EDA for Pop dataset

You can find this dataset **Pop** here.

1. Import libraries

In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

2. Import Dataset

In [4]:
```
pop = pd.read_excel("pop.xlsx")
```

3. Show Dataset

In [5]:
```
pop
```

Out[5]:

	Country	country_name	Year	Sex	Sex_m_f	Frmat	Pop1	Pop2	Pop3	Pop4
0	2090	Canada	1991-01-01	1	Male	1	13454600	201600	202100	192700.0
1	2090	Canada	1991-01-01	2	Female	1	13842300	191900	192900	183600.0
2	2090	Canada	1992-01-01	1	Male	2	14091600	204400	810900	NaN
3	2090	Canada	1992-01-01	2	Female	2	14344000	194300	771700	NaN
4	2090	Canada	1993-01-01	1	Male	1	14349600	200900	206700	208200.0
...	...	...	...	...	...	...	...	...	...	...
63	5020	Australia	1995-01-01	2	Female	1	9073400	125100	125600	126400.0
64	5020	Australia	1996-01-01	1	Male	1	9108300	130900	133200	133600.0
65	5020	Australia	1996-01-01	2	Female	1	9203900	124300	126300	126600.0
66	5020	Australia	1997-01-01	1	Male	0	9203171	130061	131587	134236.0
67	5020	Australia	1997-01-01	2	Female	0	9314393	123230	124829	127209.0

68 rows × 32 columns

4. Basic EDA

- **To get first 5 rows of dataset**

In [6]:
```
pop.head()
```

Out[6]:

	Country	country_name	Year	Sex	Sex_m_f	Frmat	Pop1	Pop2	Pop3	Pop4	..
0	2090	Canada	1991-01-01	1	Male	1	13454600	201600	202100	192700.0	..
1	2090	Canada	1991-01-01	2	Female	1	13842300	191900	192900	183600.0	..
2	2090	Canada	1992-01-01	1	Male	2	14091600	204400	810900	NaN	..
3	2090	Canada	1992-01-01	2	Female	2	14344000	194300	771700	NaN	..
4	2090	Canada	1993-01-01	1	Male	1	14349600	200900	206700	208200.0	..

5 rows × 32 columns

- **To get bottom 5 rows of dataset**

In [7]:
```
pop.tail()
```

Out[7]:

	Country	country_name	Year	Sex	Sex_m_f	Frmat	Pop1	Pop2	Pop3	Pop4	..
63	5020	Australia	1995-01-01	2	Female	1	9073400	125100	125600	126400.0	..
64	5020	Australia	1996-01-01	1	Male	1	9108300	130900	133200	133600.0	..
65	5020	Australia	1996-01-01	2	Female	1	9203900	124300	126300	126600.0	..
66	5020	Australia	1997-01-01	1	Male	0	9203171	130061	131587	134236.0	..
67	5020	Australia	1997-01-01	2	Female	0	9314393	123230	124829	127209.0	..

5 rows × 32 columns

In [8]:
```
type(pop)
```

Out[8]:
```
pandas.core.frame.DataFrame
```

- **To show all the column names of dataset**

In [9]:
```
pop.columns
```

Out[9]:
```
Index(['Country', 'country_name', 'Year', 'Sex', 'Sex_m_f', 'Frmat', 'Pop1',
       'Pop2', 'Pop3', 'Pop4', 'Pop5', 'Pop6', 'Pop7', 'Pop8', 'Pop9', 'Pop10',
       'Pop11', 'Pop12', 'Pop13', 'Pop14', 'Pop15', 'Pop16', 'Pop17', 'Pop18',
       'Pop19', 'Pop20', 'Pop21', 'Pop22', 'Pop23', 'Pop24', 'Pop25', 'Lb'],
      dtype='object')
```

- **To fetch any perticular column from dataset**

In [10]:
```
pop['country_name']
```

Out[10]:
```
0        Canada
1        Canada
2        Canada
3        Canada
4        Canada
        ...
63    Australia
64    Australia
65    Australia
66    Australia
67    Australia
Name: country_name, Length: 68, dtype: object
```

In [15]:
```
type(pop['Pop1'])
```

Out[15]:
```
pandas.core.series.Series
```

In [12]:
```
pop.shape
```

Out[12]:
```
(68, 32)
```

In [13]:
```
pop.describe()
```

Out[13]:

	Country	Sex	Frmat	Pop1	Pop2	Pop3	Pop4
count	68.000000	68.000000	68.000000	6.800000e+01	6.800000e+01	6.800000e+01	26.000000
mean	3187.647059	1.500000	1.588235	3.852596e+07	5.553822e+05	2.047987e+06	160440.192308
std	1194.260050	0.503718	0.552749	4.768081e+07	7.189696e+05	2.985305e+06	34882.199114
min	2090.000000	1.000000	0.000000	5.328900e+06	7.400000e+04	1.245000e+05	123200.000000
25%	2150.000000	1.000000	1.000000	8.844950e+06	1.253250e+05	1.911000e+05	127831.750000
50%	2450.000000	1.500000	2.000000	1.467805e+07	1.945500e+05	3.380000e+05	133918.000000
75%	4080.000000	2.000000	2.000000	2.973582e+07	3.693250e+05	1.515900e+06	196075.000000
max	5020.000000	2.000000	2.000000	1.366184e+08	2.052000e+06	8.113700e+06	208200.000000

8 rows × 29 columns

- **To know how many unique values are there in each columns of dataset**

In [14]:
```
pop.nunique()
```

Out[14]:
```
Country          5
country_name     5
Year             7
Sex              2
Sex_m_f          2
Frmat            3
Pop1            68
Pop2            68
```

```
Pop3            68
Pop4            26
Pop5            25
Pop6            26
Pop7            68
Pop8            68
Pop9            67
Pop10           68
Pop11           68
Pop12           68
Pop13           67
Pop14           67
Pop15           68
Pop16           67
Pop17           68
Pop18           67
Pop19           67
Pop20           68
Pop21           68
Pop22           67
Pop23           67
Pop24            2
Pop25            2
Lb              68
dtype: int64
```

In [16]:

```
pop.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 68 entries, 0 to 67
Data columns (total 32 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Country       68 non-null     int64
 1   country_name  68 non-null     object
 2   Year          68 non-null     datetime64[ns]
 3   Sex           68 non-null     int64
 4   Sex_m_f       68 non-null     object
 5   Frmat         68 non-null     int64
 6   Pop1          68 non-null     int64
 7   Pop2          68 non-null     int64
 8   Pop3          68 non-null     int64
 9   Pop4          26 non-null     float64
 10  Pop5          26 non-null     float64
 11  Pop6          26 non-null     float64
 12  Pop7          68 non-null     int64
 13  Pop8          68 non-null     int64
 14  Pop9          68 non-null     int64
 15  Pop10         68 non-null     int64
 16  Pop11         68 non-null     int64
 17  Pop12         68 non-null     int64
 18  Pop13         68 non-null     int64
 19  Pop14         68 non-null     int64
 20  Pop15         68 non-null     int64
 21  Pop16         68 non-null     int64
 22  Pop17         68 non-null     int64
 23  Pop18         68 non-null     int64
 24  Pop19         68 non-null     int64
 25  Pop20         68 non-null     int64
 26  Pop21         68 non-null     int64
 27  Pop22         68 non-null     int64
 28  Pop23         68 non-null     int64
```

```
 29  Pop24         2 non-null      float64
 30  Pop25         2 non-null      float64
 31  Lb            68 non-null     int64
dtypes: datetime64[ns](1), float64(5), int64(24), object(2)
memory usage: 17.1+ KB
```

- **To know about corelation between each column of dataset**

In [17]:
```
pop.corr()
```

Out[17]:

	Country	Sex	Frmat	Pop1	Pop2	Pop3	Pop4	I
Country	1.000000e+00	-1.974456e-17	-3.650567e-01	-0.273437	-0.283163	-0.281064	-0.986630	-0.98
Sex	-1.974456e-17	1.000000e+00	1.190284e-17	0.017073	-0.018408	-0.016233	-0.121696	-0.11
Frmat	-3.650567e-01	1.190284e-17	1.000000e+00	0.430549	0.421165	0.478763	0.250806	0.24
Pop1	-2.734370e-01	1.707314e-02	4.305494e-01	1.000000	0.996123	0.995208	0.981089	0.98
Pop2	-2.831634e-01	-1.840814e-02	4.211648e-01	0.996123	1.000000	0.996673	0.986831	0.97
Pop3	-2.810638e-01	-1.623333e-02	4.787630e-01	0.995208	0.996673	1.000000	0.994653	0.98
Pop4	-9.866305e-01	-1.216957e-01	2.508056e-01	0.981089	0.986831	0.994653	1.000000	0.99
Pop5	-9.823326e-01	-1.164822e-01	2.469789e-01	0.983134	0.978926	0.988025	0.997585	1.00
Pop6	-9.844367e-01	-1.117692e-01	2.426231e-01	0.986863	0.975922	0.985275	0.994085	0.99
Pop7	-2.871284e-01	-1.871359e-02	4.153900e-01	0.997143	0.998008	0.995977	0.992942	0.99
Pop8	-2.747366e-01	-1.948875e-02	4.255635e-01	0.997981	0.997927	0.996969	0.994398	0.99
Pop9	-2.741789e-01	-2.150685e-02	4.304929e-01	0.997380	0.996636	0.996086	0.988165	0.98
Pop10	-2.734489e-01	-1.497640e-02	4.398473e-01	0.995908	0.999090	0.997035	0.994698	0.99
Pop11	-2.843991e-01	-3.718419e-03	4.356311e-01	0.997075	0.999085	0.996288	0.980319	0.97
Pop12	-2.899222e-01	1.301793e-03	4.203628e-01	0.997804	0.998918	0.995594	0.993076	0.99
Pop13	-2.840737e-01	3.494200e-03	4.142711e-01	0.999190	0.996565	0.994669	0.982148	0.98
Pop14	-2.749587e-01	6.488882e-03	4.237087e-01	0.999189	0.994662	0.993892	0.975484	0.97
Pop15	-2.652368e-01	8.838817e-03	4.273014e-01	0.995474	0.986722	0.988505	0.948523	0.96
Pop16	-2.714988e-01	1.435132e-02	4.232858e-01	0.996406	0.988245	0.988988	0.934477	0.94
Pop17	-2.595141e-01	2.581045e-02	4.416987e-01	0.998564	0.990801	0.991674	0.970241	0.97
Pop18	-2.472167e-01	4.374254e-02	4.521256e-01	0.995244	0.988579	0.988268	0.979009	0.97
Pop19	-2.446311e-01	7.332582e-02	4.461465e-01	0.993064	0.983132	0.983274	0.934153	0.93
Pop20	-2.443101e-01	1.059935e-01	4.395666e-01	0.987782	0.973351	0.974496	0.832089	0.84
Pop21	-2.608568e-01	1.437336e-01	4.169149e-01	0.977163	0.960041	0.959726	0.713423	0.71
Pop22	-2.227403e-01	2.165646e-01	4.329147e-01	0.943606	0.916800	0.920381	0.591049	0.60
Pop23	-1.931020e-01	3.103956e-01	4.210766e-01	0.873571	0.835779	0.841578	0.493846	0.50

	Country	Sex	Frmat	Pop1	Pop2	Pop3	Pop4	I
Pop24	NaN	1.000000e+00	NaN	1.000000	-1.000000	-1.000000	-1.000000	-1.00
Pop25	NaN	1.000000e+00	NaN	1.000000	-1.000000	-1.000000	-1.000000	-1.00
Lb	-2.805946e-01	-1.877639e-02	4.198442e-01	0.996215	0.999963	0.996592	0.975595	0.96

5. To chceck missing, incomplete, or NULL value in each column

In [18]:
```
pop.isnull().sum()
```

Out[18]:
```
Country          0
country_name     0
Year             0
Sex              0
Sex_m_f          0
Frmat            0
Pop1             0
Pop2             0
Pop3             0
Pop4            42
Pop5            42
Pop6            42
Pop7             0
Pop8             0
Pop9             0
Pop10            0
Pop11            0
Pop12            0
Pop13            0
Pop14            0
Pop15            0
Pop16            0
Pop17            0
Pop18            0
Pop19            0
Pop20            0
Pop21            0
Pop22            0
Pop23            0
Pop24           66
Pop25           66
Lb               0
dtype: int64
```

6. To check outliers in the dataset

In [20]:
```
pop.plot(kind="box", subplots=True, layout=(8,4), figsize=(30,30))
```

Out[20]:
```
Country       AxesSubplot(0.125,0.799681;0.168478x0.0803191)
Sex         AxesSubplot(0.327174,0.799681;0.168478x0.0803191)
Frmat       AxesSubplot(0.529348,0.799681;0.168478x0.0803191)
Pop1        AxesSubplot(0.731522,0.799681;0.168478x0.0803191)
Pop2           AxesSubplot(0.125,0.703298;0.168478x0.0803191)
Pop3        AxesSubplot(0.327174,0.703298;0.168478x0.0803191)
```

```
Pop4       AxesSubplot(0.529348,0.703298;0.168478x0.0803191)
Pop5       AxesSubplot(0.731522,0.703298;0.168478x0.0803191)
Pop6          AxesSubplot(0.125,0.606915;0.168478x0.0803191)
Pop7       AxesSubplot(0.327174,0.606915;0.168478x0.0803191)
Pop8       AxesSubplot(0.529348,0.606915;0.168478x0.0803191)
Pop9       AxesSubplot(0.731522,0.606915;0.168478x0.0803191)
Pop10         AxesSubplot(0.125,0.510532;0.168478x0.0803191)
Pop11      AxesSubplot(0.327174,0.510532;0.168478x0.0803191)
Pop12      AxesSubplot(0.529348,0.510532;0.168478x0.0803191)
Pop13      AxesSubplot(0.731522,0.510532;0.168478x0.0803191)
Pop14         AxesSubplot(0.125,0.414149;0.168478x0.0803191)
Pop15      AxesSubplot(0.327174,0.414149;0.168478x0.0803191)
Pop16      AxesSubplot(0.529348,0.414149;0.168478x0.0803191)
Pop17      AxesSubplot(0.731522,0.414149;0.168478x0.0803191)
Pop18         AxesSubplot(0.125,0.317766;0.168478x0.0803191)
Pop19      AxesSubplot(0.327174,0.317766;0.168478x0.0803191)
Pop20      AxesSubplot(0.529348,0.317766;0.168478x0.0803191)
Pop21      AxesSubplot(0.731522,0.317766;0.168478x0.0803191)
Pop22         AxesSubplot(0.125,0.221383;0.168478x0.0803191)
Pop23      AxesSubplot(0.327174,0.221383;0.168478x0.0803191)
Pop24      AxesSubplot(0.529348,0.221383;0.168478x0.0803191)
Pop25      AxesSubplot(0.731522,0.221383;0.168478x0.0803191)
Lb               AxesSubplot(0.125,0.125;0.168478x0.0803191)
dtype: object
```