Understanding the Dataset

The World Development Indicators (WDI) dataset is a comprehensive and widely used resource for analyzing global development trends. It is curated by the World Bank and provides a vast collection of data on over 1,400 indicators that cover a wide range of economic, social, environmental, and governance topics. Features In the Dataset:

- Country Name
- Country Code
- Series Name
- Series Code
- 2020 [YR2020]
- 2021 [YR2021]
- 2022 [YR2022]
- 2023 [YR2023]

Load The Dataset

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

df=pd.read_csv('C:\\Users\\KAMESH\\OneDrive\\Desktop\\project.csv')
```

Preview the Data

```
df.head()
```

```
  Country Name Country Code  \
0        India          IND
1        India          IND
2        India          IND
3        India          IND
4        India          IND

                                          Series Name        Series
Code  \
0  Access to clean fuels and technologies for coo...
EG.CFT.ACCS.ZS
1            Access to electricity (% of population)
EG.ELC.ACCS.ZS
2  Access to electricity, urban (% of urban popul...
EG.ELC.ACCS.UR.ZS
3  Women who were first married by age 15 (% of w...
SP.M15.2024.FE.ZS
4  Women who were first married by age 18 (% of w...
SP.M18.2024.FE.ZS
```

```
  2020 [YR2020] 2021 [YR2021] 2022 [YR2022] 2023 [YR2023]
0          66.8          70.5          74.5            ..
1          96.5          99.6          99.2            ..
2            99           100           100            ..
3            ..             4            ..            ..
4            ..          22.3            ..            ..
```

```
df.tail()
```

```
                                          Country Name Country Code  \
3135                                               NaN          NaN
3136                                               NaN          NaN
3137                                               NaN          NaN
3138  Data from database: World Development Indicators          NaN
3139                           Last Updated: 06/28/2024          NaN

     Series Name Series Code 2020 [YR2020] 2021 [YR2021] 2022 [YR2022]
\
3135         NaN         NaN           NaN           NaN           NaN

3136         NaN         NaN           NaN           NaN           NaN

3137         NaN         NaN           NaN           NaN           NaN

3138         NaN         NaN           NaN           NaN           NaN

3139         NaN         NaN           NaN           NaN           NaN


     2023 [YR2023]
3135           NaN
3136           NaN
3137           NaN
3138           NaN
3139           NaN
```

```
### Check Datatypes and Missing Values
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3140 entries, 0 to 3139
Data columns (total 8 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Country Name   3137 non-null   object
 1   Country Code   3135 non-null   object
 2   Series Name    3135 non-null   object
 3   Series Code    3135 non-null   object
 4   2020 [YR2020]  3135 non-null   object
 5   2021 [YR2021]  3135 non-null   object
```

```
 6   2022 [YR2022]  3135 non-null   object
 7   2023 [YR2023]  3135 non-null   object
dtypes: object(8)
memory usage: 196.4+ KB
```

Insights or Observations:

- All Features in the dataset are object type

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

```
Country Name     3
Country Code     5
Series Name      5
Series Code      5
2020 [YR2020]    5
2021 [YR2021]    5
2022 [YR2022]    5
2023 [YR2023]    5
dtype: int64
```

Insights or Observations

- There are some Missing Values

Data Checks

```
### dropping Null Values

df_copy=df.copy()

df_copy=df_copy.drop([3135,3136,3137,3138,3139])

df_copy
```

```
     Country Name Country Code  \
0           India          IND
1           India          IND
2           India          IND
3           India          IND
4           India          IND
...           ...          ...
3130      Ukraine          UKR
3131      Ukraine          UKR
3132      Ukraine          UKR
3133      Ukraine          UKR
3134      Ukraine          UKR

                                              Series Name
Series Code  \
0     Access to clean fuels and technologies for coo...
EG.CFT.ACCS.ZS
```

```
1               Access to electricity (% of population)
EG.ELC.ACCS.ZS
2     Access to electricity, urban (% of urban popul...
EG.ELC.ACCS.UR.ZS
3     Women who were first married by age 15 (% of w...
SP.M15.2024.FE.ZS
4     Women who were first married by age 18 (% of w...
SP.M18.2024.FE.ZS
...                                                 ...
...
3130  Average transaction cost of sending remittance...
SI.RMT.COST.IB.ZS
3131  Average time to clear exports through customs ...
IC.CUS.DURS.EX
3132  Average working hours of children, working onl...
SL.TLF.0714.WK.MA.TM
3133                   Bank capital to assets ratio (%)
FB.BNK.CAPA.ZS
3134               Birth rate, crude (per 1,000 people)
SP.DYN.CBRT.IN

     2020 [YR2020] 2021 [YR2021] 2022 [YR2022] 2023 [YR2023]
0             66.8          70.5          74.5            ..
1             96.5          99.6          99.2            ..
2               99           100           100            ..
3               ..             4            ..            ..
4               ..          22.3            ..            ..
...            ...           ...           ...           ...
3130      3.099224       2.84237     1.7741665        0.7175
3131            ..            ..            ..            ..
3132            ..            ..            ..            ..
3133   7.186556348   6.887845496    5.82142597            ..
3134           7.8           7.3         7.659            ..

[3135 rows x 8 columns]
```

```
df_copy.isnull().sum()
```

```
Country Name     0
Country Code     0
Series Name      0
Series Code      0
2020 [YR2020]    0
2021 [YR2021]    0
2022 [YR2022]    0
2023 [YR2023]    0
dtype: int64
```

Insights or Observations:

- All Null Values from the Dataset has been removed

```
### Checking for Duplicate Values
df_copy.duplicated().sum()
```

```
0
```

Insights or Observations:

- There is No Duplicate Values

```
### Datatype Check
df_copy.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3135 entries, 0 to 3134
Data columns (total 8 columns):
 #   Column         Non-Null Count  Dtype 
---  ------         --------------  ----- 
 0   Country Name   3135 non-null   object
 1   Country Code   3135 non-null   object
 2   Series Name    3135 non-null   object
 3   Series Code    3135 non-null   object
 4   2020 [YR2020]  3135 non-null   object
 5   2021 [YR2021]  3135 non-null   object
 6   2022 [YR2022]  3135 non-null   object
 7   2023 [YR2023]  3135 non-null   object
dtypes: object(8)
memory usage: 196.1+ KB
```

Insights or Observations:

- Here, All Data items are of Object Type.

```
df_copy.nunique()
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[1], line 1
----> 1 df_copy.nunique()

NameError: name 'df_copy' is not defined
```

Insights or Observations:

- The Number Of Unique Values of Each Column is Displayed Above.

```
df_copy.describe()
```

```
       Country Name Country Code  \
count          3135         3135
unique           33           33
top           India          IND
freq             95           95

                                                 Series Name     Series
Code  \
count                                                  3135
3135
unique                                                   95
95
top     Access to clean fuels and technologies for coo...
EG.CFT.ACCS.ZS
freq                                                     33
33

       2020 [YR2020] 2021 [YR2021] 2022 [YR2022] 2023 [YR2023]
count           3135          3135          3135          3135
unique          1478          1332          1082           527
top               ..            ..            ..            ..
freq            1379          1536          1822          2578
```

Insights or Observations:

- Statistics of the Dataset is Shown Above

Data Cleaning

- Removing Null and Proper Datatype Conversion

```
df_copy=df_copy.replace('..',np.nan)
```

```
---------------------------------------------------------------------------
-----
NameError                                 Traceback (most recent call
last)
Cell In[3], line 1
----> 1 df_copy=df_copy.replace('..',np.nan)

NameError: name 'df_copy' is not defined
```

```
df_copy['2020 [YR2020]']=df_copy['2020 [YR2020]'].astype(float)

df_copy['2021 [YR2021]']=df_copy['2021 [YR2021]'].astype(float)
df_copy['2022 [YR2022]']=df_copy['2022 [YR2022]'].astype(float)
df_copy['2023 [YR2023]']=df_copy['2023 [YR2023]'].astype(float)

df_copy.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3135 entries, 0 to 3134
Data columns (total 8 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Country Name   3135 non-null   object
 1   Country Code   3135 non-null   object
 2   Series Name    3135 non-null   object
 3   Series Code    3135 non-null   object
 4   2020 [YR2020]  1756 non-null   float64
 5   2021 [YR2021]  1599 non-null   float64
 6   2022 [YR2022]  1313 non-null   float64
 7   2023 [YR2023]  557 non-null    float64
dtypes: float64(4), object(4)
memory usage: 196.1+ KB
```

```
df_copy.isnull().sum()
```

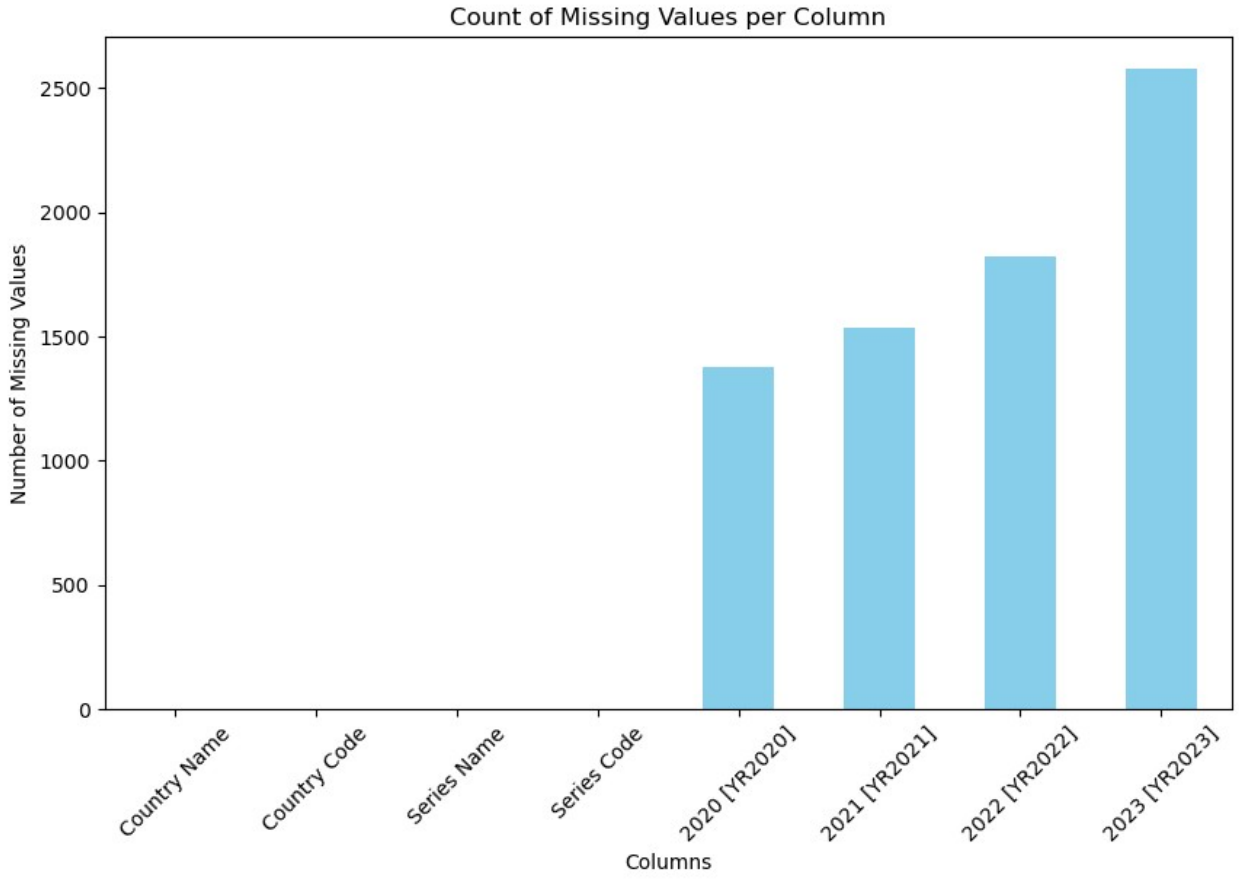
```
Country Name        0
Country Code        0
Series Name         0
Series Code         0
2020 [YR2020]    1379
2021 [YR2021]    1536
2022 [YR2022]    1822
2023 [YR2023]    2578
dtype: int64
```

Insights or Observations:

- The Conversion of Datatypes and Removal of Error Values is Done Succeesfully

- Handling Missing Values

```
import pandas as pd
import matplotlib.pyplot as plt
missing_values = df_copy.isnull().sum()
plt.figure(figsize=(10, 6))
missing_values.plot(kind='bar', color='skyblue')
plt.title('Count of Missing Values per Column')
plt.ylabel('Number of Missing Values')
plt.xlabel('Columns')
plt.xticks(rotation=45)
plt.show()
```




```
#Implementation of Mean Imputation To Remove Null Values
df_copy['2020 [YR2020]']=df_copy['2020 [YR2020]'].fillna(df_copy['2020
[YR2020]'].mean())
df_copy['2021 [YR2021]']=df_copy['2021 [YR2021]'].fillna(df_copy['2021
[YR2021]'].mean())
df_copy['2022 [YR2022]']=df_copy['2022 [YR2022]'].fillna(df_copy['2022
[YR2022]'].mean())
df_copy['2023 [YR2023]']=df_copy['2023 [YR2023]'].fillna(df_copy['2023
[YR2023]'].mean())

df_copy['2020 [YR2020]']=df_copy['2020 [YR2020]'].astype(str)
df_copy['2020 [YR2020]_new']=df_copy['2020
[YR2020]'].str.split('.').str[0]
df_copy['2020 [YR2020]_new']=df_copy['2020
[YR2020]_new'].astype('int64')

df_copy['2021 [YR2021]']=df_copy['2021 [YR2021]'].astype(str)
df_copy['2021 [YR2021]_new']=df_copy['2021
[YR2021]'].str.split('.').str[0]
df_copy['2021 [YR2021]_new']=df_copy['2021
[YR2021]_new'].astype('int64')
```

```
df_copy['2022 [YR2022]']=df_copy['2022 [YR2022]'].astype(str)
df_copy['2022 [YR2022]_new']=df_copy['2022
[YR2022]'].str.split('.').str[0]
df_copy['2022 [YR2022]_new']=df_copy['2022
[YR2022]_new'].astype('int64')

df_copy['2023 [YR2023]']=df_copy['2023 [YR2023]'].astype(str)
df_copy['2023 [YR2023]_new']=df_copy['2023
[YR2023]'].str.split('.').str[0]
df_copy['2023 [YR2023]_new']=df_copy['2023
[YR2023]_new'].astype('int64')

df_copy.drop(labels=['2020 [YR2020]','2021 [YR2021]','2022
[YR2022]','2023 [YR2023]'],axis=1,inplace=True)

df_copy
```

```
     Country Name Country Code  \
0           India          IND
1           India          IND
2           India          IND
3           India          IND
4           India          IND
...           ...          ...
3130      Ukraine          UKR
3131      Ukraine          UKR
3132      Ukraine          UKR
3133      Ukraine          UKR
3134      Ukraine          UKR

                                                Series Name
Series Code  \
0     Access to clean fuels and technologies for coo...
EG.CFT.ACCS.ZS
1                Access to electricity (% of population)
EG.ELC.ACCS.ZS
2     Access to electricity, urban (% of urban popul...
EG.ELC.ACCS.UR.ZS
3     Women who were first married by age 15 (% of w...
SP.M15.2024.FE.ZS
4     Women who were first married by age 18 (% of w...
SP.M18.2024.FE.ZS
...                                                 ...
...
3130  Average transaction cost of sending remittance...
SI.RMT.COST.IB.ZS
3131  Average time to clear exports through customs ...
IC.CUS.DURS.EX
3132  Average working hours of children, working onl...
SL.TLF.0714.WK.MA.TM
```

```
3133                    Bank capital to assets ratio (%)
FB.BNK.CAPA.ZS
3134                Birth rate, crude (per 1,000 people)
SP.DYN.CBRT.IN

      2020 [YR2020]_new  2021 [YR2021]_new  2022 [YR2022]_new  \
0                    66                  70                  74
1                    96                  99                  99
2                    99                 100                 100
3            1564447047                   4          2281880837
4            1564447047                  22          2281880837
...                 ...                 ...                 ...
3130                 3                   2                   1
3131         1564447047          1802084812          2281880837
3132         1564447047          1802084812          2281880837
3133                 7                   6                   5
3134                 7                   7                   7

      2023 [YR2023]_new
0            5474418263
1            5474418263
2            5474418263
3            5474418263
4            5474418263
...                 ...
3130                 0
3131         5474418263
3132         5474418263
3133         5474418263
3134         5474418263

[3135 rows x 8 columns]
```

Insights or Observations:

- The Mssing Values Has been SuccessFully Handled

- Covariance and Variance

```
df_copy[['2020 [YR2020]_new','2021 [YR2021]_new','2022
[YR2022]_new','2023 [YR2023]_new']].cov()

                    2020 [YR2020]_new  2021 [YR2021]_new  2022
[YR2022]_new  \
2020 [YR2020]_new       5.576877e+20       5.918822e+20
6.164809e+20
2021 [YR2021]_new       5.918822e+20       6.285536e+20
6.546308e+20
2022 [YR2022]_new       6.164809e+20       6.546308e+20
6.822893e+20
2023 [YR2023]_new       6.331403e+20       6.724942e+20
```

```
7.008457e+20
```

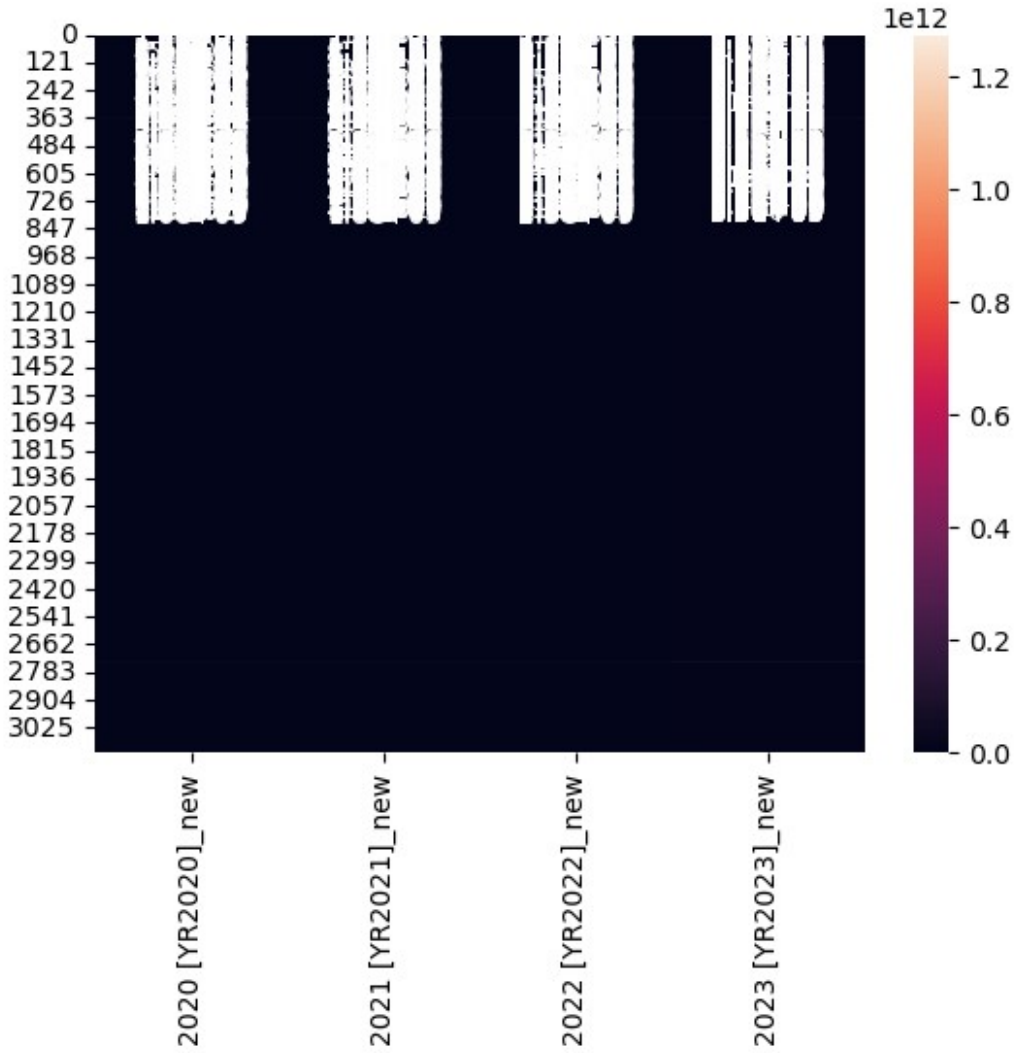
```
                    2023 [YR2023]_new
2020 [YR2020]_new       6.331403e+20
2021 [YR2021]_new       6.724942e+20
2022 [YR2022]_new       7.008457e+20
2023 [YR2023]_new       7.232213e+20
```

```
df_copy[['2020 [YR2020]_new','2021 [YR2021]_new','2022
[YR2022]_new','2023 [YR2023]_new']].var()
```

```
2020 [YR2020]_new    5.576877e+20
2021 [YR2021]_new    6.285536e+20
2022 [YR2022]_new    6.822893e+20
2023 [YR2023]_new    7.232213e+20
dtype: float64
```

```
df_num=df_copy[['2020 [YR2020]_new','2021 [YR2021]_new','2022
[YR2022]_new','2023 [YR2023]_new']]

sns.heatmap(data=df_num,annot=True)
```

```
<Axes: >
```



- Skewness And Kurtosis

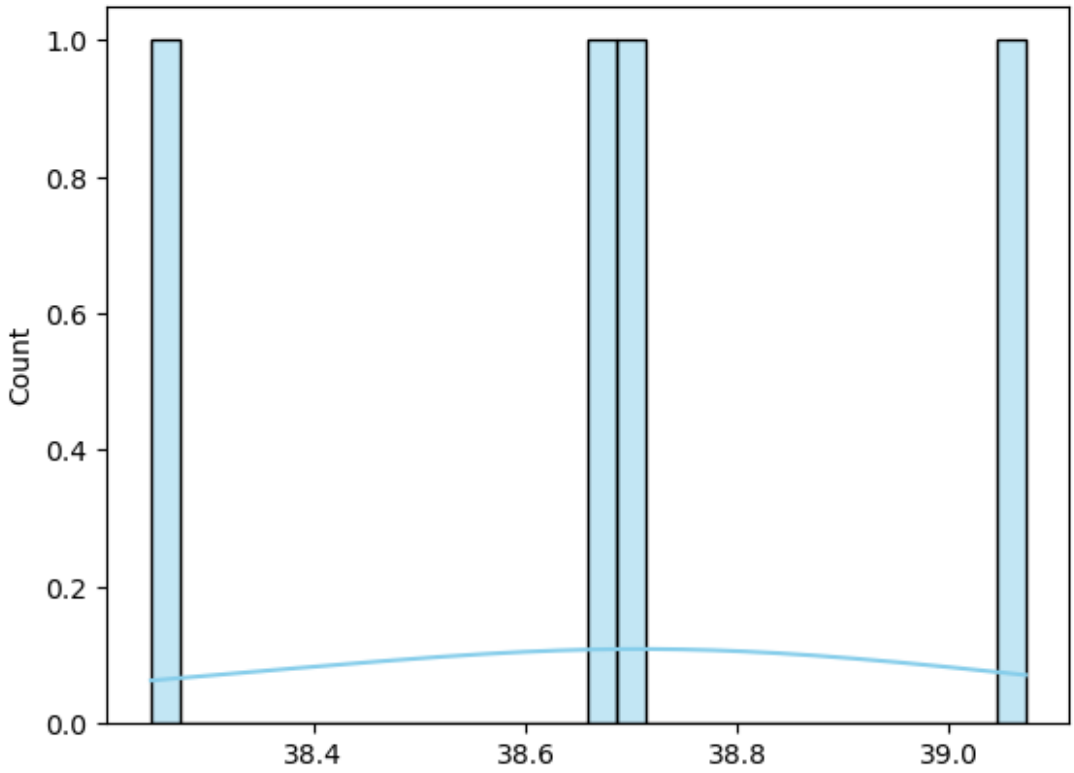
```
from scipy.stats import skew, kurtosis

skewness = df_num.apply(skew)
kurtosis_values = df_num.apply(kurtosis)

skewness
```

```
2020 [YR2020]_new    38.245963
2021 [YR2021]_new    38.714252
2022 [YR2022]_new    38.659750
2023 [YR2023]_new    39.073143
dtype: float64
```

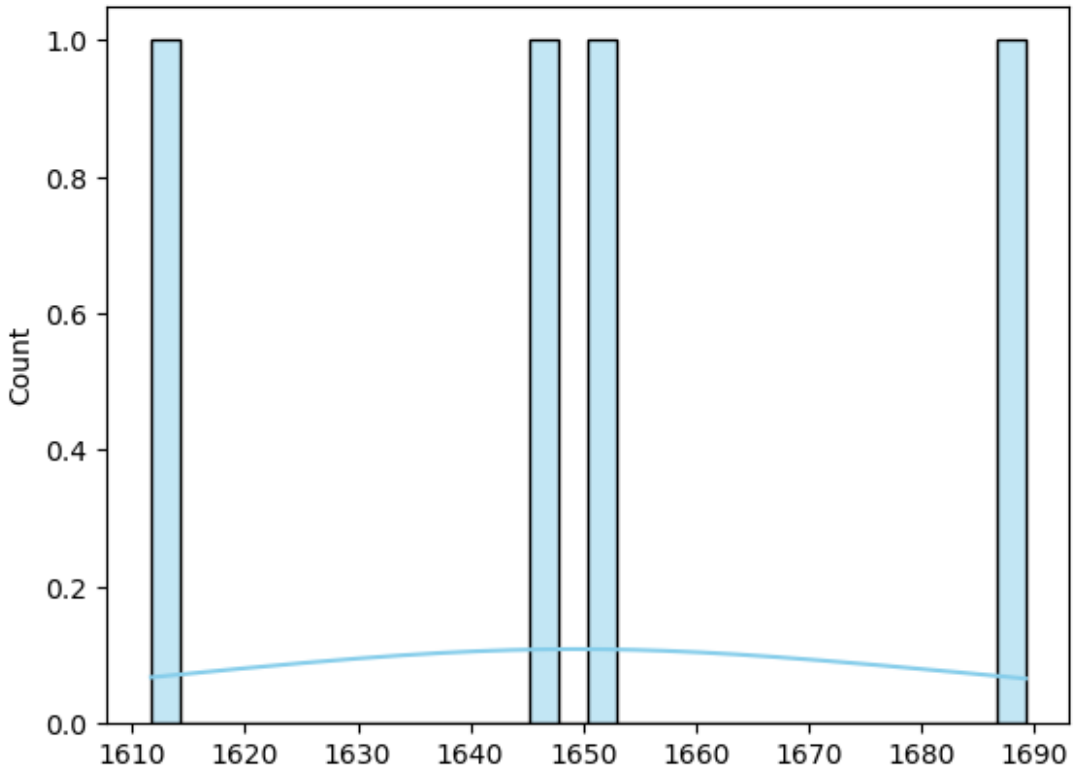
```
sns.histplot(data=skewness, kde=True, bins=30, color='skyblue')
```

```
C:\ProgramData\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
```

```
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):

<Axes: ylabel='Count'>
```




```
kurtosis_values

2020 [YR2020]_new    1611.646167
2021 [YR2021]_new    1652.035262
2022 [YR2022]_new    1647.669006
2023 [YR2023]_new    1689.264255
dtype: float64

sns.histplot(data=kurtosis_values, kde=True, bins=30, color='skyblue')

C:\ProgramData\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):

<Axes: ylabel='Count'>
```



- Visualizations And Insights From the Dataset

```
sorted_df = df_copy.sort_values(by='Series
Name').reset_index(drop=True)

sorted_df
```

```
              Country Name Country Code  \
0                    India          IND
1           United Kingdom          GBR
2            United States          USA
3              Afghanistan          AFG
4     United Arab Emirates          ARE
...                    ...          ...
3130    Russian Federation          RUS
3131                Ireland          IRL
3132                  China          CHN
3133                Finland          FIN
3134                  Sudan          SDN

                                             Series Name     Series
Code  \
0     Access to clean fuels and technologies for coo...
EG.CFT.ACCS.ZS
1     Access to clean fuels and technologies for coo...
EG.CFT.ACCS.ZS
```

```
2     Access to clean fuels and technologies for coo...
EG.CFT.ACCS.ZS
3     Access to clean fuels and technologies for coo...
EG.CFT.ACCS.ZS
4     Access to clean fuels and technologies for coo...
EG.CFT.ACCS.ZS
...                                                 ...              ..
.
3130  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3131  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3132  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3133  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3134  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG

      2020 [YR2020]_new  2021 [YR2021]_new  2022 [YR2022]_new  \
0                    66                 70                 74
1                   100                100                100
2                   100                100                100
3                    33                 34                 36
4                   100                100                100
...                 ...                ...                ...
3130         1564447047         1802084812         2281880837
3131         1564447047         1802084812         2281880837
3132         1564447047         1802084812         2281880837
3133         1564447047         1802084812         2281880837
3134               1000               1000               1000

      2023 [YR2023]_new
0            5474418263
1            5474418263
2            5474418263
3            5474418263
4            5474418263
...                 ...
3130         5474418263
3131         5474418263
3132         5474418263
3133         5474418263
3134         5474418263

[3135 rows x 8 columns]
```

```
df1=sorted_df[sorted_df['Series Name']=='Young people (ages 15-24)
newly infected with HIV']
```

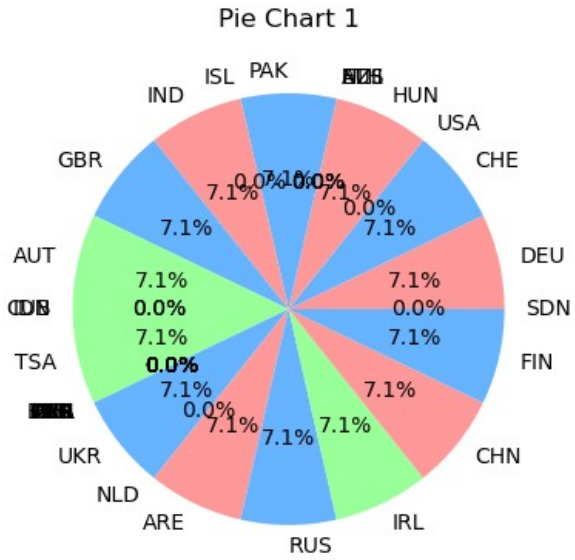
```
df1

                  Country Name Country Code  \
3102                   Germany          DEU
3103               Switzerland          CHE
3104             United States          USA
3105                   Hungary          HUN
3106                  Ethiopia          ETH
3107                 Australia          AUS
3108               New Zealand          NZL
3109                  Pakistan          PAK
3110                   Iceland          ISL
3111                     India          IND
3112            United Kingdom          GBR
3113                   Austria          AUT
3114                 Indonesia          IDN
3115                      Cuba          CUB
3116  South Asia (IDA & IBRD)          TSA
3117               Afghanistan          AFG
3118                    Canada          CAN
3119                  Thailand          THA
3120                   Belgium          BEL
3121                  Malaysia          MYS
3122                   Albania          ALB
3123                Bangladesh          BGD
3124                Azerbaijan          AZE
3125                   Algeria          DZA
3126                    France          FRA
3127                   Ukraine          UKR
3128               Netherlands          NLD
3129     United Arab Emirates          ARE
3130       Russian Federation          RUS
3131                   Ireland          IRL
3132                     China          CHN
3133                   Finland          FIN
3134                     Sudan          SDN

                                                 Series Name     Series
Code  \
3102  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3103  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3104  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3105  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3106  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3107  Young people (ages 15-24) newly infected with HIV
```

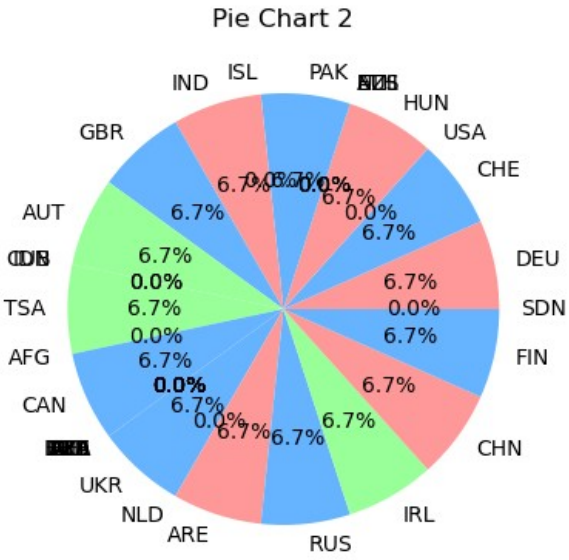
```
SH.HIV.INCD.YG
3108  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3109  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3110  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3111  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3112  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3113  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3114  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3115  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3116  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3117  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3118  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3119  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3120  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3121  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3122  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3123  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3124  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3125  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3126  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3127  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3128  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3129  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3130  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3131  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
```

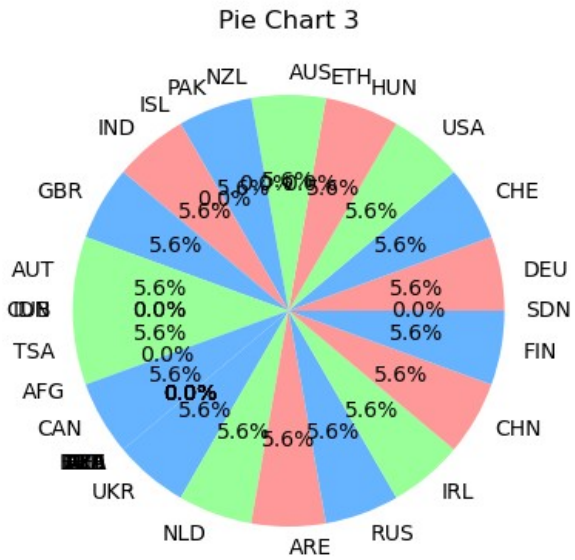
```
3132  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3133  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG
3134  Young people (ages 15-24) newly infected with HIV
SH.HIV.INCD.YG

      2020 [YR2020]_new  2021 [YR2021]_new  2022 [YR2022]_new  \
3102         1564447047         1802084812         2281880837
3103         1564447047         1802084812         2281880837
3104               5900               5600         2281880837
3105         1564447047         1802084812         2281880837
3106               2900               2700               2400
3107                100                100         2281880837
3108                100                100                100
3109         1564447047         1802084812         2281880837
3110                100                100                100
3111         1564447047         1802084812         2281880837
3112         1564447047         1802084812         2281880837
3113         1564447047         1802084812         2281880837
3114              13000              13000              12000
3115               1000               1000               1000
3116         1564447047         1802084812         2281880837
3117                500                500                500
3118                200         1802084812         2281880837
3119               4700               4500               4400
3120                100                100                100
3121               1000               1000               1000
3122                100                100                100
3123                500                500                500
3124                100                100                100
3125                500                500                500
3126               1000               1000               1000
3127         1564447047         1802084812         2281880837
3128                100                100         2281880837
3129         1564447047         1802084812         2281880837
3130         1564447047         1802084812         2281880837
3131         1564447047         1802084812         2281880837
3132         1564447047         1802084812         2281880837
3133         1564447047         1802084812         2281880837
3134               1000               1000               1000

      2023 [YR2023]_new
3102         5474418263
3103         5474418263
3104         5474418263
3105         5474418263
3106         5474418263
3107         5474418263
```

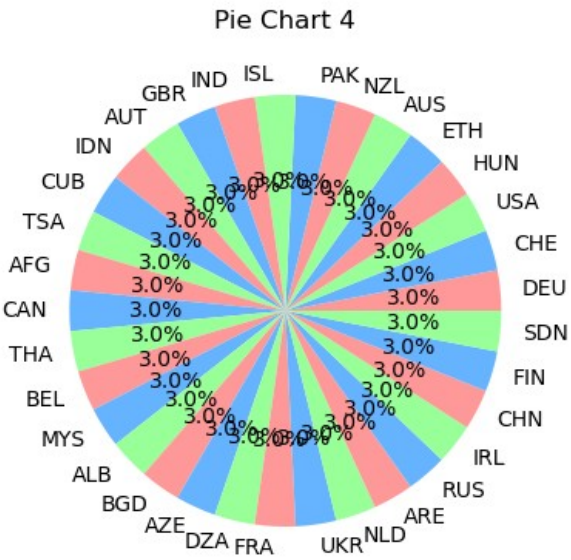
```
3108          5474418263
3109          5474418263
3110          5474418263
3111          5474418263
3112          5474418263
3113          5474418263
3114          5474418263
3115          5474418263
3116          5474418263
3117          5474418263
3118          5474418263
3119          5474418263
3120          5474418263
3121          5474418263
3122          5474418263
3123          5474418263
3124          5474418263
3125          5474418263
3126          5474418263
3127          5474418263
3128          5474418263
3129          5474418263
3130          5474418263
3131          5474418263
3132          5474418263
3133          5474418263
3134          5474418263
```

```
fig, axs = plt.subplots(2, 2, figsize=(10, 10))
axs[0, 0].pie(df1['2020 [YR2020]_new'], labels=df1['Country Code'],
autopct='%1.1f%%', colors=['#ff9999','#66b3ff','#99ff99'])
axs[0, 0].set_title('Pie Chart 1')

# Subplot 2
axs[0, 1].pie(df1['2021 [YR2021]_new'], labels=df1['Country Code'],
autopct='%1.1f%%', colors=['#ff9999','#66b3ff','#99ff99'])
axs[0, 1].set_title('Pie Chart 2')

# Subplot 3
axs[1, 0].pie(df1['2022 [YR2022]_new'], labels=df1['Country Code'],
autopct='%1.1f%%', colors=['#ff9999','#66b3ff','#99ff99'])
axs[1, 0].set_title('Pie Chart 3')

# Subplot 4
axs[1, 1].pie(df1['2023 [YR2023]_new'], labels=df1['Country Code'],
autopct='%1.1f%%', colors=['#ff9999','#66b3ff','#99ff99'])
axs[1, 1].set_title('Pie Chart 4')
plt.show()
```









Insights Or Observations:

- In 2020, The HIV for Young People Started to spread at high concern But ,In 2023 The Spread of HIV Among Young People is equal among the globe.

```
sorted_df['Series Name'].unique()

array(['Access to clean fuels and technologies for cooking (% of
population)',
       'Access to clean fuels and technologies for cooking, rural (%
of rural population)',
       'Access to clean fuels and technologies for cooking, urban (%
of urban population)',
```

```
       'Access to electricity (% of population)',
       'Access to electricity, rural (% of rural population)',
       'Access to electricity, urban (% of urban population)',
       'Adolescents out of school (% of lower secondary school age)',
       'Age dependency ratio (% of working-age population)',
       'Age dependency ratio, old (% of working-age population)',
       'Age dependency ratio, young (% of working-age population)',
       'Agricultural irrigated land (% of total agricultural land)',
       'Agricultural land (% of land area)', 'Agricultural land (sq.
km)',
       'Agricultural machinery, tractors',
       'Agricultural machinery, tractors per 100 sq. km of arable
land',
       'Agricultural methane emissions (% of total)',
       'Agricultural methane emissions (thousand metric tons of CO2
equivalent)',
       'Agricultural nitrous oxide emissions (% of total)',
       'Agriculture, forestry, and fishing, value added (% of GDP)',
       'Agriculture, forestry, and fishing, value added (constant 2015
US$)',
       'Air transport, passengers carried',
       'Alternative and nuclear energy (% of total energy use)',
       'Annual freshwater withdrawals, agriculture (% of total
freshwater withdrawal)',
       'Armed forces personnel (% of total labor force)',
       'Automated teller machines (ATMs) (per 100,000 adults)',
       'Average time to clear exports through customs (days)',
       'Average transaction cost of sending remittances to a specific
country (%)',
       'Average working hours of children, study and work, ages 7-14
(hours per week)',
       'Average working hours of children, study and work, female,
ages 7-14 (hours per week)',
       'Average working hours of children, study and work, male, ages
7-14 (hours per week)',
       'Average working hours of children, working only, ages 7-14
(hours per week)',
       'Average working hours of children, working only, female, ages
7-14 (hours per week)',
       'Average working hours of children, working only, male, ages 7-
14 (hours per week)',
       'Bank capital to assets ratio (%)',
       'Birth rate, crude (per 1,000 people)',
       'Time required to build a warehouse (days)',
       'Time required to enforce a contract (days)',
       'Total debt service (% of GNI)',
       'Total fisheries production (metric tons)',
       'Total greenhouse gas emissions (% change from 1990)',
       'Total greenhouse gas emissions (kt of CO2 equivalent)',
```

```
       'Total natural resources rents (% of GDP)',
       'Trained teachers in lower secondary education (% of total
teachers)',
       'Trained teachers in lower secondary education, female (% of
female teachers)',
       'Trained teachers in lower secondary education, male (% of male
teachers)',
       'Trained teachers in preprimary education (% of total
teachers)',
       'Trained teachers in preprimary education, female (% of female
teachers)',
       'Trained teachers in preprimary education, male (% of male
teachers)',
       'Trained teachers in primary education (% of total teachers)',
       'Trained teachers in primary education, female (% of female
teachers)',
       'Trained teachers in primary education, male (% of male
teachers)',
       'Trained teachers in secondary education (% of total
teachers)',
       'Trained teachers in secondary education, female (% of female
teachers)',
       'Trained teachers in secondary education, male (% of male
teachers)',
       'Trained teachers in upper secondary education (% of total
teachers)',
       'Trained teachers in upper secondary education, female (% of
female teachers)',
       'Trained teachers in upper secondary education, male (% of male
teachers)',
       'Transport services (% of commercial service imports)',
       'Transport services (% of service exports, BoP)',
       'Tuberculosis case detection rate (%, all forms)',
       'Tuberculosis treatment success rate (% of new cases)',
       'Unemployment, female (% of female labor force) (national
estimate)',
       'Unemployment, male (% of male labor force) (modeled ILO
estimate)',
       'Unemployment, total (% of total labor force) (national
estimate)',
       'Unemployment, youth female (% of female labor force ages 15-
24) (modeled ILO estimate)',
       'Unemployment, youth female (% of female labor force ages 15-
24) (national estimate)',
       'Unemployment, youth male (% of male labor force ages 15-24)
(modeled ILO estimate)',
       'Unemployment, youth male (% of male labor force ages 15-24)
(national estimate)',
       'Unemployment, youth total (% of total labor force ages 15-24)
```

```
(modeled ILO estimate)',
       'Unemployment, youth total (% of total labor force ages 15-24)
(national estimate)',
       'Urban land area (sq. km)',
       'Urban land area where elevation is below 5 meters (% of total
land area)',
       'Urban land area where elevation is below 5 meters (sq. km)',
       'Urban population (% of total population)',
       'Urban population living in areas where elevation is below 5
meters (% of total population)',
       'Use of insecticide-treated bed nets (% of under-5
population)',
       'Value lost due to electrical outages (% of sales for affected
firms)',
       'Voice and Accountability: Estimate',
       'Voice and Accountability: Number of Sources',
       'Voice and Accountability: Percentile Rank',
       'Vulnerable employment, male (% of male employment) (modeled
ILO estimate)',
       'Vulnerable employment, total (% of total employment) (modeled
ILO estimate)',
       'Wage and salaried workers, female (% of female employment)
(modeled ILO estimate)',
       'Wage and salaried workers, male (% of male employment)
(modeled ILO estimate)',
       'Wage and salaried workers, total (% of total employment)
(modeled ILO estimate)',
       'Wanted fertility rate (births per woman)',
       'Water productivity, total (constant 2015 US$ GDP per cubic
meter of total freshwater withdrawal)',
       'Wholesale price index (2010 = 100)',
       'Women Business and the Law Index Score (scale 1-100)',
       'Women who believe a husband is justified in beating his wife
when she goes out without telling him (%)',
       'Women who believe a husband is justified in beating his wife
when she neglects the children (%)',
       'Women who were first married by age 15 (% of women ages 20-
24)',
       'Women who were first married by age 18 (% of women ages 20-
24)',
       "Women's share of population ages 15+ living with HIV (%)",
       'Young people (ages 15-24) newly infected with HIV'],
dtype=object)
```

```
df2=sorted_df[sorted_df['Series Name']=='Access to clean fuels and
technologies for cooking (% of population)']
```

```
df2
```

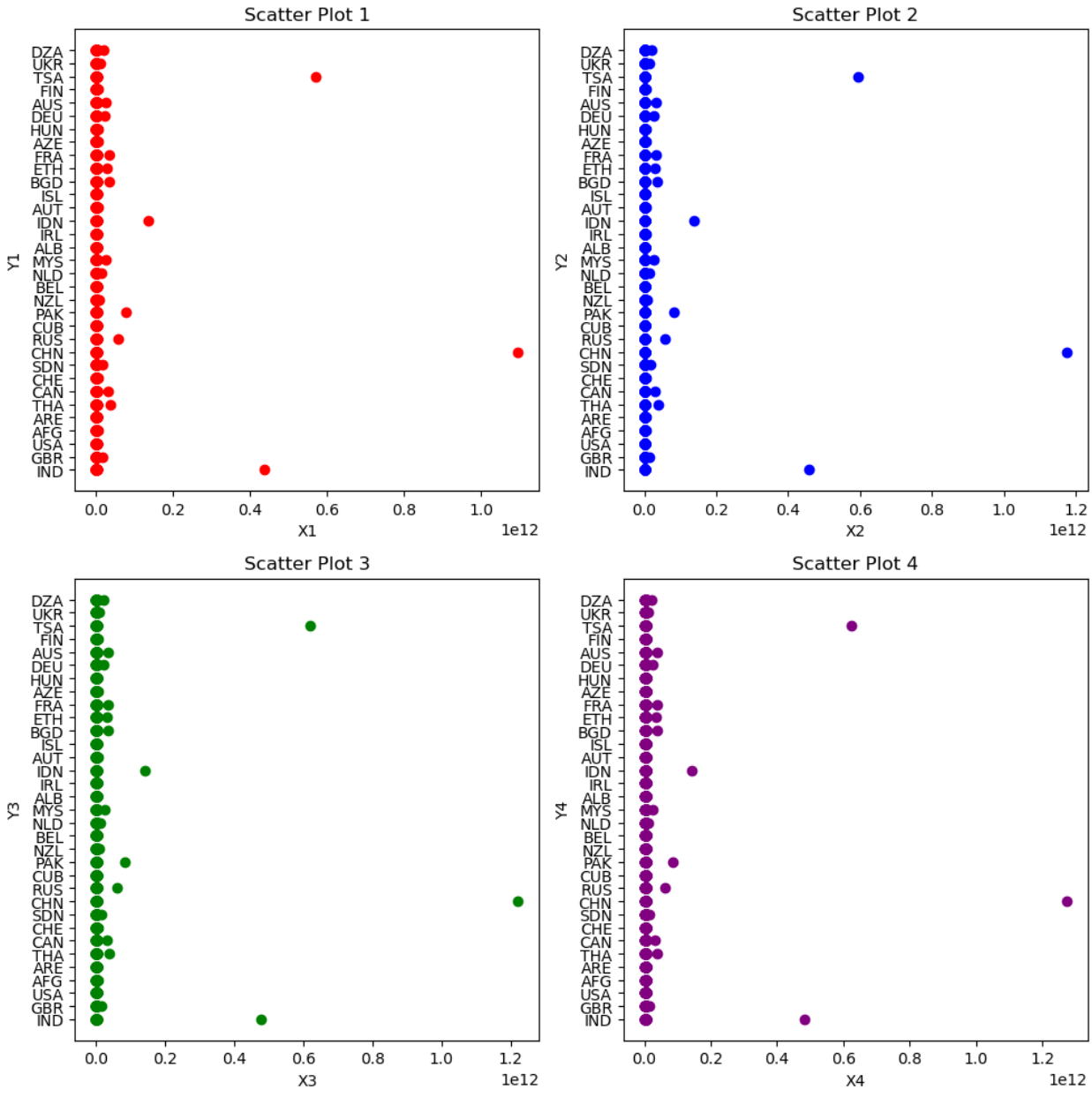
```
                Country Name Country Code  \
0                      India          IND
1             United Kingdom          GBR
2              United States          USA
3                Afghanistan          AFG
4      United Arab Emirates          ARE
5                   Thailand          THA
6                     Canada          CAN
7                Switzerland          CHE
8                      Sudan          SDN
9                      China          CHN
10       Russian Federation          RUS
11                      Cuba          CUB
12                  Pakistan          PAK
13               New Zealand          NZL
14                   Belgium          BEL
15               Netherlands          NLD
16                  Malaysia          MYS
17                   Albania          ALB
18                   Ireland          IRL
19                 Indonesia          IDN
20                   Austria          AUT
21                   Iceland          ISL
22                Bangladesh          BGD
23                  Ethiopia          ETH
24                    France          FRA
25                 Azerbaijan          AZE
26                   Hungary          HUN
27                   Germany          DEU
28                 Australia          AUS
29                   Finland          FIN
30  South Asia (IDA & IBRD)          TSA
31                   Ukraine          UKR
32                   Algeria          DZA

                                                Series Name     Series Code
\
0   Access to clean fuels and technologies for coo...  EG.CFT.ACCS.ZS

1   Access to clean fuels and technologies for coo...  EG.CFT.ACCS.ZS

2   Access to clean fuels and technologies for coo...  EG.CFT.ACCS.ZS

3   Access to clean fuels and technologies for coo...  EG.CFT.ACCS.ZS

4   Access to clean fuels and technologies for coo...  EG.CFT.ACCS.ZS

5   Access to clean fuels and technologies for coo...  EG.CFT.ACCS.ZS

6   Access to clean fuels and technologies for coo...  EG.CFT.ACCS.ZS
```

7	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
8	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
9	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
10	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
11	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
12	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
13	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
14	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
15	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
16	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
17	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
18	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
19	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
20	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
21	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
22	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
23	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
24	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
25	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
26	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
27	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
28	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
29	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
30	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
31	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS
32	Access to clean fuels and technologies for coo...	EG.CFT.ACCS.ZS

	2020 [YR2020]_new	2021 [YR2021]_new	2022 [YR2022]_new	2023 [YR2023]_new
0	66	70	74	5474418263
1	100	100	100	5474418263
2	100	100	100	5474418263
3	33	34	36	5474418263
4	100	100	100	5474418263
5	84	85	86	5474418263
6	100	100	100	5474418263
7	100	100	100	5474418263
8	61	62	65	5474418263
9	83	85	87	5474418263
10	99	99	99	5474418263
11	94	95	94	5474418263
12	49	51	52	5474418263
13	100	100	100	5474418263
14	100	100	100	5474418263
15	100	100	100	5474418263
16	86	86	84	5474418263
17	83	83	84	5474418263
18	100	100	100	5474418263
19	84	87	89	5474418263
20	100	100	100	5474418263
21	100	100	100	5474418263
22	24	26	28	

```
5474418263
23                   7                   8                   8
5474418263
24                 100                 100                 100
5474418263
25                  98                  98                  98
5474418263
26                 100                 100                 100
5474418263
27                 100                 100                 100
5474418263
28                 100                 100                 100
5474418263
29                 100                 100                 100
5474418263
30                  59                  62                  65
5474418263
31                  95                  94                  94
5474418263
32                  99                  99                  99
5474418263

fig, axs = plt.subplots(2, 2, figsize=(10, 10))

# Subplot 1
axs[0, 0].scatter(sorted_df['2020 [YR2020]_new'],sorted_df['Country
Code'], color='red')
axs[0, 0].set_title('Scatter Plot 1')
axs[0, 0].set_xlabel('X1')
axs[0, 0].set_ylabel('Y1')

# Subplot 2
axs[0, 1].scatter(sorted_df['2021 [YR2021]_new'],sorted_df['Country
Code'],color='blue')
axs[0, 1].set_title('Scatter Plot 2')
axs[0, 1].set_xlabel('X2')
axs[0, 1].set_ylabel('Y2')

# Subplot 3
axs[1, 0].scatter(sorted_df['2022 [YR2022]_new'],sorted_df['Country
Code'], color='green')
axs[1, 0].set_title('Scatter Plot 3')
axs[1, 0].set_xlabel('X3')
axs[1, 0].set_ylabel('Y3')

# Subplot 4
axs[1, 1].scatter(sorted_df['2023 [YR2023]_new'],sorted_df['Country
Code'],color='purple')
axs[1, 1].set_title('Scatter Plot 4')
axs[1, 1].set_xlabel('X4')
```

```
axs[1, 1].set_ylabel('Y4')

# Adjust layout so titles and labels don't overlap
plt.tight_layout()

# Display the plots
plt.show()
```



Insights or Observations:

- China is Consistent in Giving Access to clean fuels and technologies for cooking (% of population) While Many Developed Countries are Continuously Unable to Achive Growth in Providing Clean Fuels to its population

```
df3=sorted_df[sorted_df['Series Name']=='Unemployment, total (% of
total labor force) (national estimate)']

df3
```

```
                  Country Name Country Code  \
2079                     India          IND
2080                   Finland          FIN
2081       Russian Federation           RUS
2082                Bangladesh          BGD
2083                   Albania          ALB
2084               Afghanistan          AFG
2085                     Sudan          SDN
2086                   Ireland          IRL
2087                 Azerbaijan          AZE
2088                   Thailand          THA
2089                       Cuba          CUB
2090                     France          FRA
2091                Netherlands          NLD
2092             United States          USA
2093           United Kingdom           GBR
2094                  Pakistan          PAK
2095                   Austria          AUT
2096                    Canada          CAN
2097                New Zealand          NZL
2098                  Australia          AUS
2099                Switzerland          CHE
2100                      China          CHN
2101                   Ukraine          UKR
2102                  Malaysia          MYS
2103                  Ethiopia          ETH
2104                   Hungary          HUN
2105                   Germany          DEU
2106     United Arab Emirates           ARE
2107                   Algeria          DZA
2108  South Asia (IDA & IBRD)           TSA
2109                   Belgium          BEL
2110                 Indonesia          IDN
2111                   Iceland          ISL

                                                 Series Name         Series
Code  \
2079  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2080  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2081  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2082  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
```

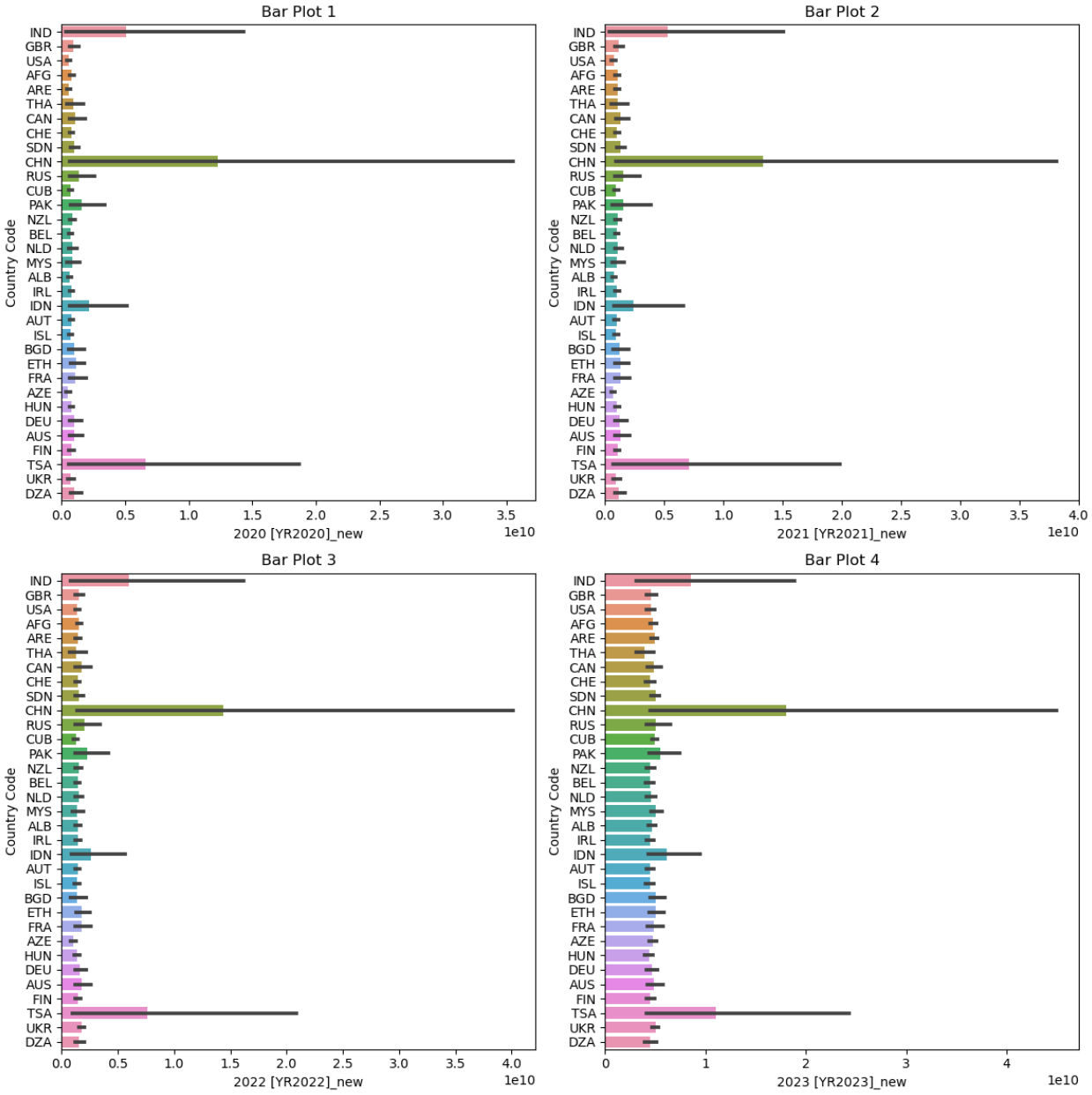
```
2083  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2084  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2085  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2086  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2087  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2088  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2089  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2090  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2091  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2092  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2093  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2094  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2095  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2096  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2097  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2098  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2099  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2100  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2101  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2102  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2103  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2104  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2105  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2106  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2107  Unemployment, total (% of total labor force) (...
```

```
SL.UEM.TOTL.NE.ZS
2108  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2109  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2110  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS
2111  Unemployment, total (% of total labor force) (...
SL.UEM.TOTL.NE.ZS

      2020 [YR2020]_new  2021 [YR2021]_new  2022 [YR2022]_new  \
2079                  7                  6                  4
2080                  7                  7                  6
2081                  5                  4                  3
2082         1564447047         1802084812                  5
2083         1564447047         1802084812         2281880837
2084                 11                  5         2281880837
2085         1564447047         1802084812                  7
2086                  5                  6                  4
2087                  7                  6                  5
2088                  1                  1                  0
2089         1564447047         1802084812         2281880837
2090                  8                  7                  7
2091                  3                  4                  3
2092                  8                  5                  3
2093                  4                  4                  3
2094         1564447047                  6         2281880837
2095                  5                  6                  4
2096                  9                  7                  5
2097                  4                  3                  3
2098                  6                  5                  3
2099                  4                  5                  4
2100                  5                  5         2281880837
2101                  9                  9         2281880837
2102                  4                  4                  3
2103         1564447047                  3         2281880837
2104                  4                  4                  3
2105                  3                  3                  3
2106                  4                  3                  2
2107         1564447047         1802084812         2281880837
2108                  7                  6                  4
2109                  5                  6                  5
2110                  4                  3                  3
2111                  5                  6                  3

      2023 [YR2023]_new
2079                  4
2080                  7
2081                  3
```

```
2082         5474418263
2083         5474418263
2084         5474418263
2085         5474418263
2086                  4
2087         5474418263
2088                  0
2089         5474418263
2090                  7
2091                  3
2092                  3
2093                  3
2094         5474418263
2095                  5
2096                  5
2097                  3
2098                  3
2099                  4
2100         5474418263
2101         5474418263
2102         5474418263
2103         5474418263
2104                  4
2105                  3
2106         5474418263
2107         5474418263
2108                  4
2109                  5
2110         5474418263
2111                  3
```

```
fig, axs = plt.subplots(2, 2, figsize=(12, 12))

# Subplot 1
sns.barplot(x=sorted_df['2020 [YR2020]_new'], y=sorted_df['Country
Code'], data=df3, ax=axs[0, 0])
axs[0, 0].set_title('Bar Plot 1')

# Subplot 2
sns.barplot(x=sorted_df['2021 [YR2021]_new'], y=sorted_df['Country
Code'], data=df3, ax=axs[0, 1])
axs[0, 1].set_title('Bar Plot 2')

# Subplot 3
sns.barplot(x=sorted_df['2022 [YR2022]_new'], y=sorted_df['Country
Code'], data=df3, ax=axs[1, 0])
axs[1, 0].set_title('Bar Plot 3')

# Subplot 4
sns.barplot(x=sorted_df['2023 [YR2023]_new'], y=sorted_df['Country
```

```
Code'], data=df3, ax=axs[1, 1])
axs[1, 1].set_title('Bar Plot 4')

# Adjust layout so titles and labels don't overlap
plt.tight_layout()

# Display the plots
plt.show()
```



Insights or Observations:

- China has the most Unemployment When Compared to Its total % of total labor force)

```
df4=sorted_df[sorted_df['Series Name']=='Tuberculosis treatment
success rate (% of new cases)']

df4
```

```
                  Country Name Country Code  \
1980                    Canada          CAN
1981                 Indonesia          IDN
1982             United States          USA
1983            United Kingdom          GBR
1984                  Ethiopia          ETH
1985               Netherlands          NLD
1986                   Germany          DEU
1987                  Malaysia          MYS
1988                   Algeria          DZA
1989  South Asia (IDA & IBRD)          TSA
1990                  Thailand          THA
1991                     Sudan          SDN
1992               Switzerland          CHE
1993                 Australia          AUS
1994                Azerbaijan          AZE
1995                     India          IND
1996                    France          FRA
1997                Bangladesh          BGD
1998                   Iceland          ISL
1999     United Arab Emirates          ARE
2000                   Finland          FIN
2001                     China          CHN
2002                   Hungary          HUN
2003                  Pakistan          PAK
2004               New Zealand          NZL
2005               Afghanistan          AFG
2006       Russian Federation          RUS
2007                   Austria          AUT
2008                   Ukraine          UKR
2009                   Albania          ALB
2010                      Cuba          CUB
2011                   Ireland          IRL
2012                   Belgium          BEL

                                            Series Name     Series
Code  \
1980  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1981  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1982  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1983  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
```

```
1984  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1985  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1986  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1987  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1988  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1989  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1990  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1991  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1992  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1993  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1994  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1995  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1996  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1997  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1998  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
1999  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
2000  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
2001  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
2002  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
2003  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
2004  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
2005  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
2006  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
2007  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
2008  Tuberculosis treatment success rate (% of new ...
```

```
SH.TBS.CURE.ZS
2009  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
2010  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
2011  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
2012  Tuberculosis treatment success rate (% of new ...
SH.TBS.CURE.ZS
```

	2020 [YR2020]_new	2021 [YR2021]_new	2022 [YR2022]_new	2023 [YR2023]_new
1980	1564447047	1802084812	2281880837	5474418263
1981	86	87	2281880837	5474418263
1982	74	76	2281880837	5474418263
1983	78	86	2281880837	
1984	86	87	2281880837	
1985	83	82	2281880837	
1986	74	67	2281880837	
1987	78	79	2281880837	
1988	89	90	2281880837	
1989	1564447047	1802084812	2281880837	
1990	83	85	2281880837	
1991	86	1802084812	2281880837	
1992	74	75	2281880837	
1993	90	88	2281880837	
1994	82	82	2281880837	
1995	85	87	2281880837	
1996	37	37	2281880837	
1997	95	97	2281880837	
1998	25	57	2281880837	
1999	77	85	2281880837	
2000	25	7	2281880837	
2001	95	94	2281880837	
2002	66	64	2281880837	
2003	94	94	2281880837	
2004	89	87	2281880837	
2005	95	93	2281880837	
2006	62	60	2281880837	
2007	70	75	2281880837	
2008	77	75	2281880837	
2009	89	90	2281880837	
2010	79	81	2281880837	
2011	6	3	2281880837	
2012	67	78	2281880837	

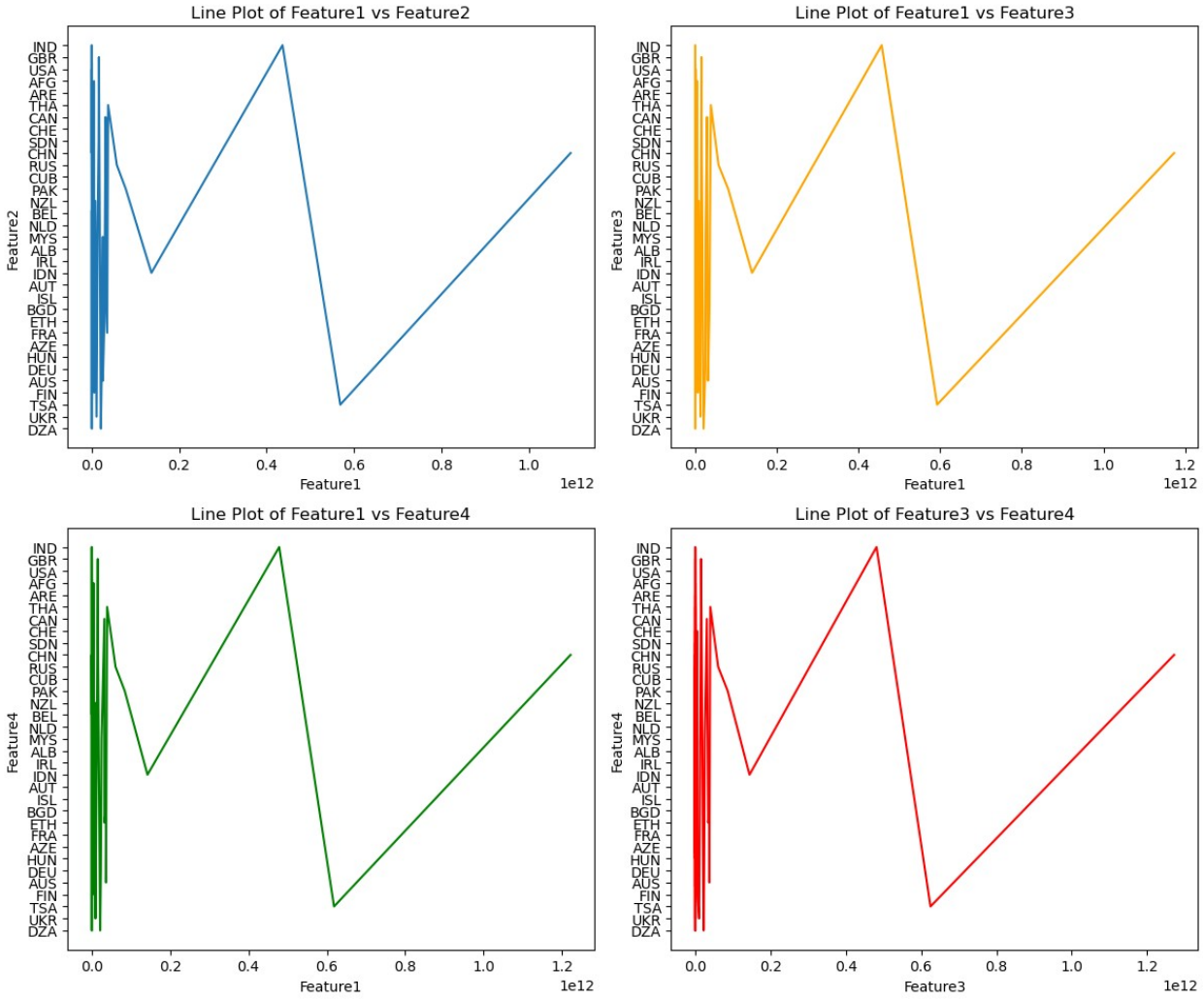
```
1983          5474418263
1984          5474418263
1985          5474418263
1986          5474418263
1987          5474418263
1988          5474418263
1989          5474418263
1990          5474418263
1991          5474418263
1992          5474418263
1993          5474418263
1994          5474418263
1995          5474418263
1996          5474418263
1997          5474418263
1998          5474418263
1999          5474418263
2000          5474418263
2001          5474418263
2002          5474418263
2003          5474418263
2004          5474418263
2005          5474418263
2006          5474418263
2007          5474418263
2008          5474418263
2009          5474418263
2010          5474418263
2011          5474418263
2012          5474418263
```

```
fig, axs = plt.subplots(2, 2, figsize=(12, 10))

# First subplot
sns.lineplot(x=sorted_df['2020 [YR2020]_new'], y=sorted_df['Country
Code'], ax=axs[0, 0])
axs[0, 0].set_title('Line Plot of Feature1 vs Feature2')
axs[0, 0].set_xlabel('Feature1')
axs[0, 0].set_ylabel('Feature2')

# Second subplot
sns.lineplot(x=sorted_df['2021 [YR2021]_new'], y=sorted_df['Country
Code'], ax=axs[0, 1], color='orange')
axs[0, 1].set_title('Line Plot of Feature1 vs Feature3')
axs[0, 1].set_xlabel('Feature1')
axs[0, 1].set_ylabel('Feature3')

# Third subplot
sns.lineplot(x=sorted_df['2022 [YR2022]_new'], y=sorted_df['Country
Code'], ax=axs[1, 0], color='green')
```

```
axs[1, 0].set_title('Line Plot of Feature1 vs Feature4')
axs[1, 0].set_xlabel('Feature1')
axs[1, 0].set_ylabel('Feature4')

# Fourth subplot
sns.lineplot(x=sorted_df['2023 [YR2023]_new'], y=sorted_df['Country Code'], ax=axs[1, 1], color='red')
axs[1, 1].set_title('Line Plot of Feature3 vs Feature4')
axs[1, 1].set_xlabel('Feature3')
axs[1, 1].set_ylabel('Feature4')

# Adjust layout for better spacing
plt.tight_layout()

# Display the plots
plt.show()
```

```
C:\ProgramData\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\ProgramData\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\ProgramData\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\ProgramData\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\ProgramData\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\ProgramData\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\ProgramData\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating
```

```
instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\ProgramData\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
```



Insights or Observations:

- While Many Countries Have Some Poor Rate Over Tuberclosis Treatment, Countries Like Russia, China,Etc haev good Success Rate Over Tuberclosis Treatment.