

TASK-01


Create a bar chart or histogram to visualize the distribution of a categorical or continuous variable, such as the distribution of ages or genders in a population. Dataset :- <https://data.worldbank.org/indicator/SP.POP.TOTL>

IMPORTED REQUIRED LIBRARIES

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

BROUGHT DATA TO ON BOARD

```
from google.colab import files
data=files.upload()
```

 Choose Files

No file chosen

Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

Saving WorldBank.csv to WorldBank.csv

VIEWING THE DATASET

```
df=pd.read_csv('WorldBank.csv')
df
```

	Country Name	Country Code	Indicator Name	Indicator Code	1960	1961	1962	1963	1964	1965	...
0	Aruba	ABW	Population, total	SP.POP.TOTL	54608.0	55811.0	56682.0	57475.0	58178.0	58782.0	...
1	Africa Eastern and Southern	AFE	Population, total	SP.POP.TOTL	130692579.0	134169237.0	137835590.0	141630546.0	145605995.0	149742351.0	... 56
2	Afghanistan	AFG	Population, total	SP.POP.TOTL	8622466.0	8790140.0	8969047.0	9157465.0	9355514.0	9565147.0	... 3
3	Africa Western and Central	AFW	Population, total	SP.POP.TOTL	97256290.0	99314028.0	101445032.0	103667517.0	105959979.0	108336203.0	... 36
4	Angola	AGO	Population, total	SP.POP.TOTL	5357195.0	5441333.0	5521400.0	5599827.0	5673199.0	5736582.0	... 2
...
261	Kosovo	XKX	Population, total	SP.POP.TOTL	947000.0	966000.0	994000.0	1022000.0	1050000.0	1078000.0	...
262	Yemen, Rep.	YEM	Population, total	SP.POP.TOTL	5542459.0	5646668.0	5753386.0	5860197.0	5973803.0	6097298.0	... 2
263	South Africa	ZAF	Population, total	SP.POP.TOTL	16520441.0	16989464.0	17503133.0	18042215.0	18603097.0	19187194.0	... 5
264	Zambia	ZMB	Population, total	SP.POP.TOTL	3119430.0	3219451.0	3323427.0	3431381.0	3542764.0	3658024.0	... 1
265	Zimbabwe	ZWE	Population, total	SP.POP.TOTL	3806310.0	3925952.0	4049778.0	4177931.0	4310332.0	4447149.0	... 1

266 rows × 67 columns

OBSERVING THE RECORD AND FEATURE

```
df.shape
(266, 67)
```

VIEWING FIRST 10 RECORD

```
df.head(10)
```

	Country Name	Country Code	Indicator Name	Indicator Code	1960	1961	1962	1963	1964	1965	...	
0	Aruba	ABW	Population, total	SP.POP.TOTL	54608.0	55811.0	56682.0	57475.0	58178.0	58782.0	...	1
1	Africa Eastern and Southern	AFE	Population, total	SP.POP.TOTL	130692579.0	134169237.0	137835590.0	141630546.0	145605995.0	149742351.0	...	5678
2	Afghanistan	AFG	Population, total	SP.POP.TOTL	8622466.0	8790140.0	8969047.0	9157465.0	9355514.0	9565147.0	...	315
3	Africa Western and Central	AFW	Population, total	SP.POP.TOTL	97256290.0	99314028.0	101445032.0	103667517.0	105959979.0	108336203.0	...	3872
4	Angola	AGO	Population, total	SP.POP.TOTL	5357195.0	5441333.0	5521400.0	5599827.0	5673199.0	5736582.0	...	261
5	Albania	ALB	Population, total	SP.POP.TOTL	1608800.0	1659800.0	1711319.0	1762621.0	1814135.0	1864791.0	...	28
6	Andorra	AND	Population, total	SP.POP.TOTL	9443.0	10216.0	11014.0	11839.0	12690.0	13563.0	...	
7	Arab World	ARB	Population, total	SP.POP.TOTL	93359407.0	95760348.0	98268683.0	100892507.0	103618568.0	106444103.0	...	3891
8	United Arab Emirates	ARE	Population, total	SP.POP.TOTL	133426.0	140984.0	148877.0	157006.0	165305.0	173797.0	...	87
9	Argentina	ARG	Population, total	SP.POP.TOTL	20349744.0	20680653.0	21020359.0	21364017.0	21708487.0	22053661.0	...	422

10 rows × 67 columns

VIEWNING LAST 10 RECORD

```
df.tail(10)
```

	Country Name	Country Code	Indicator Name	Indicator Code	1960	1961	1962	1963	1964	1965	.
256	Virgin Islands (U.S.)	VIR	Population, total	SP.POP.TOTL	3.250000e+04	3.430000e+04	3.500000e+04	3.980000e+04	4.080000e+04	4.350000e+04	
257	Viet Nam	VNM	Population, total	SP.POP.TOTL	3.271846e+07	3.362198e+07	3.453389e+07	3.552673e+07	3.650917e+07	3.746608e+07	
258	Vanuatu	VUT	Population, total	SP.POP.TOTL	6.460800e+04	6.646200e+04	6.839100e+04	7.040000e+04	7.249300e+04	7.467700e+04	
259	World	WLD	Population, total	SP.POP.TOTL	3.031474e+09	3.072422e+09	3.126850e+09	3.193429e+09	3.260442e+09	3.328209e+09	
260	Samoa	WSM	Population, total	SP.POP.TOTL	1.133350e+05	1.168200e+05	1.201630e+05	1.234160e+05	1.265820e+05	1.297890e+05	
261	Kosovo	XKX	Population, total	SP.POP.TOTL	9.470000e+05	9.660000e+05	9.940000e+05	1.022000e+06	1.050000e+06	1.078000e+06	
262	Yemen, Rep.	YEM	Population, total	SP.POP.TOTL	5.542459e+06	5.646668e+06	5.753386e+06	5.860197e+06	5.973803e+06	6.097298e+06	
263	South Africa	ZAF	Population, total	SP.POP.TOTL	1.652044e+07	1.698946e+07	1.750313e+07	1.804222e+07	1.860310e+07	1.918719e+07	
264	Zambia	ZMB	Population, total	SP.POP.TOTL	3.119430e+06	3.219451e+06	3.323427e+06	3.431381e+06	3.542764e+06	3.658024e+06	
265	Zimbabwe	ZWE	Population, total	SP.POP.TOTL	3.806310e+06	3.925952e+06	4.049778e+06	4.177931e+06	4.310332e+06	4.447149e+06	

10 rows × 67 columns

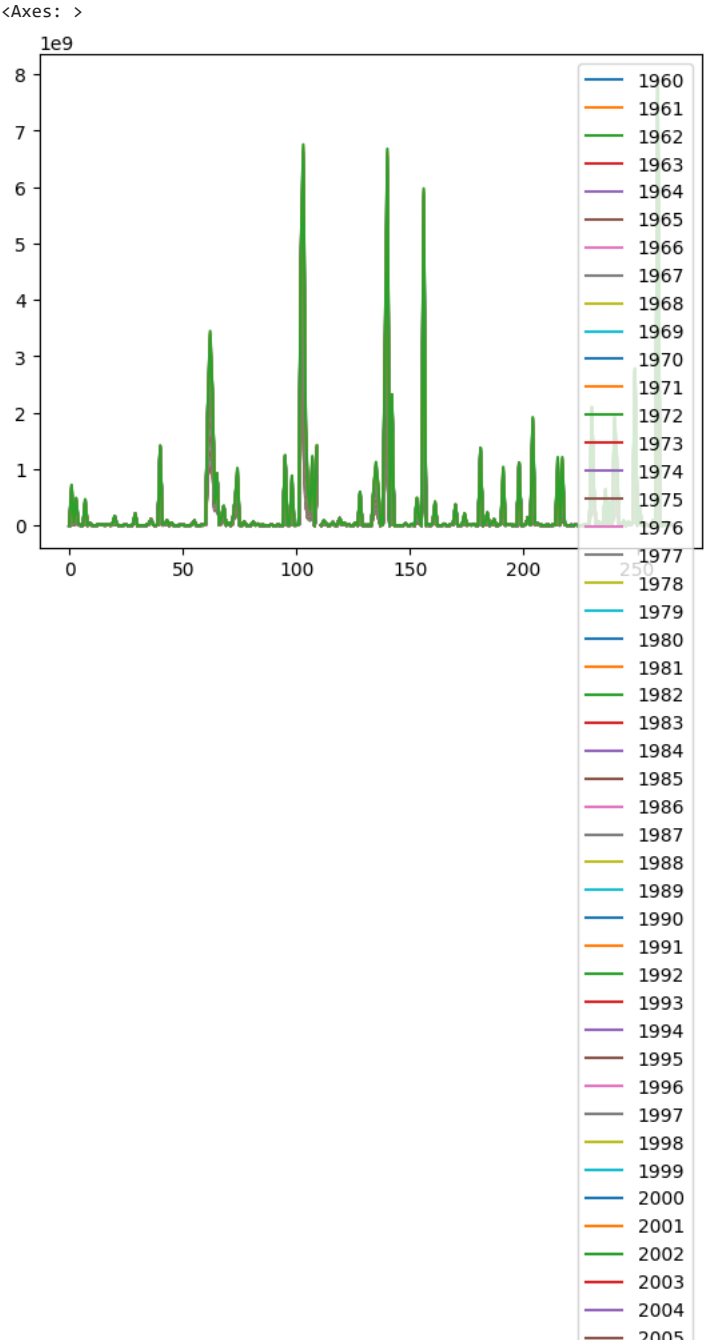
VIEWING THE OVER ALL COLUMNS

```
df.columns
```

```
Index(['Country Name', 'Country Code', 'Indicator Name', 'Indicator Code',  
      '1960', '1961', '1962', '1963', '1964', '1965', '1966', '1967', '1968',  
      '1969', '1970', '1971', '1972', '1973', '1974', '1975', '1976', '1977',  
      '1978', '1979', '1980', '1981', '1982', '1983', '1984', '1985', '1986',
```

```
'1987', '1988', '1989', '1990', '1991', '1992', '1993', '1994', '1995',
'1996', '1997', '1998', '1999', '2000', '2001', '2002', '2003', '2004',
'2005', '2006', '2007', '2008', '2009', '2010', '2011', '2012', '2013',
'2014', '2015', '2016', '2017', '2018', '2019', '2020', '2021', '2022'],
dtype='object')
```

```
df.plot()
```



OBSERVING WHICH COLUMNS COMES UNDER STRING(OBJECT)/WHICH COLUMN COMES UNDER INT,FLOAT.

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 266 entries, 0 to 265
Data columns (total 67 columns):
#   Column          Non-Null Count  Dtype
---  -
0   Country Name    266 non-null   object
1   Country Code    266 non-null   object
2   Indicator Name  266 non-null   object
3   Indicator Code  266 non-null   object
4   1960            264 non-null   float64
5   1961            264 non-null   float64
6   1962            264 non-null   float64
7   1963            264 non-null   float64
8   1964            264 non-null   float64
9   1965            264 non-null   float64
10  1966            264 non-null   float64
11  1967            264 non-null   float64
```

12	1968	264 non-null	float64
13	1969	264 non-null	float64
14	1970	264 non-null	float64
15	1971	264 non-null	float64
16	1972	264 non-null	float64
17	1973	264 non-null	float64
18	1974	264 non-null	float64
19	1975	264 non-null	float64
20	1976	264 non-null	float64
21	1977	264 non-null	float64
22	1978	264 non-null	float64
23	1979	264 non-null	float64
24	1980	264 non-null	float64
25	1981	264 non-null	float64
26	1982	264 non-null	float64
27	1983	264 non-null	float64
28	1984	264 non-null	float64
29	1985	264 non-null	float64
30	1986	264 non-null	float64
31	1987	264 non-null	float64
32	1988	264 non-null	float64
33	1989	264 non-null	float64
34	1990	265 non-null	float64
35	1991	265 non-null	float64
36	1992	265 non-null	float64
37	1993	265 non-null	float64
38	1994	265 non-null	float64
39	1995	265 non-null	float64
40	1996	265 non-null	float64
41	1997	265 non-null	float64
42	1998	265 non-null	float64
43	1999	265 non-null	float64
44	2000	265 non-null	float64
45	2001	265 non-null	float64
46	2002	265 non-null	float64
47	2003	265 non-null	float64
48	2004	265 non-null	float64
49	2005	265 non-null	float64
50	2006	265 non-null	float64
51	2007	265 non-null	float64
52	2008	265 non-null	float64

VIEWING THE MINI STATISTICAL REPORT

```
df.describe()
```

	1960	1961	1962	1963	1964	1965	1966	1967	1968	
count	2.640000e+02	2.640000e+02	2.640000e+02	2.640000e+02	2.640000e+02	2.640000e+02	2.640000e+02	2.640000e+02	2.640000e+02	2.640000e+02
mean	1.172712e+08	1.188807e+08	1.210511e+08	1.237333e+08	1.264378e+08	1.291813e+08	1.320404e+08	1.348980e+08	1.378358e+08	1.407333e+08
std	3.695439e+08	3.740897e+08	3.808061e+08	3.895039e+08	3.982439e+08	4.071153e+08	4.164504e+08	4.257424e+08	4.353218e+08	4.450934e+08
min	2.646000e+03	2.888000e+03	3.171000e+03	3.481000e+03	3.811000e+03	4.161000e+03	4.531000e+03	4.930000e+03	5.354000e+03	5.773872e+03
25%	5.132212e+05	5.231345e+05	5.337595e+05	5.449288e+05	5.566630e+05	5.651150e+05	5.691470e+05	5.773872e+05	5.832700e+05	5.888000e+05
50%	3.757486e+06	3.887144e+06	4.023896e+06	4.139356e+06	4.224612e+06	4.277636e+06	4.331825e+06	4.385700e+06	4.450934e+06	4.509340e+06
75%	2.670606e+07	2.748694e+07	2.830289e+07	2.914708e+07	3.001684e+07	3.084892e+07	3.163010e+07	3.209247e+07	3.249927e+07	3.291684e+07
max	3.031474e+09	3.072422e+09	3.126850e+09	3.193429e+09	3.260442e+09	3.328209e+09	3.398480e+09	3.468371e+09	3.540164e+09	3.615556e+09

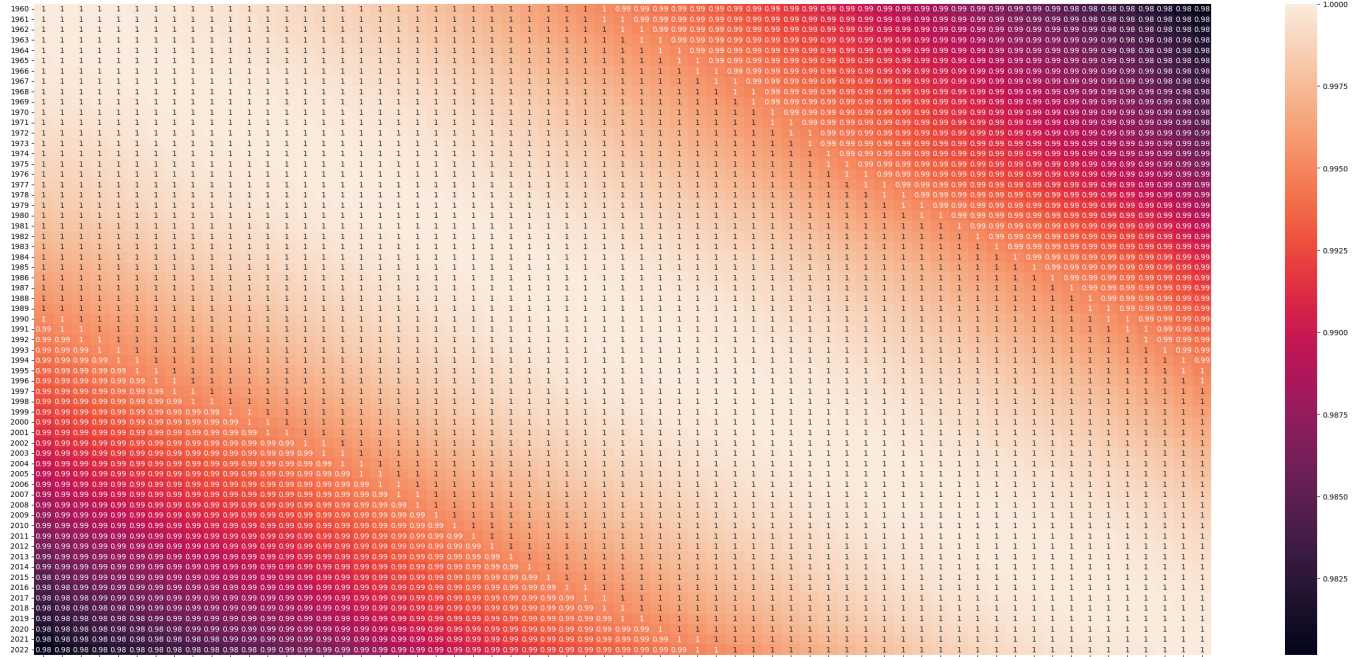
8 rows × 63 columns

OBSERVING THE CORRELATION OF THE GIVEN DATA

```
plt.figure(figsize=(36,16))
sns.heatmap(df.corr(),annot=True)
```

```
<ipython-input-29-0f4404c9a57a>: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future ver  
sns.heatmap(df.corr(),annot=True)
```

<Axes: >



DATA CLEANING

```
pd.options.display.max_rows = 70  
df.isnull().sum()
```

Country Name	0
Country Code	0
Indicator Name	0
Indicator Code	0
1960	2
1961	2
1962	2
1963	2
1964	2
1965	2
1966	2
1967	2
1968	2
1969	2
1970	2
1971	2
1972	2
1973	2
1974	2
1975	2
1976	2
1977	2
1978	2
1979	2
1980	2
1981	2
1982	2
1983	2
1984	2
1985	2
1986	2
1987	2
1988	2
1989	2
1990	1
1991	1

1992	1
1993	1
1994	1
1995	1
1996	1
1997	1
1998	1
1999	1
2000	1
2001	1
2002	1
2003	1
2004	1
2005	1
2006	1
2007	1
2008	1
2009	1
2010	1
2011	1
2012	1
2013	1

HANDLING THE MISSING VALUES

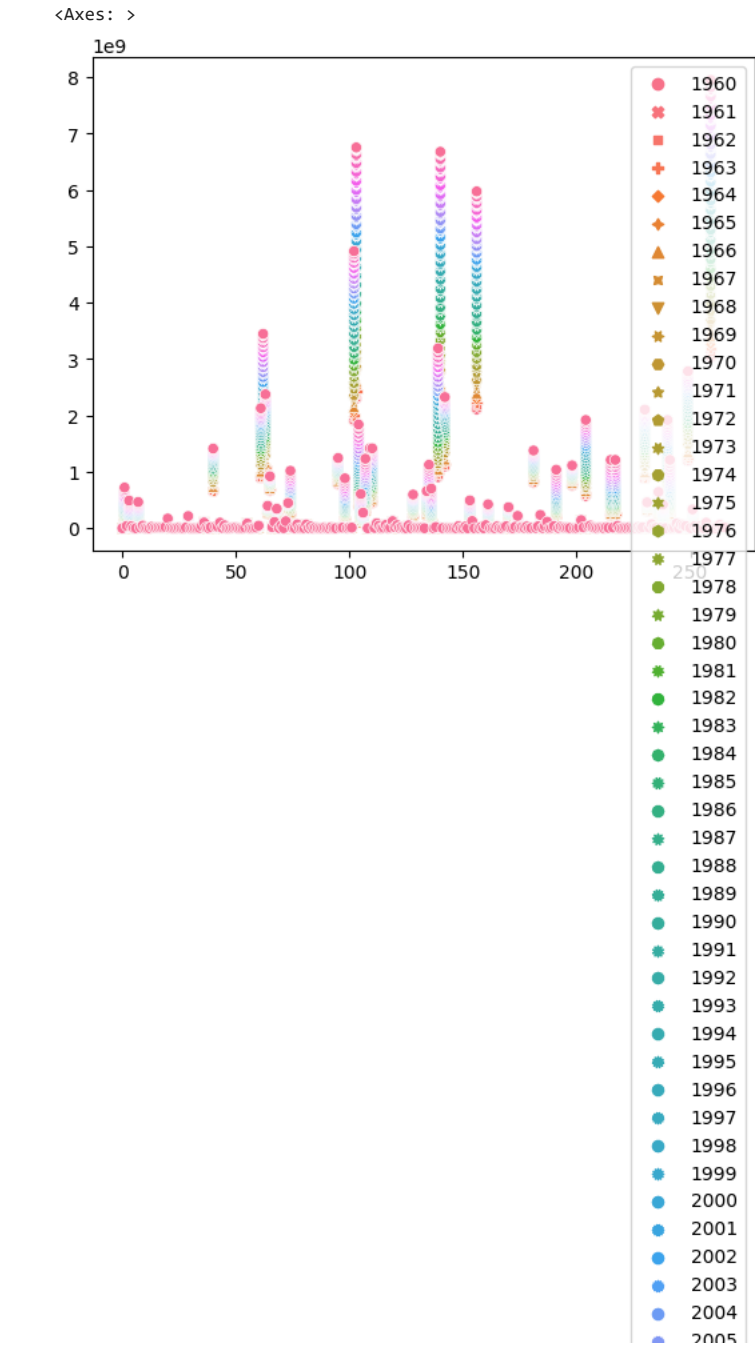
```
df = df.fillna(method="ffill")
```

```
df.isnull().sum()
```

Country Name	0
Country Code	0
Indicator Name	0
Indicator Code	0
1960	0
1961	0
1962	0
1963	0
1964	0
1965	0
1966	0
1967	0
1968	0
1969	0
1970	0
1971	0
1972	0
1973	0
1974	0
1975	0
1976	0
1977	0
1978	0
1979	0
1980	0
1981	0
1982	0
1983	0
1984	0
1985	0
1986	0
1987	0
1988	0
1989	0
1990	0
1991	0
1992	0
1993	0
1994	0
1995	0
1996	0
1997	0
1998	0
1999	0
2000	0
2001	0
2002	0
2003	0
2004	0
2005	0
2006	0
2007	0
2008	0
2009	0
2010	0
2011	0
2012	0
2013	0

DATA VISUALIZATION

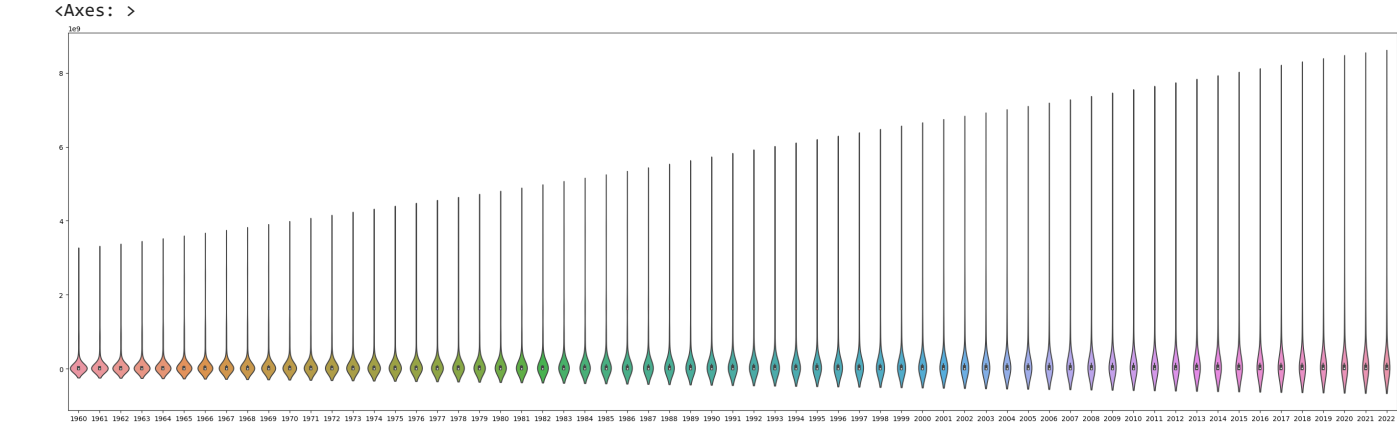
```
sns.scatterplot(df)
```



VIOLINE PLOT

```
dim = (35, 10)
fig, ab = plt.subplots(figsize=dim)
sns.violinplot(ab=ab, data=df)
```

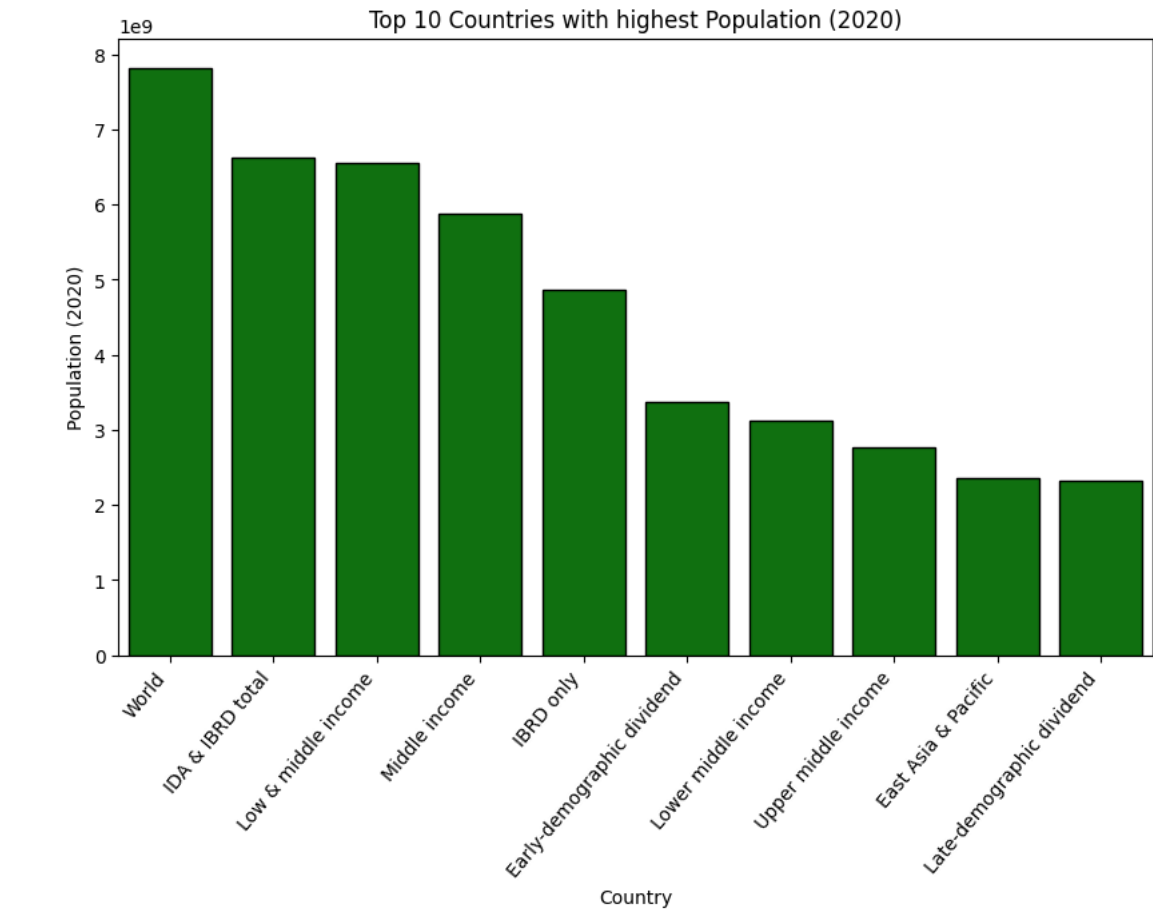




VIEWING TOP 10 COUNTRIES IN THE GIVEN DATA

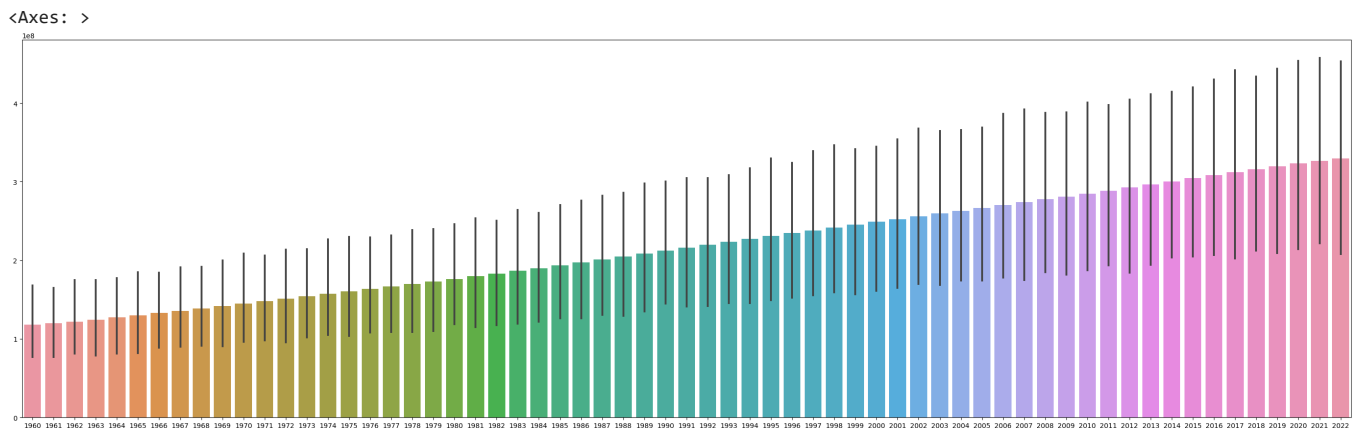
```
def plot_top_countries(df, column, ascending=True,color='green',title=''):
    df_sort = df.sort_values(column, ascending=ascending).head(10)
    plt.figure(figsize=(10, 6))
    sns.barplot(x=df_sort['Country Name'],y=df_sort[column], color=color, edgecolor='black')
    plt.xlabel('Country')
    plt.ylabel(f'Population ({column})')
    plt.title(title)
    plt.xticks(rotation=50, ha='right')

plot_top_countries(df,'2020',ascending=False,title='Top 10 Countries with highest Population (2020)')
```



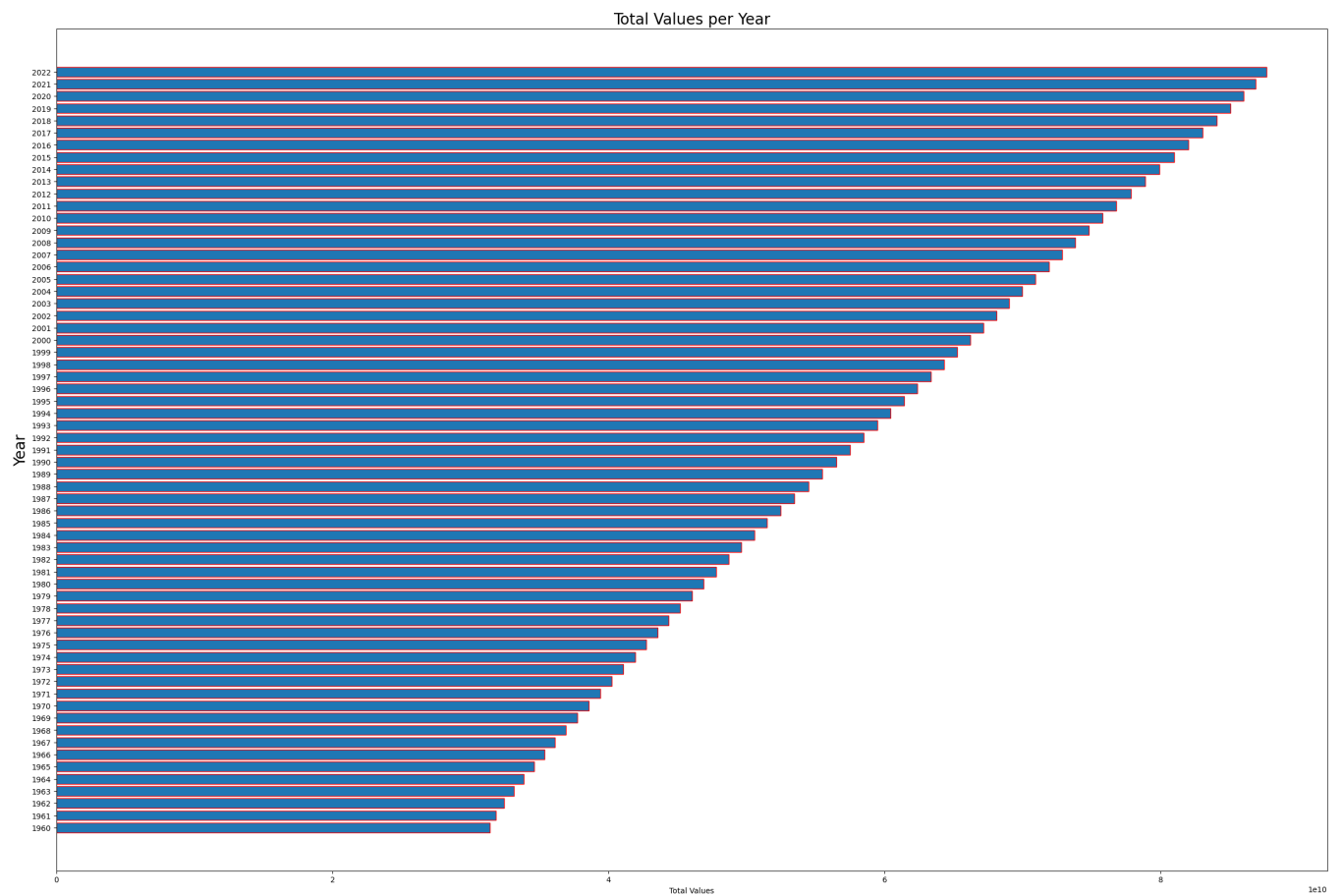
BAR PLOT


```
dim2 = (35, 10)
fig, ax = plt.subplots(figsize=dim2)
sns.barplot(ax=ax,data=df)
```



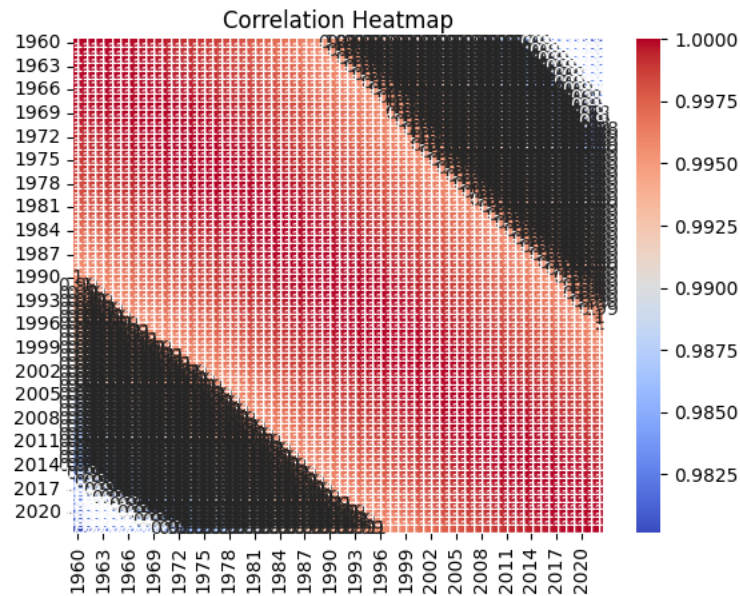
```
years = ['1960', '1961', '1962', '1963', '1964', '1965', '1966', '1967', '1968',
        '1969', '1970', '1971', '1972', '1973', '1974', '1975', '1976', '1977',
        '1978', '1979', '1980', '1981', '1982', '1983', '1984', '1985', '1986',
        '1987', '1988', '1989', '1990', '1991', '1992', '1993', '1994', '1995',
        '1996', '1997', '1998', '1999', '2000', '2001', '2002', '2003', '2004',
        '2005', '2006', '2007', '2008', '2009', '2010', '2011', '2012', '2013',
        '2014', '2015', '2016', '2017', '2018', '2019', '2020', '2021', '2022']
```

```
years = df.columns[4:]
values = df[years].sum()
plt.figure(figsize=(30,20))
plt.barh(years,values,edgecolor='red')
plt.xlabel('Total Values')
plt.ylabel('Year', size=20)
plt.title('Total Values per Year', size=20)
plt.show()
```



```
correlation_matrix = df.corr()
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```

```
<ipython-input-23-182fd031f822>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future ver
correlation_matrix = df.corr()
```



```
total_population = df.sort_values(by='2022',ascending=False)
total_population
```

	Country Name	Country Code	Indicator Name	Indicator Code	1960	1961	1962	1963	1964	1965	...
259	World	WLD	Population, total	SP.POP.TOTL	3.031474e+09	3.072422e+09	3.126850e+09	3.193429e+09	3.260442e+09	3.328209e+09	..
103	IDA & IBRD total	IBT	Population, total	SP.POP.TOTL	2.297972e+09	2.329504e+09	2.374276e+09	2.431314e+09	2.488809e+09	2.547220e+09	..
140	Low & middle income	LMY	Population, total	SP.POP.TOTL	2.243179e+09	2.274050e+09	2.318173e+09	2.374537e+09	2.431355e+09	2.489219e+09	..
156	Middle income	MIC	Population, total	SP.POP.TOTL	2.107416e+09	2.135116e+09	2.175940e+09	2.228877e+09	2.282094e+09	2.336193e+09	..
102	IBRD only	IBD	Population, total	SP.POP.TOTL	1.904347e+09	1.926043e+09	1.960606e+09	2.007061e+09	2.053555e+09	2.100537e+09	..
...
147	St. Martin (French part)	MAF	Population, total	SP.POP.TOTL	4.135000e+03	4.258000e+03	4.388000e+03	4.524000e+03	4.666000e+03	4.832000e+03	..
255	British Virgin Islands	VGB	Population, total	SP.POP.TOTL	7.850000e+03	7.885000e+03	7.902000e+03	7.919000e+03	7.949000e+03	8.018000e+03	..
188	Palau	PLW	Population, total	SP.POP.TOTL	9.446000e+03	9.639000e+03	9.851000e+03	1.007600e+04	1.031800e+04	1.056300e+04	..
179	Nauru	NRU	Population, total	SP.POP.TOTL	4.582000e+03	4.753000e+03	4.950000e+03	5.198000e+03	5.484000e+03	5.804000e+03	..
245	Tuvalu	TUV	Population, total	SP.POP.TOTL	5.404000e+03	5.436000e+03	5.471000e+03	5.503000e+03	5.525000e+03	5.548000e+03	..

266 rows × 67 columns

CHECKING THE UNIQUE OF THE COLUMN

```
df['Country Name'].unique()

array(['Aruba', 'Africa Eastern and Southern', 'Afghanistan',
'Africa Western and Central', 'Angola', 'Albania', 'Andorra',
'Arab World', 'United Arab Emirates', 'Argentina', 'Armenia',
'American Samoa', 'Antigua and Barbuda', 'Australia', 'Austria',
'Azerbaijan', 'Burundi', 'Belgium', 'Benin', 'Burkina Faso',
'Bangladesh', 'Bulgaria', 'Bahrain', 'Bahamas, The',
'Bosnia and Herzegovina', 'Belarus', 'Belize', 'Bermuda',
```

```
'Bolivia', 'Brazil', 'Barbados', 'Brunei Darussalam', 'Bhutan',  
'Botswana', 'Central African Republic', 'Canada',  
'Central Europe and the Baltics', 'Switzerland', 'Channel Islands',  
'Chile', 'China', 'Cote d'Ivoire', 'Cameroon', 'Congo, Dem. Rep.',  
'Congo, Rep.', 'Colombia', 'Comoros', 'Cabo Verde', 'Costa Rica',
```