```
from google.colab import drive

drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
import pandas as pd
df = pd.read_csv('/content/drive/MyDrive/Coffee_domestic_consumption.csv')
```

```
pd.set_option('display.max_columns', None)
```

```
# Dropping rows where Total_domestic_consumption is 0
df = df[df['Total_domestic_consumption'] != 0]
# Reset the index
df = df.reset_index(drop=True)

# Check the resulting DataFrame
print(df)
```

25	19020000	19020000	19020000	19020000	410260140
26	12000000	12000000	12000000	11880000	359880000
27	17520000	18000000	18000000	17520000	417242040
28	214260000	218580000	221100000	226860000	4536540000
29	30000	30000	30000	30000	1123140
30	120000	120000	120000	120000	9970800
31	23400000	23700000	23700000	23580000	590880000
32	3000000	3000000	3000000	3000000	86730000
33	540000	540000	540000	540000	9203040
34	22200000	22500000	22500000	21000000	471850680
35	86400000	88200000	88500000	87000000	2093460000
36	540000	540000	540000	540000	18688020
37	3600000	3900000	4200000	4290000	95190000
38	9000000	9300000	9300000	9420000	157980000
39	300000	300000	300000	300000	8640000
40	141600000	144000000	147000000	145500000	3189660000
41	13003680	13080000	13200000	12600000	299700300
42	2400000	2400000	2400000	2400000	70740000
43	4020000	4020000	4020000	4020000	122916960
44	300000	300000	300000	300000	10080000
45	2100000	2100000	2100000	2100000	58300020
46	81000000	82500000	84000000	84000000	1248600000
47	15240	15240	15240	15600	2167620
48	600000	600000	600000	600000	21090000
49	14400000	14688000	15000000	15240000	284816400
50	99000000	96000000	93000000	76500000	2386067999
51	144000000	150000000	156000000	159000000	1920928320
52	6000000	5400000	3900000	3840000	121620000

```
# Dropping rows where any column contains the value "0"
df = df[(df != 0).all(axis=1)]

# Reset the index
df = df.reset_index(drop=True)

# Check the resulting DataFrame
print(df)
```

7	60000	60000	60000	60000	2340000
8	120000	120000	120000	120000	3608400
9	1200000	1200000	1200000	1200000	35100000
10	15000000	15000000	15000000	15000000	402000000
11	60000	60000	60000	60000	2139960
12	240000	240000	240000	240000	8595960
13	180000	180000	180000	180000	5360040
14	13200000	13200000	13200000	12960000	384006000
15	23400000	23400000	23400000	22620000	642823380
16	20400000	20400000	20700000	20700000	600600000
17	181800000	190800000	198000000	195000000	2807280000
18	5478000	5500020	5520000	5700000	76425060
19	4620000	4680000	4680000	4620000	143450940
20	1200000	1200000	1200000	1200000	24794400
21	104186484	107596260	107484600	121486440	2536776384
22	19767420	21200040	21900000	21120000	665335200
23	19020000	19020000	19020000	19020000	410260140
24	12000000	12000000	12000000	11880000	359880000
25	17520000	18000000	18000000	17520000	417242040
26	214260000	218580000	221100000	226860000	4536540000
27	30000	30000	30000	30000	1123140
28	120000	120000	120000	120000	9970800
29	23400000	23700000	23700000	23580000	590880000
30	3000000	3000000	3000000	3000000	86730000
31	22200000	22500000	22500000	21000000	471850680
32	86400000	88200000	88500000	87000000	2093460000
33	540000	540000	540000	540000	18688020
34	3600000	3900000	4200000	4290000	95190000
35	300000	300000	300000	300000	8640000
36	141600000	144000000	147000000	145500000	3189660000
37	13003680	13080000	13200000	12600000	299700300
38	2400000	2400000	2400000	2400000	70740000
39	4020000	4020000	4020000	4020000	122916960
40	300000	300000	300000	300000	10080000
41	2100000	2100000	2100000	2100000	58300020
42	81000000	82500000	84000000	84000000	1248600000
43	15240	15240	15240	15600	2167620
44	600000	600000	600000	600000	21090000
45	14400000	14688000	15000000	15240000	284816400
46	99000000	96000000	93000000	76500000	2386067999
47	144000000	150000000	156000000	159000000	1920928320

```
df.head()
```

	Country	Coffee type	1990/91	1991/92	1992/93	1993/94	1994/95
0	Angola	Robusta/Arabica	1200000	1800000	2100000	1200000	1500000
1	Bolivia (Plurinational State of)	Arabica	1500000	1620000	1650000	1710000	1770000
2	Brazil	Arabica/Robusta	492000000	510000000	534000000	546000000	558000000
3	Burundi	Arabica/Robusta	120000	96000	102000	114600	120000
4	Ecuador	Arabica/Robusta	21000000	21000000	21000000	21000000	21000000

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48 entries, 0 to 47
Data columns (total 33 columns):
 #   Column                       Non-Null Count  Dtype 
---  ------                       --------------  ----- 
 0   Country                      48 non-null     object
 1   Coffee type                  48 non-null     object
 2   1990/91                      48 non-null     int64 
 3   1991/92                      48 non-null     int64 
 4   1992/93                      48 non-null     int64 
 5   1993/94                      48 non-null     int64 
 6   1994/95                      48 non-null     int64 
 7   1995/96                      48 non-null     int64 
 8   1996/97                      48 non-null     int64 
 9   1997/98                      48 non-null     int64 
 10  1998/99                      48 non-null     int64 
 11  1999/00                      48 non-null     int64 
 12  2000/01                      48 non-null     int64 
 13  2001/02                      48 non-null     int64 
 14  2002/03                      48 non-null     int64 
 15  2003/04                      48 non-null     int64 
 16  2004/05                      48 non-null     int64 
 17  2005/06                      48 non-null     int64 
 18  2006/07                      48 non-null     int64 
 19  2007/08                      48 non-null     int64 
 20  2008/09                      48 non-null     int64 
 21  2009/10                      48 non-null     int64 
 22  2010/11                      48 non-null     int64 
 23  2011/12                      48 non-null     int64 
 24  2012/13                      48 non-null     int64 
 25  2013/14                      48 non-null     int64 
 26  2014/15                      48 non-null     int64 
 27  2015/16                      48 non-null     int64 
 28  2016/17                      48 non-null     int64 
 29  2017/18                      48 non-null     int64 
 30  2018/19                      48 non-null     int64 
 31  2019/20                      48 non-null     int64 
 32  Total_domestic_consumption   48 non-null     int64 
```

```
dtypes: int64(31), object(2)
memory usage: 12.5+ KB
```

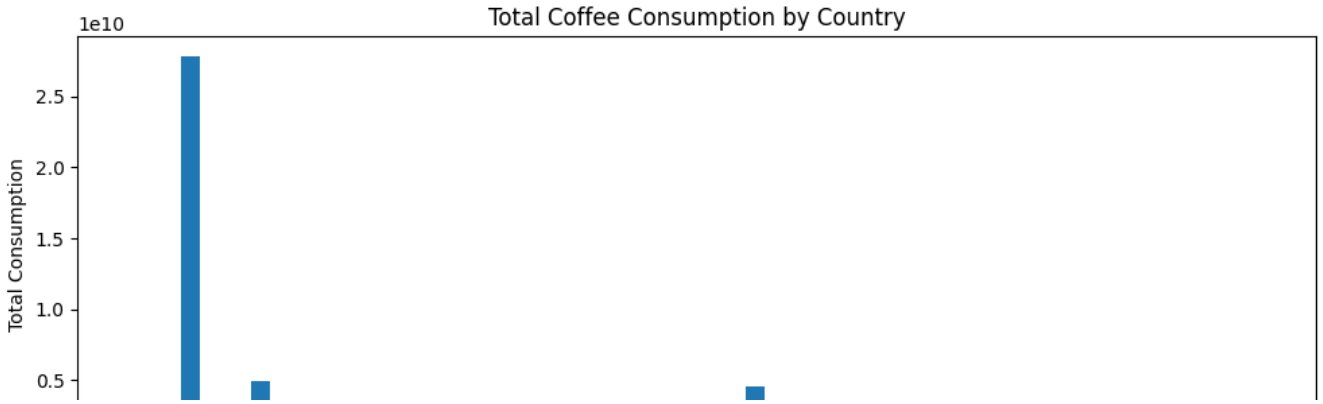
```
df.describe()
```

	1990/91	1991/92	1992/93	1993/94	1994/95	1995/96
count	4.800000e+01	4.800000e+01	4.800000e+01	4.800000e+01	4.800000e+01	4.800000e+01
mean	2.438464e+07	2.528325e+07	2.618752e+07	2.637113e+07	2.669833e+07	2.825246e+07
std	7.246522e+07	7.483134e+07	7.835745e+07	7.995455e+07	8.175599e+07	8.861621e+07
min	6.000000e+04	6.000000e+04	4.800000e+04	4.800000e+04	6.000000e+04	4.800000e+04
25%	5.250000e+05	6.825000e+05	8.250000e+05	9.150000e+05	9.000000e+05	7.620000e+05
50%	3.890010e+06	4.260000e+06	3.978000e+06	4.260000e+06	4.700010e+06	4.700010e+06
75%	1.830000e+07	1.830000e+07	1.935000e+07	1.935000e+07	1.890000e+07	2.010000e+07
max	4.920000e+08	5.100000e+08	5.340000e+08	5.460000e+08	5.580000e+08	6.060000e+08

There's steady increase in average consumption of coffee annually.

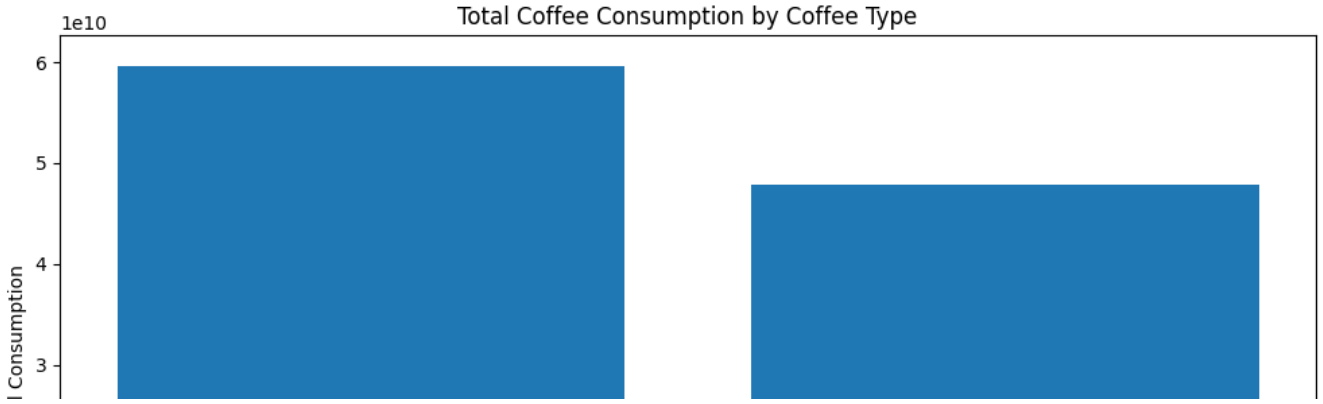
The significant standard deviations indicate variability in consumption levels from year to year. This variability could be influenced by factors such as economic conditions, coffee type, price fluctuations, or changes in consumer behavior.

```
import matplotlib.pyplot as plt

# Plotting
plt.figure(figsize=(10, 6))
plt.bar(df['Country'], df['Total_domestic_consumption'])
plt.xlabel('Country')
plt.ylabel('Total Consumption')
plt.title('Total Coffee Consumption by Country')
plt.xticks(rotation=90)  # Rotate x-axis labels for better readability
plt.tight_layout()  # Adjust layout to prevent clipping of labels
plt.show()
```



Brazil consumes the most amount of coffee by a very wide margin. Brazil's rich coffee heritage, abundant production, economic significance, diverse culture, and effective promotion contribute to its high levels of coffee consumption.

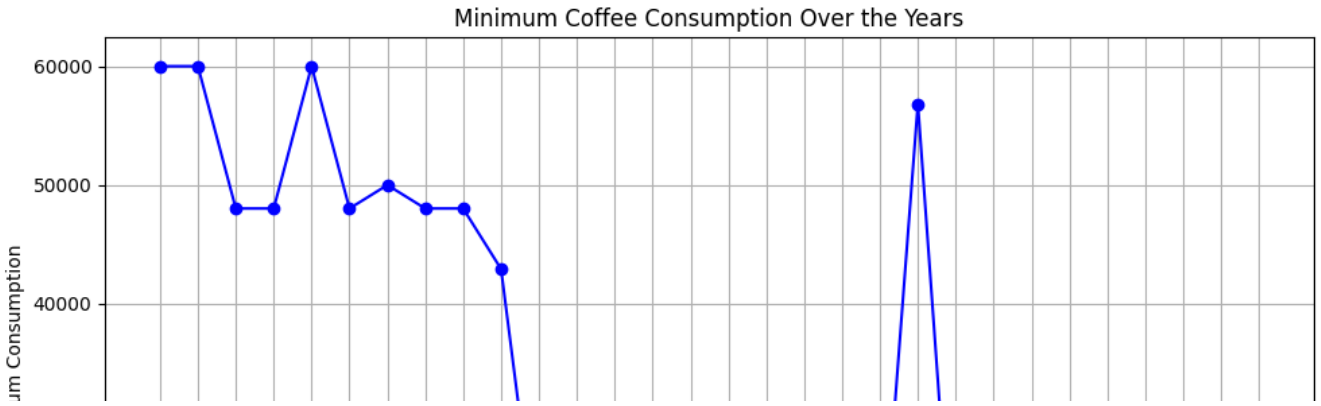
```
import matplotlib.pyplot as plt

# Split rows with multiple coffee types into separate rows
split_df = df.assign(Coffee_type=df['Coffee type'].str.split('/')).explode('Coffee_type')

# Group by coffee type and calculate total consumption
grouped_df = split_df.groupby('Coffee_type')['Total_domestic_consumption'].sum().reset_ind

# Plotting
plt.figure(figsize=(10, 6))
plt.bar(grouped_df['Coffee_type'], grouped_df['Total_domestic_consumption'])
plt.xlabel('Coffee Type')
plt.ylabel('Total Consumption')
plt.title('Total Coffee Consumption by Coffee Type')
plt.xticks(rotation=45)  # Rotate x-axis labels for better readability
plt.tight_layout()  # Adjust layout to prevent clipping of labels
plt.show()
```



Arabica is the most consumed coffee type

FURTHER EDA

```
import matplotlib.pyplot as plt

# Extracting the years and minimum consumption
years = df.columns[2:-1]  # Extracting the years from the column names
min_consumption = df.iloc[:, 2:-1].min()  # Calculating the minimum consumption for each y

# Plotting
plt.figure(figsize=(10, 6))
plt.plot(years, min_consumption, marker='o', linestyle='-', color='blue')
plt.xlabel('Years')
plt.ylabel('Minimum Consumption')
plt.title('Minimum Coffee Consumption Over the Years')
plt.xticks(rotation=45)  # Rotate x-axis labels for better readability
plt.grid(True)
plt.tight_layout()
plt.show()
```



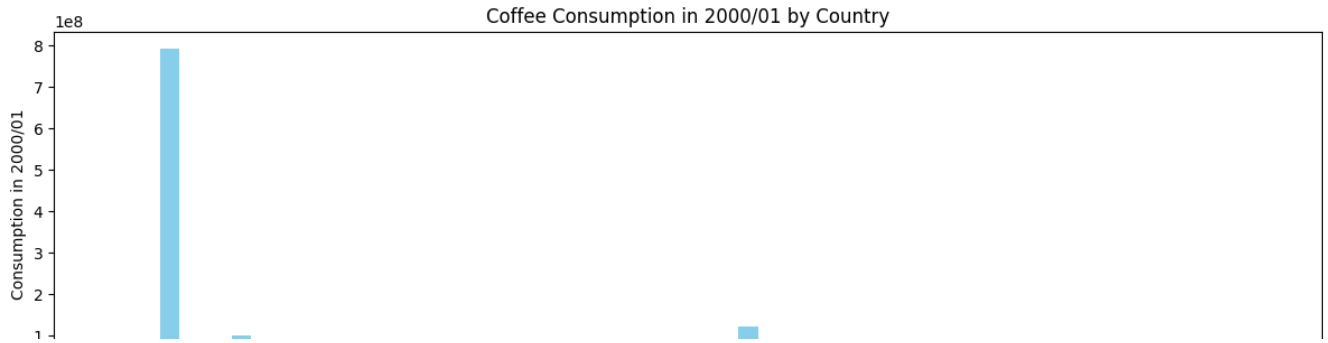
There was a sharp drop in minimum amount of coffee consumed in 2000/01, a sharp rise in 2010/11 and a sharp drop again in 2011/12.

```
# Sort the DataFrame by consumption in 2000/01 in ascending order
sorted_df = df.sort_values(by='2000/01')

# Get the country with the least consumption in 2000/01
country_least_consumption_2000_01 = sorted_df.iloc[0]['Country']

print("Country with the least coffee consumption in 2000/01:", country_least_consumption_2
```

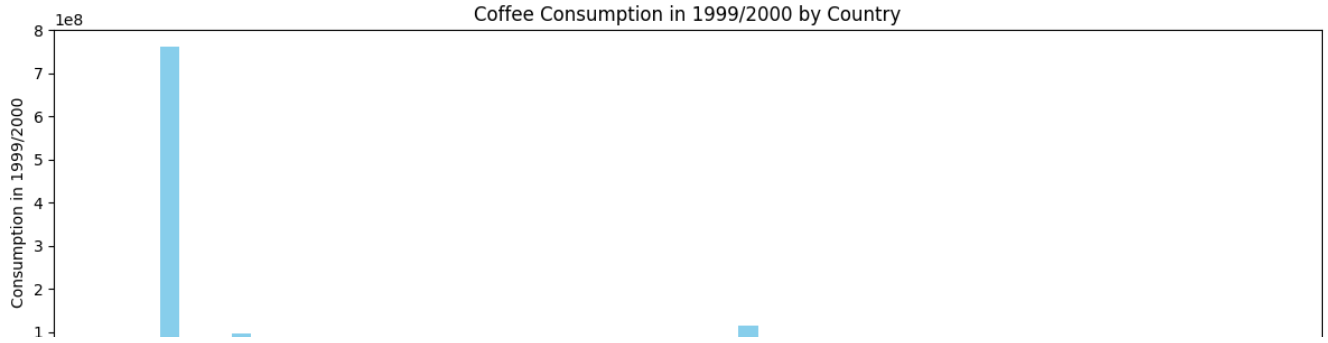
```
Country with the least coffee consumption in 2000/01: Gabon
```

```
import matplotlib.pyplot as plt

# Extracting the country names and consumption in 2000/01
countries = df['Country']
consumption_2000_01 = df['2000/01']
```

```
# Plotting
plt.figure(figsize=(12, 6))
plt.bar(countries, consumption_2000_01, color='skyblue')
plt.xlabel('Country')
plt.ylabel('Consumption in 2000/01')
plt.title('Coffee Consumption in 2000/01 by Country')
plt.xticks(rotation=90)  # Rotate x-axis labels for better readability
plt.tight_layout()
plt.show()
```




```
import matplotlib.pyplot as plt

# Extracting the country names and consumption in 1999/2000
countries = df['Country']
consumption_1999_2000 = df['1999/00']

# Plotting
plt.figure(figsize=(12, 6))
plt.bar(countries, consumption_1999_2000, color='skyblue')
plt.xlabel('Country')
plt.ylabel('Consumption in 1999/2000')
plt.title('Coffee Consumption in 1999/2000 by Country')
plt.xticks(rotation=90)  # Rotate x-axis labels for better readability
plt.tight_layout()
plt.show()
```



Country with the least coffee consumption in 2000/01 was Gabon, this may have been as a result of the fall in coffee price between 2000 and 2004. lower coffee prices could have negatively impacted coffee-producing regions, including Gabon. If coffee farmers in Gabon experienced reduced income due to lower prices, it could have had ripple effects on the local economy, leading to decreased purchasing power and reduced coffee consumption among the population.

I also compared the coffee consumption in all countries for 2000/01 and the past year (1999/00) to check for any change in pattern for other countries. There was no change in pattern, only Gabon was deeply affected.

```
# Selecting columns from '2000/01' to '2011/12'
consumption_columns = df.columns[2:-1]

# Dictionary to store the country with the lowest consumption for each year
min_consumption_countries = {}

# Iterating over each year and finding the country with the lowest consumption
for year in consumption_columns:
    min_country = df.loc[df[year].idxmin()]['Country']
    min_consumption_countries[year] = min_country

# Printing the results
for year, country in min_consumption_countries.items():
    print(f"Year: {year}, Country with the lowest consumption: {country}")
```

```
Year: 1990/91, Country with the lowest consumption: Rwanda
Year: 1991/92, Country with the lowest consumption: Rwanda
Year: 1992/93, Country with the lowest consumption: Gabon
Year: 1993/94, Country with the lowest consumption: Gabon
Year: 1994/95, Country with the lowest consumption: Rwanda
Year: 1995/96, Country with the lowest consumption: Gabon
Year: 1996/97, Country with the lowest consumption: Gabon
Year: 1997/98, Country with the lowest consumption: Gabon
Year: 1998/99, Country with the lowest consumption: Gabon
Year: 1999/00, Country with the lowest consumption: Gabon
Year: 2000/01, Country with the lowest consumption: Gabon
Year: 2001/02, Country with the lowest consumption: Gabon
Year: 2002/03, Country with the lowest consumption: Gabon
Year: 2003/04, Country with the lowest consumption: Gabon
Year: 2004/05, Country with the lowest consumption: Gabon
Year: 2005/06, Country with the lowest consumption: Gabon
Year: 2006/07, Country with the lowest consumption: Gabon
Year: 2007/08, Country with the lowest consumption: Gabon
Year: 2008/09, Country with the lowest consumption: Gabon
Year: 2009/10, Country with the lowest consumption: Gabon
Year: 2010/11, Country with the lowest consumption: Gabon
Year: 2011/12, Country with the lowest consumption: Togo
Year: 2012/13, Country with the lowest consumption: Togo
Year: 2013/14, Country with the lowest consumption: Togo
Year: 2014/15, Country with the lowest consumption: Togo
Year: 2015/16, Country with the lowest consumption: Togo
Year: 2016/17, Country with the lowest consumption: Togo
Year: 2017/18, Country with the lowest consumption: Togo
Year: 2018/19, Country with the lowest consumption: Togo
Year: 2019/20, Country with the lowest consumption: Togo
```

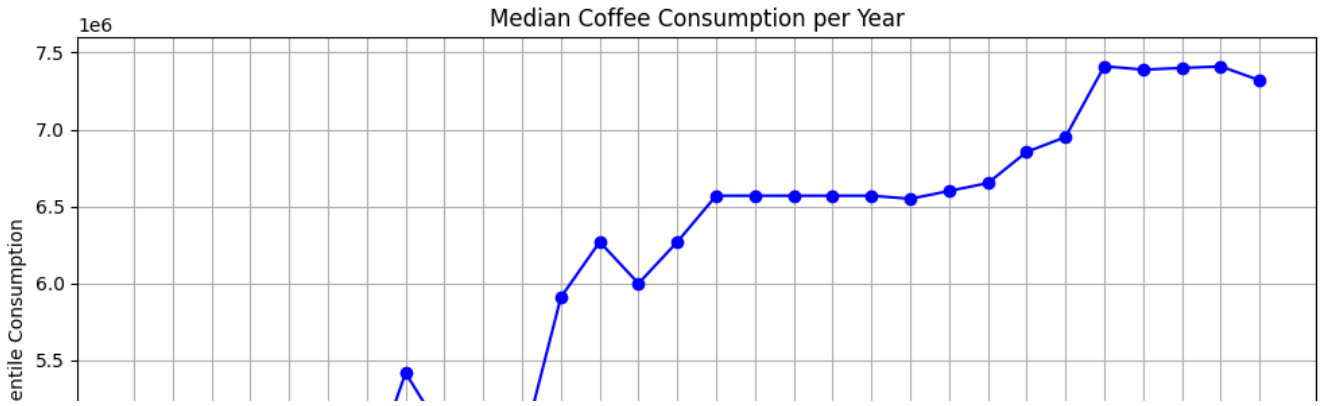
```
# Filter the DataFrame to get Gabon's row
gabon_row = df[df['Country'] == 'Gabon']
```

```
# Get Gabon's consumption in 2011/12
gabon_consumption_2011_12 = gabon_row['2011/12'].values[0]

print("Gabon's coffee consumption in 2011/12:", gabon_consumption_2011_12)
```

```
Gabon's coffee consumption in 2011/12: 61920
```

The minimum coffee consumption spiked in 2010/11 with Gabon, then dropped sharply again in 2011/12 with Togo. Togo took the baton as the country with the lowest coffee consumption from 2011/12 till the end date of the dataset (2019/20). Gabon's coffee consumption in 2011/12 was far from the minimum.

```
import matplotlib.pyplot as plt

# Extracting the years and the 50th percentile values
years = df.columns[2:-1]  # Extracting the years from the column names
median_consumption = df.iloc[:, 2:-1].apply(lambda x: x.median())  # Calculating the media

# Plotting
plt.figure(figsize=(10, 6))
plt.plot(years, median_consumption, marker='o', linestyle='-', color='blue')
plt.xlabel('Years')
plt.ylabel('50th Percentile Consumption')
plt.title('Median Coffee Consumption per Year')
plt.xticks(rotation=45)  # Rotate x-axis labels for better readability
plt.grid(True)
plt.tight_layout()
plt.show()
```

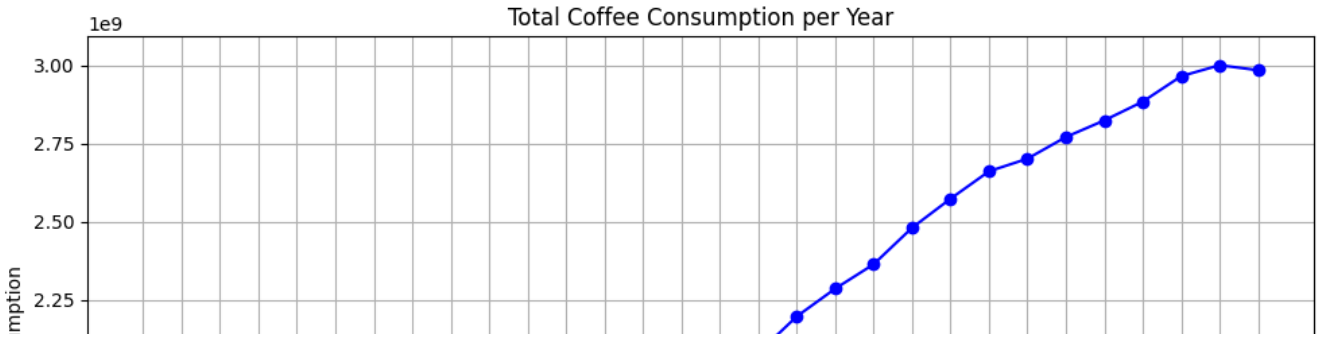


there was a notable drop in coffee consumption in 1999/00 which didn't fully recover until 2001/02. The 50th percentile value (median) dropped significantly. This indicates that half of the countries experienced a decrease in consumption during that period. This drop in consumption could have been influenced by various factors such as the drop in worldwide coffee price in 2000 and the early 2000 economic recession. Surprisingly enough, the 2008 economic recession did not have much effect on median coffee consumption, it had a little bit of effect on the minimum coffee consumption.

```
data = {
    'Country': ['Angola', 'Bolivia', 'Brazil', 'Burundi', 'Ecuador', 'Indonesia', 'Madagas
    'Latitude': [-11.2027, -16.2902, -14.2350, -3.3731, -1.8312, -0.7893, -18.7669, -13.25
    'Longitude': [17.8739, -63.5887, -51.9253, 29.9189, -78.1834, 113.9213, 46.8691, 34.30
}

df2 = pd.DataFrame(data)
import folium

# Create a map centered around the mean of latitudes and longitudes
m = folium.Map(location=[df2['Latitude'].mean(), df2['Longitude'].mean()], zoom_start=2)

# Add markers for each country
for index, row in df2.iterrows():
    folium.Marker(location=[row['Latitude'], row['Longitude']], popup=row['Country']).add_

# Display the map
m
```

Map showing the countries in the dataset

```
import matplotlib.pyplot as plt

# Extracting the years and total consumption
years = df.columns[2:-1]  # Extracting the years from the column names
total_consumption_per_year = df.iloc[:, 2:-1].sum()  # Calculating the total consumption f

# Plotting
plt.figure(figsize=(10, 6))
plt.plot(years, total_consumption_per_year, marker='o', linestyle='-', color='blue')
plt.xlabel('Years')
plt.ylabel('Total Consumption')
plt.title('Total Coffee Consumption per Year')
plt.xticks(rotation=45)  # Rotate x-axis labels for better readability
plt.grid(True)
plt.tight_layout()
plt.show()
```


Total Coffee Consumption per Year
1e9
3.00
2.75
2.50
2.25
mption