Double-click (or enter) to edit

Connecting to Google drive

```
from google.colab import drive
drive.mount('/content/drive')
```

```
    Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

Importing required libraries.

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

Reading the csv file (data):

```
d=pd.read_csv("/content/drive/MyDrive/Colab Notebooks/World-population-by-countries-dataset.csv")
d=d.dropna()
d.head()
```

	Country Name	Country Code	1960	1961	1962	1963	1964	1965	1966	1967	...	2012	2013	2014	2015	2016	
0	Aruba	ABW	54208.0	55434.0	56234.0	56699.0	57029.0	57357.0	57702.0	58044.0	...	102565.0	103165.0	103776.0	104339.0	104865.0	10
1	Africa Eastern and Southern	AFE	130836765.0	134159786.0	137614644.0	141202036.0	144920186.0	148769974.0	152752671.0	156876454.0	...	547482863.0	562601578.0	578075373.0	593871847.0	609978946.0	62639
2	Afghanistan	AFG	8996967.0	9169406.0	9351442.0	9543200.0	9744772.0	9956318.0	10174840.0	10399936.0	...	31161378.0	32269592.0	33370804.0	34413603.0	35383028.0	3629
3	Africa Western and Central	AFW	96396419.0	98407221.0	100506960.0	102691339.0	104953470.0	107289875.0	109701811.0	112195950.0	...	370243017.0	380437896.0	390882979.0	401586651.0	412551299.0	42376
4	Angola	AGO	5454938.0	5531451.0	5608499.0	5679409.0	5734995.0	5770573.0	5781305.0	5774440.0	...	25107925.0	26015786.0	26941773.0	27884380.0	28842482.0	2981

5 rows × 64 columns

```
d.tail()
```

	Country Name	Country Code	1960	1961	1962	1963	1964	1965	1966	1967	...	2012	2013	2014	2015	2016	2017	2
261	Kosovo	XKX	947000.0	966000.0	994000.0	1022000.0	1050000.0	1078000.0	1106000.0	1135000.0	...	1807106.0	1818117.0	1812771.0	1788196.0	1777557.0	1791003.0	17970
262	Yemen, Rep.	YEM	5315351.0	5393034.0	5473671.0	5556767.0	5641598.0	5727745.0	5816241.0	5907873.0	...	24473176.0	25147112.0	25823488.0	26497881.0	27168210.0	27834811.0	284986
263	South Africa	ZAF	17099836.0	17524533.0	17965733.0	18423157.0	18896303.0	19384838.0	19888259.0	20406863.0	...	52832659.0	53687125.0	54544184.0	55386369.0	56207649.0	57009751.0	577925
264	Zambia	ZMB	3070780.0	3164330.0	3260645.0	3360099.0	3463211.0	3570466.0	3681953.0	3797877.0	...	14465148.0	14926551.0	15399793.0	15879370.0	16363449.0	16853608.0	173517
265	Zimbabwe	ZWE	3776679.0	3905038.0	4039209.0	4178726.0	4322854.0	4471178.0	4623340.0	4779825.0	...	13115149.0	13350378.0	13586710.0	13814642.0	14030338.0	14236599.0	144388

5 rows × 64 columns

```
print(d.keys())
```

```
Index(['Country Name', 'Country 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'],
      dtype='object')
```

```
d[['Country Name','2021']]
```

	Country Name	2021
0	Aruba	107195.0
1	Africa Eastern and Southern	694665117.0
2	Afghanistan	39835428.0
3	Africa Western and Central	470898870.0
4	Angola	33933611.0
...	...	...
261	Kosovo	1806279.0
262	Yemen, Rep.	30490639.0
263	South Africa	60041996.0
264	Zambia	18920657.0
265	Zimbabwe	15092171.0

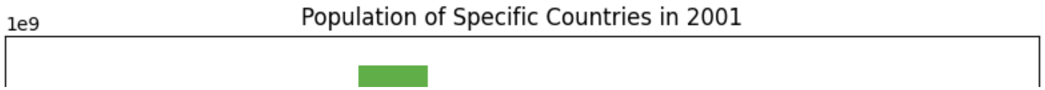
262 rows × 2 columns

```
d.describe()
```

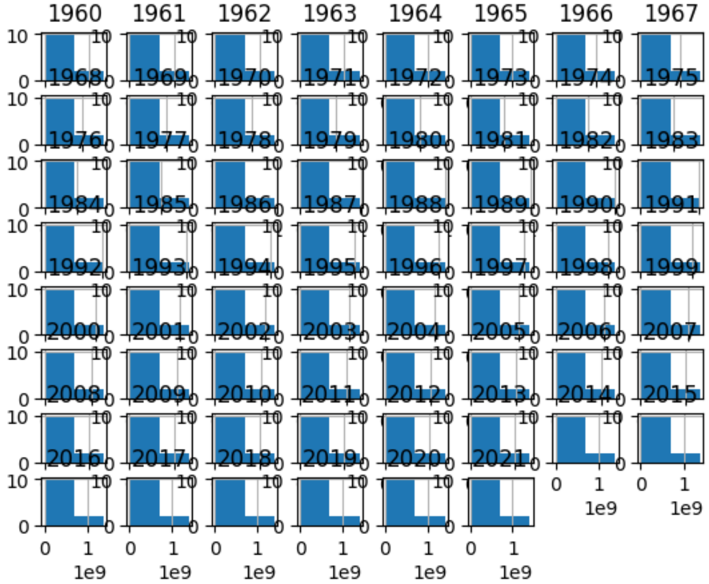
	1960	1961	1962	1963	1964	1965	1966	1967	1968	1969	...	2012	2013	2014	
count	2.620000e+02	2.620000e+02	2.620000e+02	2.620000e+02	2.620000e+02	2.620000e+02	2.620000e+02	2.620000e+02	2.620000e+02	2.620000e+02	...	2.620000e+02	2.620000e+02	2.620000e+02	2.620000e
mean	1.181073e+08	1.196648e+08	1.217890e+08	1.244281e+08	1.270884e+08	1.298145e+08	1.326840e+08	1.355457e+08	1.384770e+08	1.415136e+08	...	2.896568e+08	2.934915e+08	2.973388e+08	3.011787e
std	3.708464e+08	3.752044e+08	3.817404e+08	3.902524e+08	3.988079e+08	4.075981e+08	4.169476e+08	4.262349e+08	4.357762e+08	4.457073e+08	...	9.048559e+08	9.160772e+08	9.272859e+08	9.384288e
min	2.833000e+03	3.077000e+03	3.367000e+03	3.703000e+03	4.063000e+03	4.460000e+03	4.675000e+03	4.922000e+03	5.194000e+03	5.461000e+03	...	1.013600e+04	1.020800e+04	1.028900e+04	1.037400e
25%	5.149835e+05	5.257388e+05	5.372568e+05	5.462485e+05	5.555595e+05	5.652400e+05	5.752960e+05	5.858988e+05	5.973332e+05	6.074150e+05	...	1.409856e+06	1.427346e+06	1.444861e+06	1.463192e
50%	3.821420e+06	3.923497e+06	4.029190e+06	4.138240e+06	4.215254e+06	4.269828e+06	4.362524e+06	4.437342e+06	4.496863e+06	4.587932e+06	...	9.925639e+06	1.002641e+07	1.022601e+07	1.031988e
75%	2.654210e+07	2.731494e+07	2.810353e+07	2.897732e+07	2.988023e+07	3.079075e+07	3.157015e+07	3.203917e+07	3.273505e+07	3.321787e+07	...	6.266009e+07	6.315469e+07	6.364901e+07	6.401981e
max	3.032156e+09	3.071596e+09	3.124561e+09	3.189656e+09	3.255146e+09	3.322047e+09	3.392098e+09	3.461620e+09	3.532783e+09	3.606554e+09	...	7.089255e+09	7.175500e+09	7.261847e+09	7.347679e

Plotting a bar chart for "Population of Specific Countries in 2001".

```
countries = ['Angola', 'Kosovo', 'Zambia','South Africa','Zimbabwe','Aruba','Afghanistan','India','China','Brazil','Sri Lanka','Russia','Turkey','Japan']
filtered_data = d[d['Country Name'].isin(countries)]
plt.figure(figsize=(10, 6))
plt.title("Population of Specific Countries in 2001")
sns.barplot(x='Country Name', y='2001', data=filtered_data)
plt.ylabel("Population")
plt.xlabel("Countries")

plt.show()
```



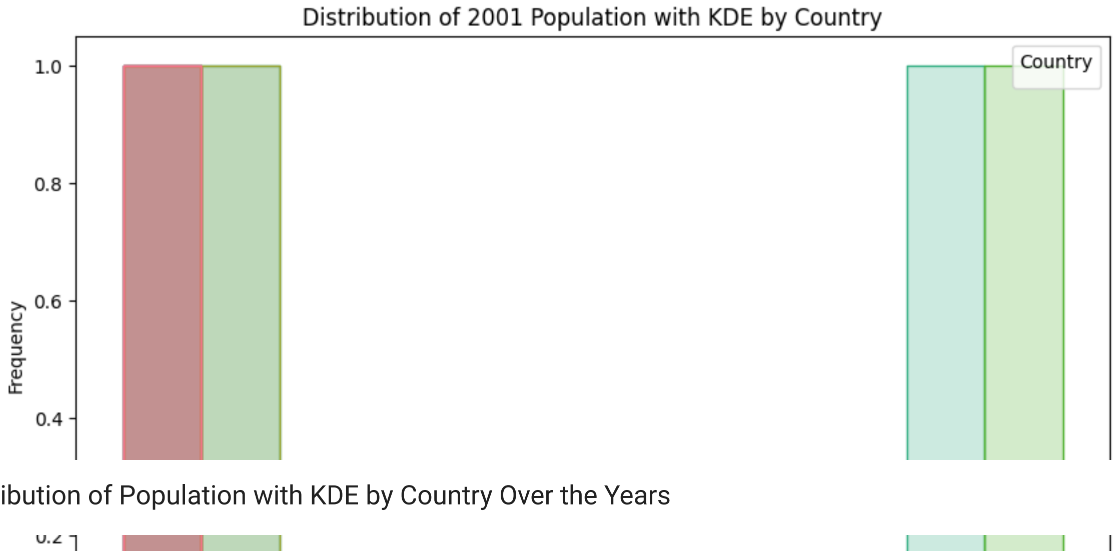
```
hist=filtered_data.hist(bins=2)
```



Distribution of 2001 Population with KDE by Country

```
plt.figure(figsize=(10, 6))
sns.histplot(data=filtered_data, x="2001", kde=True, hue="Country Name", element="step")

plt.title("Distribution of 2001 Population with KDE by Country")
plt.xlabel("Population in 2001")
plt.ylabel("Frequency")

plt.legend(title="Country")
plt.show()
```

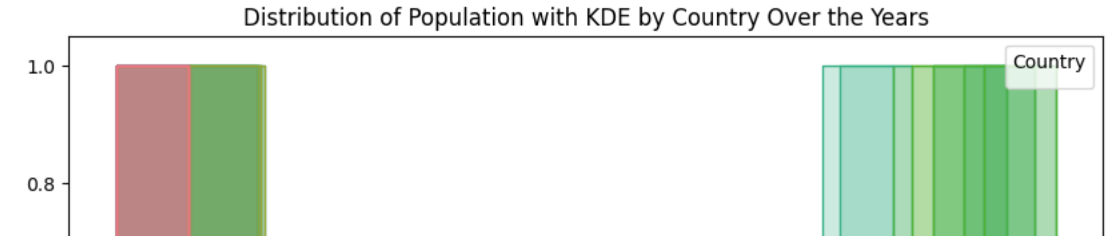
```
WARNING:matplotlib.legend:No artists with labels found to put in legend.  Note that artists whose label start with an underscore are ignored when legend() is called with no argument.
```



Distribution of Population with KDE by Country Over the Years

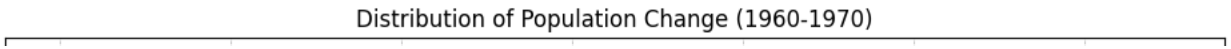
```
years = ['2001', '2005', '2010', '2015', '2020']

plt.figure(figsize=(10, 6))
for year in years:
    sns.histplot(data=filtered_data, x=year, kde=True, hue="Country Name", element="step")

plt.title("Distribution of Population with KDE by Country Over the Years")
plt.xlabel("Population")
plt.ylabel("Frequency")

plt.legend(title="Country")
plt.show()
```

```
WARNING:matplotlib.legend:No artists with labels found to put in legend.  Note that artists whose label start with an underscore are ignored when legend() is called with no argument.
```



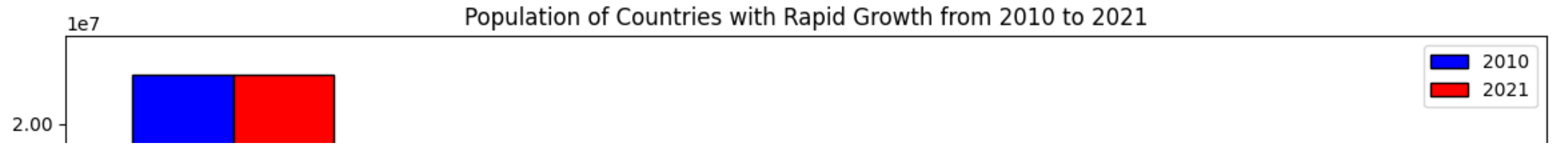
Distribution of Population Change (1960-1970)

```
d['population_change'] = d['1970'] - d['1960']

# Plotting the Histogram
plt.figure(figsize=(10,6))
plt.hist(d['population_change'], bins=10, color='blue', edgecolor='black')
plt.title('Distribution of Population Change (1960-1970)')
plt.xlabel('Population Change')
plt.ylabel('Number of Countries')
plt.grid(True, which="both", ls="--", linewidth=0.5)
plt.tight_layout()
plt.show()
```



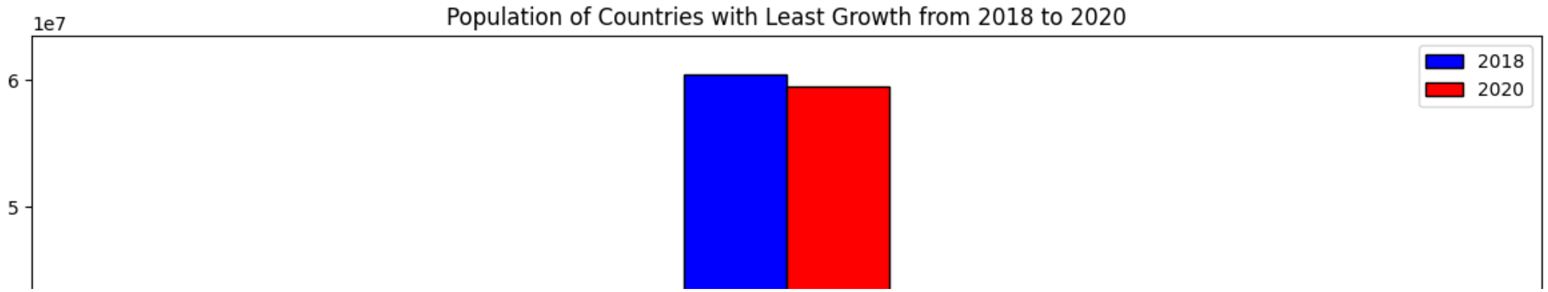
```
d['percentage_increase'] = ((d['2010'] - d['2021']) / d['2021']) * 100
# Get the top 5 countries with the highest percentage increase
top_countries = d.nlargest(5, 'percentage_increase')
```

```
#  Visualize
bar_width = 0.35
index = range(len(top_countries))

plt.figure(figsize=(12, 7))
bar1 = plt.bar(index, top_countries['2010'], bar_width, label='2010', color='b', edgecolor='black')
bar2 = plt.bar([i+bar_width for i in index], top_countries['2010'], bar_width, label='2021', color='r', edgecolor='black')

plt.xlabel('Country')
plt.ylabel('Population')
plt.title('Population of Countries with Rapid Growth from 2010 to 2021')
plt.xticks([i+bar_width/2 for i in index], top_countries['Country Name'])
plt.legend()
plt.tight_layout()
plt.show()
```



Population of Countries with Least Growth from 2018 to 2020

```
d['percentage_increase'] = ((d['2020'] - d['2018']) / (d['2018'] + 1e-10)) * 100

# Get the bottom 5 countries with the smallest percentage increase (including decreases)
bottom_countries = d.nsmallest(5, 'percentage_increase')

#  Visualize
bar_width = 0.35
index = range(len(bottom_countries))

plt.figure(figsize=(12, 7))
bar1 = plt.bar(index, bottom_countries['2018'], bar_width, label='2018', color='b', edgecolor='black')
bar2 = plt.bar([i+bar_width for i in index], bottom_countries['2020'], bar_width, label='2020', color='r', edgecolor='black')

plt.xlabel('Country')
plt.ylabel('Population')
plt.title('Population of Countries with Least Growth from 2018 to 2020')
plt.xticks([i+bar_width/2 for i in index], bottom_countries['Country Name'])
plt.legend()
plt.tight_layout()
plt.show()
```

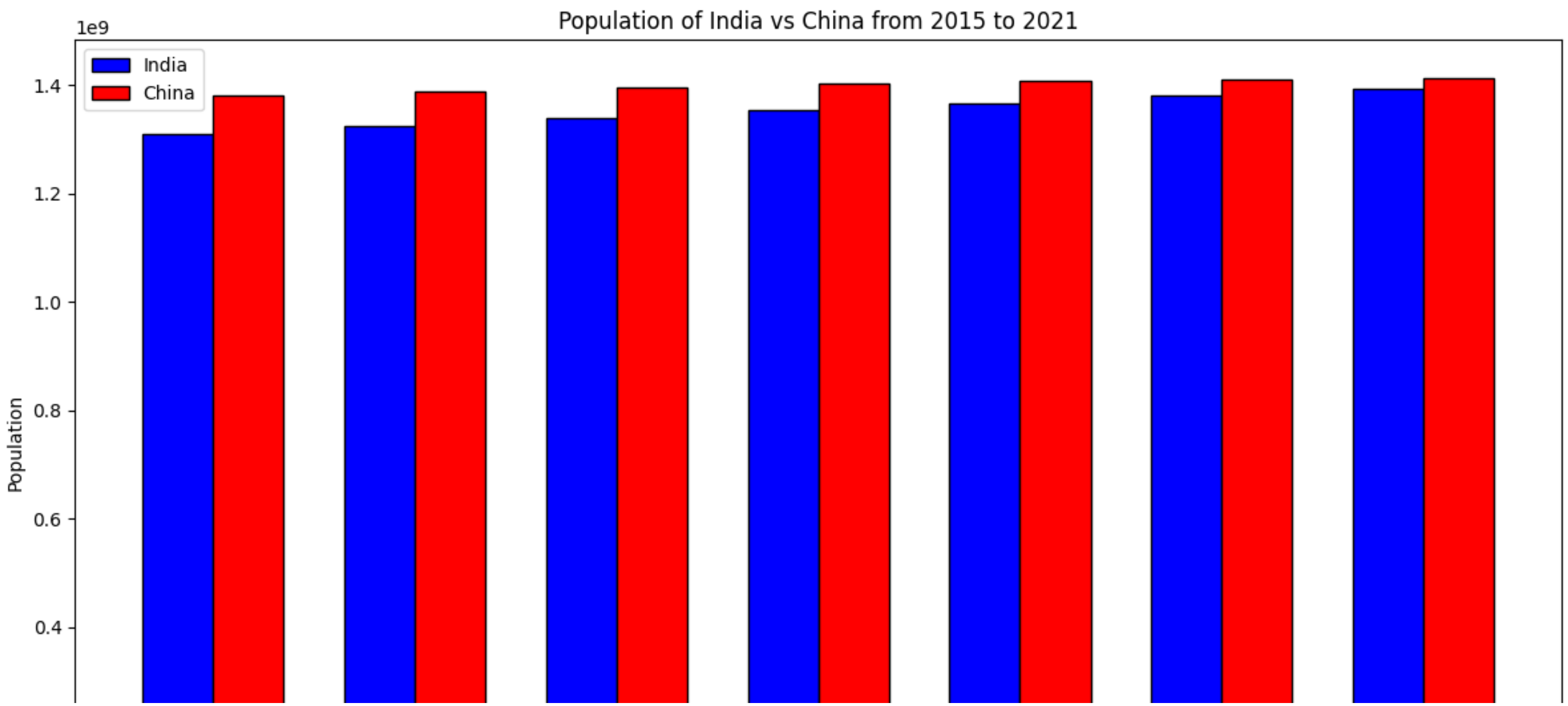


```
d['population_change_2019_2020'] = d['2020'] - d['2019']
country_with_largest_dip = d.nsmallest(1, 'population_change_2019_2020')

print(country_with_largest_dip[['Country Name', 'population_change_2019_2020']])
```

```
    Country Name  population_change_2019_2020
119        Japan                    -372000.0
```

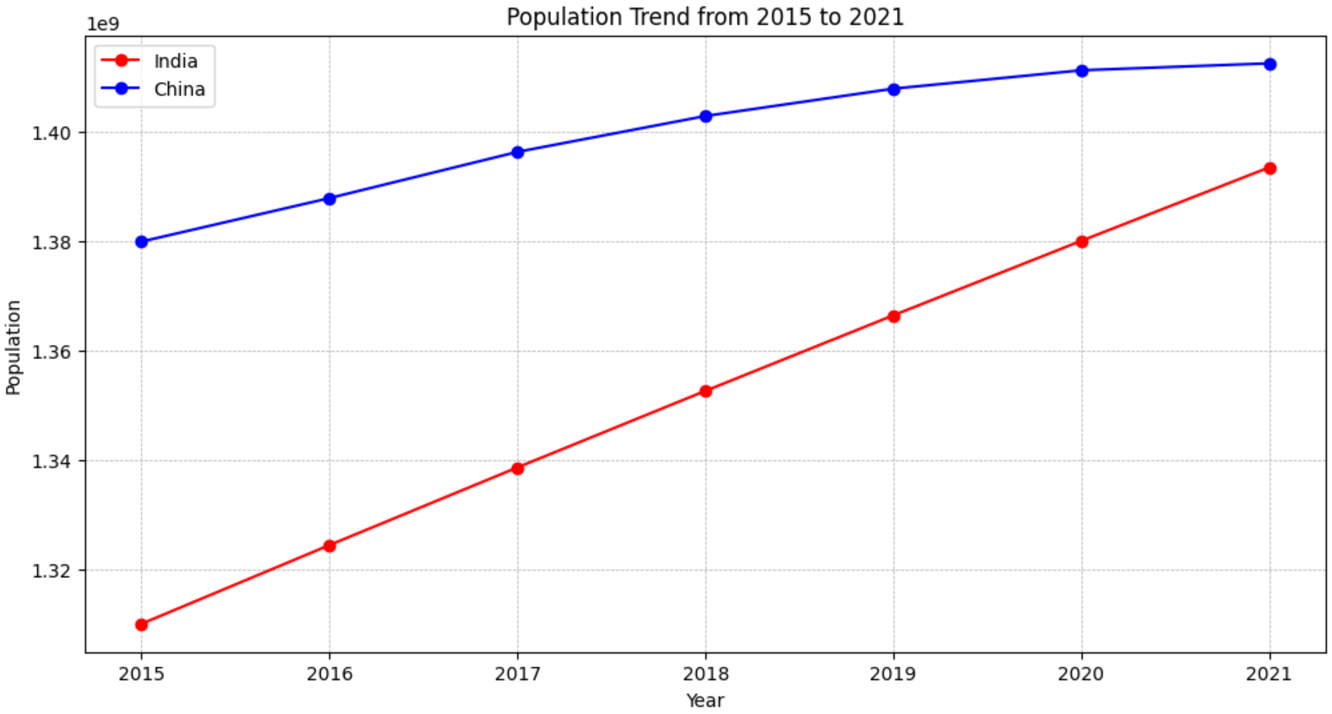
Population of India vs China from 2015 to 2021

```
country_data_1 = d[d['Country Name'] == 'India'].iloc[0]
country_data_2 = d[d['Country Name'] == 'China'].iloc[0]

years = ['2015', '2016', '2017', '2018', '2019', '2020', '2021']
population_values_1 = country_data_1[years]
population_values_2 = country_data_2[years]

#Plot
bar_width = 0.35
index = range(len(years))

plt.figure(figsize=(12, 7))
bars1 = plt.bar(index, population_values_1, bar_width, label='India', color='blue', edgecolor='black')
bars2 = plt.bar([i+bar_width for i in index], population_values_2, bar_width, label='China', color='red', edgecolor='black')

plt.xlabel('Year')
plt.ylabel('Population')
plt.title('Population of India vs China from 2015 to 2021')
plt.xticks([i+bar_width/2 for i in index], years)
plt.legend()
plt.tight_layout()
plt.show()
```



Population Trend from 2015 to 2021

```
plt.figure(figsize=(12, 6))
plt.plot(years, population_values_1, marker='o', label='India', color='red')
plt.plot(years, population_values_2, marker='o', label='China', color='blue')
plt.xlabel('Year')
plt.ylabel('Population')
plt.title('Population Trend from 2015 to 2021')
plt.legend()
plt.grid(True, which='both', linestyle='--', linewidth=0.5)
plt.show()
```


Population Trend from 2015 to 2021
1e9
India
China
1.40
1.38
1.36
1.34
1.32
Population
2015
2016
2017
2018
2019
2020
2021
Year