

```
inflation = df.groupby('date')['inflation_annual
%'].mean().reset_index()

plt.figure(figsize=(12,6))
plt.plot(inflation['date'], inflation['inflation_annual%'],
color='orange')
plt.title('Global Average Inflation (1960-2022)')
plt.xlabel('Year')
plt.ylabel('Inflation (%)')
plt.xticks(inflation['date'][::5], rotation=45)
plt.show()
```



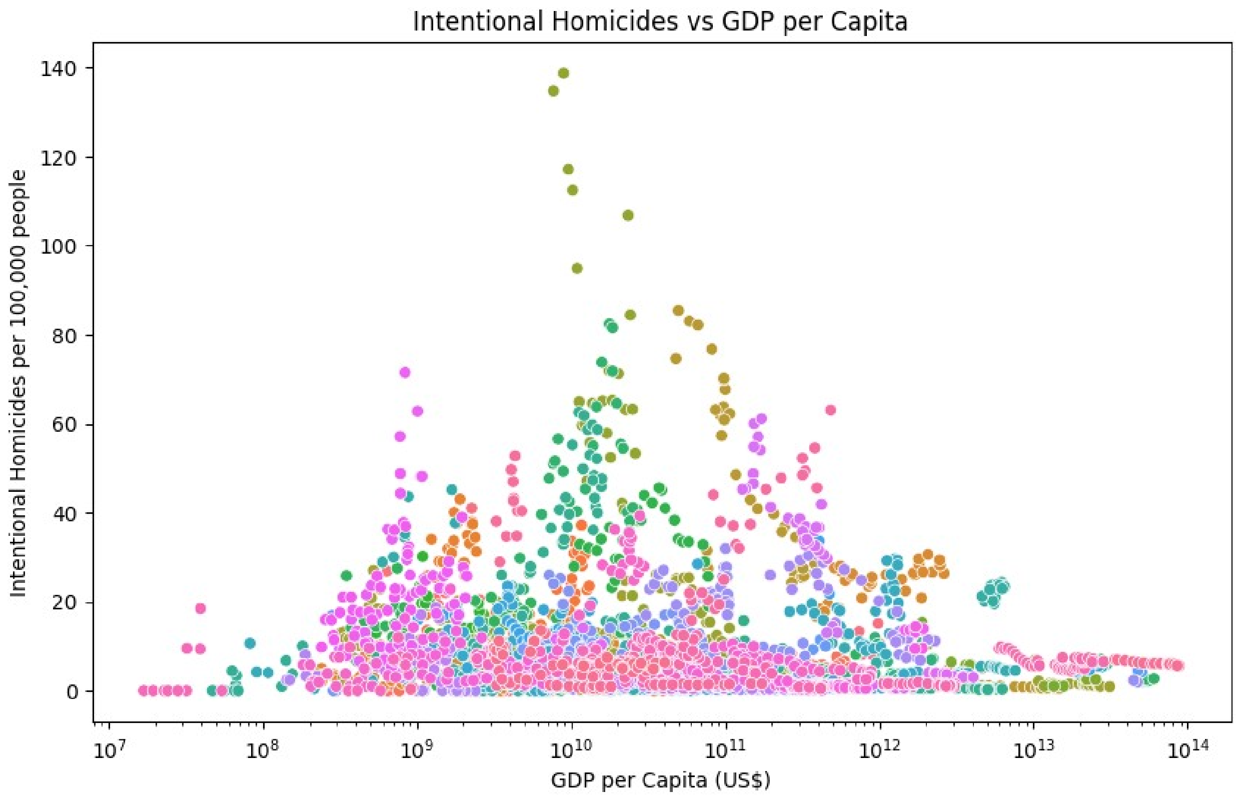
13) **Health Spending:** What is the relationship between government health expenditure and life expectancy?

```
plt.figure(figsize=(10,6))
sns.scatterplot(data=df, x='government_health_expenditure%',
y='life_expectancy_at_birth', hue='country', legend=False)
plt.title('Government Health Expenditure vs Life Expectancy')
plt.xlabel('Health Expenditure (% of GDP)')
plt.ylabel('Life Expectancy at Birth')
plt.show()
```



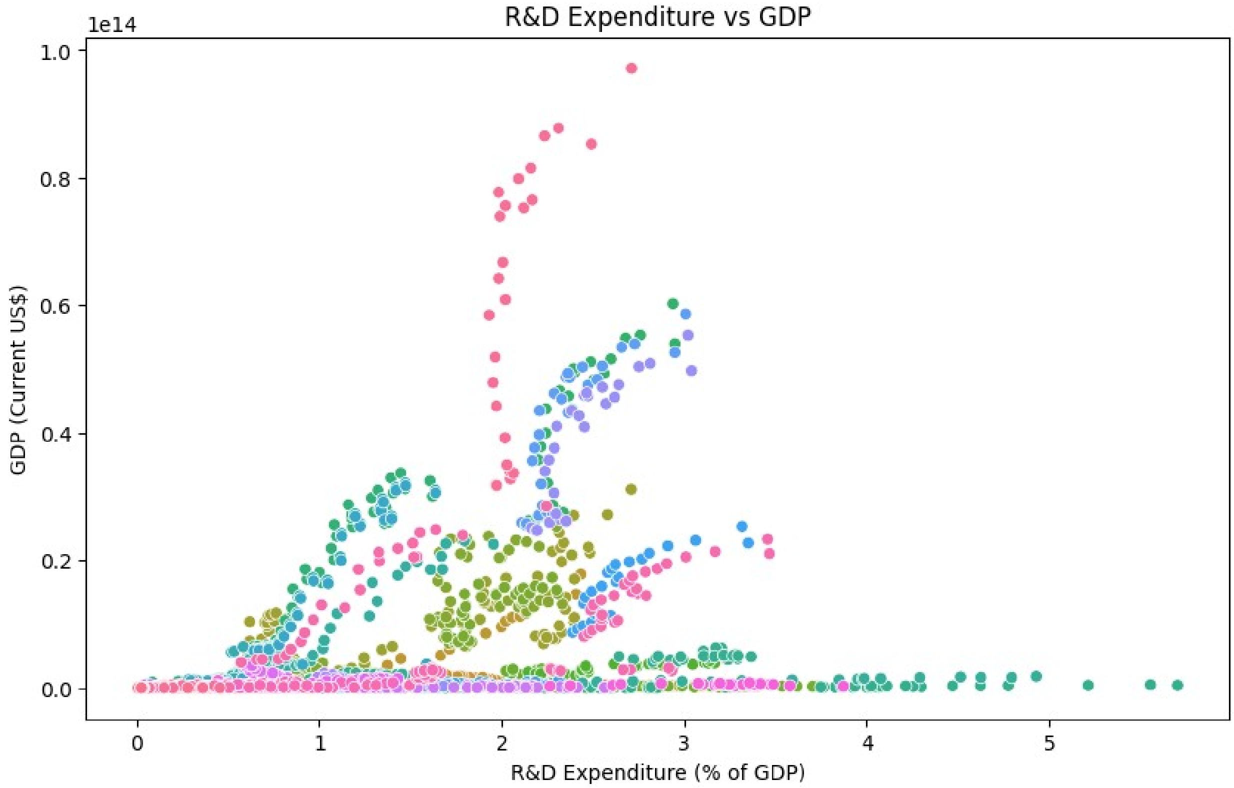
14) **Safety and Wealth:** Is there a correlation between intentional homicide rates and GDP per capita?

```
plt.figure(figsize=(10,6))
sns.scatterplot(data=df, x='GDP_current_US',
y='intentional_homicides', hue='country', legend=False)
plt.xscale('log')
plt.title('Intentional Homicides vs GDP per Capita')
plt.xlabel('GDP per Capita (US$)')
plt.ylabel('Intentional Homicides per 100,000 people')
plt.show()
```



15) **Research & Innovation:** How does investment in R&D relate to national economic output?

```
plt.figure(figsize=(10,6))
sns.scatterplot(data=df, x='research_and_development_expenditure%',
y='GDP_current_US', hue='country', legend=False)
plt.title('R&D Expenditure vs GDP')
plt.xlabel('R&D Expenditure (% of GDP)')
plt.ylabel('GDP (Current US$)')
plt.show()
```


R&D Expenditure vs GDP
1e14
1.0
0.8
0.6
0.4
0.2
0.0
GDP (Current US$)
0
1
2
3
4
5
R&D Expenditure (% of GDP)