Practical 1

Aim: data exploration and visualization using mathematical and statistical tools.

Theory:

This practical aim to exploring and visualizing data with python or php or java language using mathematical and statistical tools such as Tableau, Matplotlib and Seaborn and following are the processes that is to be performed on the data:

- Reading CSV file
- Process on it to clean data
- Perform mathematical and statistical techniques – mean, mode, median, summation, groupby and standard deviation using NumPy and Pandas library.
- Visualize the relations and distributions of the data using Plot and Graph techniques.

Practical:

Details of dataset:

The dataset is downloaded from the **UNdata** (https://data.un.org/)

1. Total Fertility Rate (Live births per woman)
2. Average Income per person – Total population, both sexes combined (Income in thousands)

***Step: 1*:** Reading Dataset

First, apply following filters before downloading the **Fertility rate** dataset:

i. Select years from 1950 to 2010.
ii. Select all countries. (Default selected)

Apply following filters before downloading the **Income Data**:

i. Deselect the current filters (High-Income to Upper-middle-income countries).
ii. Select years from 1950 to 2010.
iii. Select all countries. (Default selected)

Here, the **Jupyter Notebook** is used to perform all the exploring, cleaning and visualizing task using **Python** programming language.

Import all the following python libraries:

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

Read the dataset saved in current working directory.

```
df = pd.read_csv("UN_Fertility_Rate_Data.csv")
df.head()
```

	Country or Area	Year(s)	Variant	Value
0	Afghanistan	2005-2010	Medium	6.478
1	Afghanistan	2000-2005	Medium	7.182
2	Afghanistan	1995-2000	Medium	7.654
3	Afghanistan	1990-1995	Medium	7.482
4	Afghanistan	1985-1990	Medium	7.469

***Step: 2*:** Explore the dataset.

```
df.columns
        Index(['Country or Area', 'Year(s)', 'Variant', 'Value'], dtype='object')

df['Variant'].unique()
        array(['Medium'], dtype=object)

df['Year(s)'].unique()
        array(['2005-2010', '2000-2005', '1995-2000', '1990-1995', '1985-1990',
       '1980-1985', '1975-1980', '1970-1975', '1965-1970', '1960-1965',
       '1955-1960', '1950-1955'], dtype=object)

df = df.rename(columns={'Country or Area':'Country','Year(s)':'Years','Value':'Rate'})
df = df.drop(['Variant'], axis = 1)
df2 = df
df
```

	Country	Years	Rate
0	Afghanistan	2005-2010	6.478
1	Afghanistan	2000-2005	7.182
2	Afghanistan	1995-2000	7.654
3	Afghanistan	1990-1995	7.482
4	Afghanistan	1985-1990	7.469

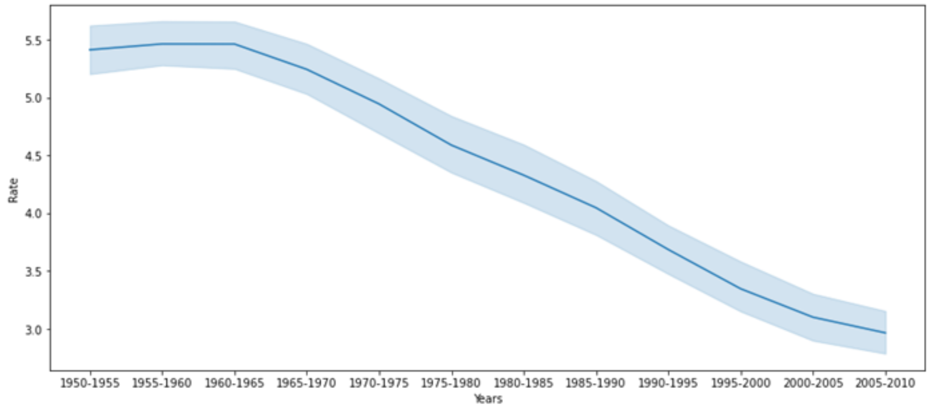
	Country	Years	Rate
...	...	...	...
3463	Zimbabwe	1970-1975	7.400
3464	Zimbabwe	1965-1970	7.400
3465	Zimbabwe	1960-1965	7.300
3466	Zimbabwe	1955-1960	7.000
3467	Zimbabwe	1950-1955	6.800

3468 rows × 3 columns

```
df['Index'] = df.index
df = df.set_index(['Country','Index'])
a = [1,2,3,4,5,6,7,8,9,10,11,12]
a = a*289
df['Index'] = a
```

Step: 3: Plot the graph of Rate vs. Years.

```
df2 = df
plt.figure(figsize=(14,6))
sns.lineplot(y = df['Rate'], x = df['Years'])
<matplotlib.axes._subplots.AxesSubplot at 0x55ddfa0>
```

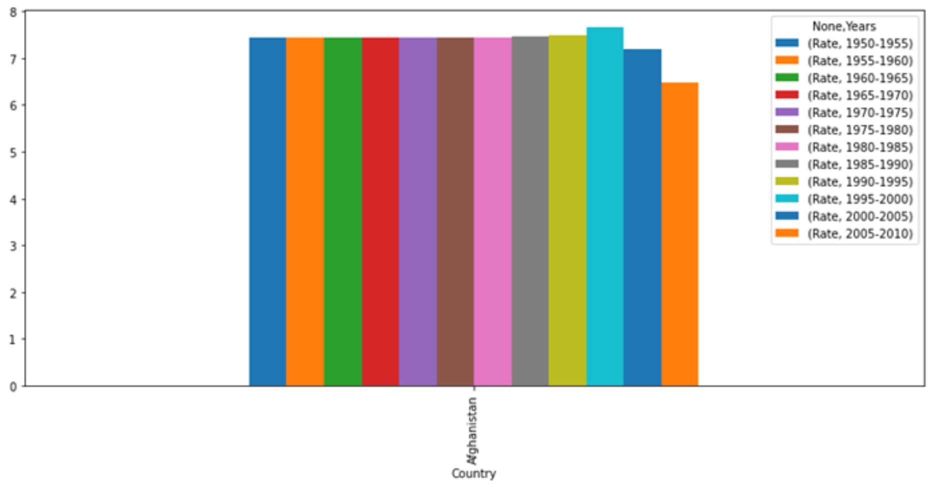


Conclusion: This graph says that the fertility rate per woman is continuously decreasing over the years 1950 to 2010.

Step: 4: Pivot table to make the Country as a Index, Years as column and Rate as a value.

Below is the graph of Fertility rate over the years 1950 to 2010 for the Afghanistan Country. It shows the bar for each 5 year span. It shows that

```
df2.loc[['Afghanistan']].plot(kind ='bar',figsize=(14,6))
<matplotlib.axes._subplots.AxesSubplot at 0x1354b550>
```



Step: 5: Cleaning the Dataset.

```
df2.columns.values
df3 = pd.DataFrame(
    np.arange(24).reshape(2, 12),
    columns=[('Rate', '1950-1955'), ('Rate', '1955-1960'),
       ('Rate', '1960-1965'), ('Rate', '1965-1970'),
       ('Rate', '1970-1975'), ('Rate', '1975-1980'),
```

```
        ('Rate', '1980-1985'), ('Rate', '1985-1990'),
        ('Rate', '1990-1995'), ('Rate', '1995-2000'),
        ('Rate', '2000-2005'), ('Rate', '2005-2010')])
df3.rename(columns='_'.join, inplace=True)
x = df3.columns

df2.columns = x
df2
```

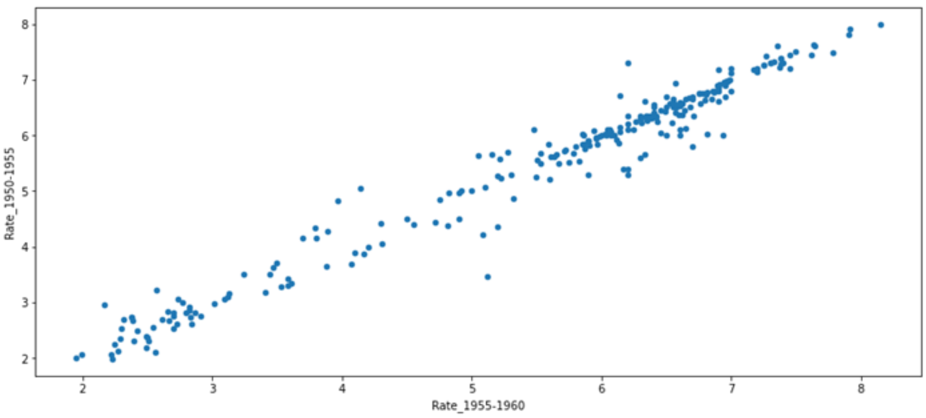
248 rows × 12 columns

Country	Rate_1950-1955	Rate_1955-1960	Rate_1960-1965	Rate_1965-1970	Rate_1970-1975	Rate_1975-1980	Rate_1980-1985	Rate_1985-1990	Rate_1990-1995	Rate_1995-2000	Rate_2000-2005	Rate_2005-2010
Afghanistan	7.450	7.450	7.450	7.450	7.450	7.450	7.450	7.469	7.482	7.654	7.182	6.478
Africa	6.573	6.625	6.699	6.706	6.703	6.640	6.501	6.187	5.724	5.351	5.077	4.900
Albania	6.230	6.545	6.230	5.259	4.600	3.900	3.409	3.150	2.786	2.384	1.946	1.640
Algeria	7.278	7.384	7.648	7.648	7.572	7.175	6.315	5.302	4.120	2.885	2.384	2.724
Angola	6.000	6.500	6.900	7.300	7.500	7.456	7.456	7.400	7.100	6.750	6.550	6.350
...	...	...	...	...	...	...	...	...	...	...	...	...
Western Sahara	6.342	6.424	6.534	6.600	6.573	6.234	5.332	4.600	4.000	3.400	2.850	2.550
World	4.967	4.897	5.018	4.926	4.471	3.881	3.588	3.439	3.005	2.777	2.651	2.584
Yemen	7.800	7.900	8.000	8.250	8.500	8.600	8.800	8.800	8.200	6.800	5.900	5.000
Zambia	6.700	6.950	7.250	7.300	7.400	7.250	6.900	6.600	6.300	6.100	5.950	5.600
Zimbabwe	6.800	7.000	7.300	7.400	7.400	7.300	6.302	5.373	4.415	3.885	3.720	3.885

248 rows × 12 columns

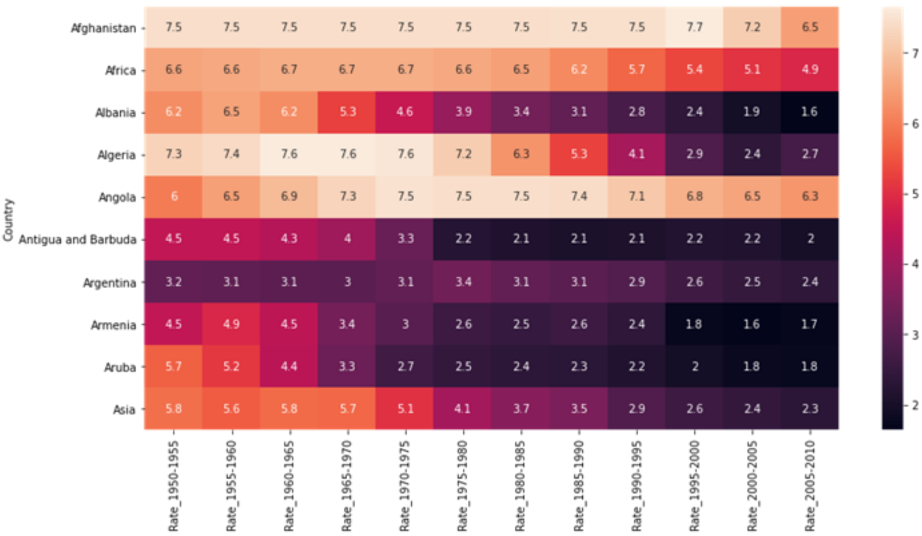
Step: 6: Below is the graph of values of fertility rate for all countries for the year 1955-1960 vs. 1950-1955. It shows that ths scattering of the values a

```
df2.plot(kind ='scatter',figsize=(14,6),x = 'Rate_1955-1960',y='Rate_1950-1955')
<matplotlib.axes._subplots.AxesSubplot at 0x136dfec8>
```



Step: 7: Below heat map is for the fertilty rate values of first 10 countires of the dataframe over the years 1950-1955 to 2005-2010.

```
df3 = df2.iloc[0:10, :-1]
df3
plt.figure(figsize=(14,7))
sns.heatmap(data = df3, annot = True)
<matplotlib.axes._subplots.AxesSubplot at 0x1334b0e8>
```



Step: 8: Here we will create a new and main Data frame of this practical and will name it as 'Data'.
The Data dataframe will look as following:

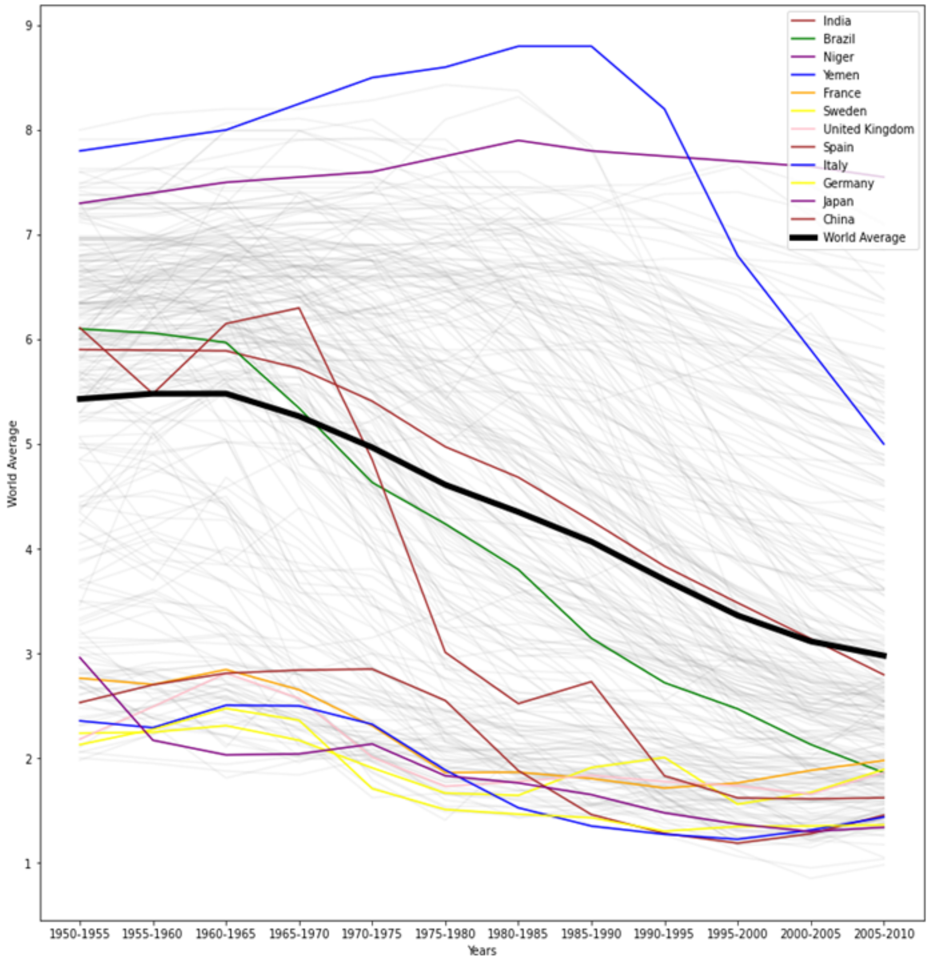
```
X = df.Years.unique()
Data = df
Data = Data.pivot_table(index = 'Years',columns ='Country', values =['Rate'])
df3 = pd.DataFrame(
    np.arange(496).reshape(2, 248),
    columns=['Afghanistan', 'Africa', 'Albania', 'Algeria', 'Angola',
       'Antigua and Barbuda', 'Argentina', 'Armenia', 'Aruba', 'Asia',
       'Australia', 'Australia/New Zealand', 'Austria', 'Azerbaijan',
       . . . .,
       'Western Africa', 'Western Asia', 'Western Europe',
       'Western Sahara', 'World', 'Yemen', 'Zambia', 'Zimbabwe'])
x = df3.columns

Data.columns = x
Data
```

]:

a	Algeria	Angola	Antigua and Barbuda	Argentina	Armenia	Aruba	Asia	...	Venezuela (Bolivarian Republic of)	Viet Nam	Western Africa	Western Asia	Western Europe	Western Sahara	World	Yemen	Zambia	Zimbabwe
0	7.278	6.000	4.50	3.154	4.494	5.650	5.831	...	6.458	5.399	6.433	6.346	2.390	6.342	4.967	7.80	6.70	6.80C
6	7.384	6.500	4.50	3.127	4.900	5.150	5.591	...	6.458	6.164	6.489	6.201	2.489	6.424	4.897	7.90	6.95	7.00C
0	7.648	6.900	4.30	3.090	4.453	4.399	5.797	...	6.180	6.418	6.560	6.188	2.652	6.534	5.018	8.00	7.25	7.30C
9	7.648	7.300	4.00	3.050	3.447	3.301	5.745	...	5.700	6.465	6.627	6.039	2.474	6.600	4.926	8.25	7.30	7.40C
0	7.572	7.500	3.26	3.150	3.037	2.651	5.056	...	4.941	6.329	6.798	5.777	1.962	6.573	4.471	8.50	7.40	7.40C
0	7.175	7.456	2.24	3.400	2.600	2.450	4.097	...	4.468	5.499	6.900	5.360	1.652	6.234	3.861	8.60	7.25	7.30C
9	6.315	7.456	2.14	3.150	2.500	2.358	3.689	...	3.957	4.600	6.860	4.994	1.619	5.332	3.588	8.80	6.90	6.302
0	5.302	7.400	2.07	3.053	2.600	2.300	3.497	...	3.648	3.850	6.661	4.512	1.574	4.600	3.439	8.80	6.60	5.373
6	4.120	7.100	2.09	2.914	2.380	2.174	2.896	...	3.250	3.227	6.395	4.036	1.488	4.000	3.005	8.20	6.30	4.415
4	2.885	6.750	2.20	2.630	1.750	1.953	2.607	...	2.942	2.249	6.153	3.595	1.518	3.400	2.777	6.80	6.10	3.885
6	2.384	6.550	2.16	2.480	1.650	1.816	2.447	...	2.723	1.921	5.950	3.245	1.583	2.850	2.651	5.90	5.95	3.72C
0	2.724	6.350	2.00	2.370	1.720	1.760	2.328	...	2.547	1.928	5.742	3.029	1.639	2.550	2.584	5.00	5.60	3.885

Step: 9: The Next is the most important graph. This line graph is plotted where each line shows each country. The graph is for the values of fertility rate from year 1950 to 2010. Here, all the countries are shown by a light colour gray line. Some of the countries are highlighted. From highlighted Countries, Yemen and Niger are poor countires. India, Brazil and Spain are the average developing countries. All other highlighted can be considered as rich countries.

```
plt.figure(figsize=(14,14))

m = Data.mean(axis = 1)
Data['World Average'] = m
for i in range(0,248):
    sns.lineplot(data = Data, y = Data[Data.columns[i]], x = Data.index, color='grey', alpha = 0.1)
sns.lineplot(data = Data, y = 'India', x = Data.index, color = 'brown', label= 'India')
sns.lineplot(data = Data, y = 'Brazil', x = Data.index, color = 'green',  label= 'Brazil')
sns.lineplot(data = Data, y = 'Niger', x = Data.index, color = 'purple', label= 'Niger')
sns.lineplot(data = Data, y = 'Yemen', x = Data.index, color = 'blue', label= 'Yemen')
sns.lineplot(data = Data, y = 'France', x = Data.index, color = 'orange', label= 'France')
sns.lineplot(data = Data, y = 'Sweden', x = Data.index, color = 'yellow', label= 'Sweden')
sns.lineplot(data = Data, y = 'United Kingdom', x = Data.index, color = 'pink', label= 'United Kingdom')
sns.lineplot(data = Data, y = 'Spain', x = Data.index, color = 'brown', label= 'Spain')
sns.lineplot(data = Data, y = 'Italy', x = Data.index, color = 'blue', label= 'Italy')
sns.lineplot(data = Data, y = 'Germany', x = Data.index, color = 'yellow', label= 'Germany')
sns.lineplot(data = Data, y = 'Japan', x = Data.index, color = 'purple', label= 'Japan')
sns.lineplot(data = Data, y = 'China', x = Data.index, color = 'brown', label= 'China')
sns.lineplot(data = Data, y = 'World Average', x = Data.index, color = 'black', label= 'World Average', linewidth = 5)
savefig('sample.png')
```



Conclusion: From this graph we can say that the poor countries have more fertility rate. i.e. there are more number of children per woman in poor countries. Average or Developing countries have fertility rate between 3 to 6. And Rich countries have verry low fertility rate.

Now, By plotting the world average of each countries over this years, we can say that the line has decreasing curve. That means the fertility rate for almost all countries have been decreasing from 1950 to 2010.

Part: 2:

Here we will read the next dataset i.e. Average income per person.

Step: 1: Read the dataset of Average income per person for all the countries over the years 1950 to 2010.

```
income = pd.read_csv('UN_Income_Data.csv')
income.head()
```

	Country or Area	Year(s)	Variant	Value
0	Afghanistan	2010	Constant fertility	29185.507
1	Afghanistan	2009	Constant fertility	28394.813
2	Afghanistan	2008	Constant fertility	27722.276
3	Afghanistan	2007	Constant fertility	27100.536
4	Afghanistan	2006	Constant fertility	26433.049

Step: 2: Explore the dataset and match the no of countries with the previous dataset.

```
income = income.drop('Variant', axis = 1)
income = income.rename(columns = {'Country or Area' : 'Country', 'Year(s)' : 'Years', 'Value' : 'Income'})
```

```
imean = []
for item in np.split(income, 3477):
    imean.append(item['Income'].mean())
#     print(item['Income'].mean())
imean[1:13]
IncomeCol = imean[0:3384]
# IncomeCol
len(IncomeCol)
        3384

len(imean)
        3477
```

```
country = income['Country'].unique()
len(country)
        282

x = [0]*282
for i in range(0,282):
    x[i] = i
x = np.repeat(x, 12)
x = list(x)
len(x)
        3384
```

Step: 3:

Aim of the Part 2 is that we want to see that does the average income per person in a perticular countires affect the fertility rate of that country or not.

Here we make a hypothesis of the result as follows:

- The graph will be combination of line and scatter.
- The Income of the countries will be represented as line and fertility rate values will be represented as scatter points.
- We want ot see the relationship between this two variables.
- Here we assume that both the graph will coinside on each other.
- It shows that the countries with high income per person has low fertility rate and countries with low income has high frtility rate.

So, we will explore the dataset first:

```
avgincome = pd.DataFrame(zip(IncomeCol, x), columns = ['Income', 'Country'])
years = df['Years'][0:3383]
avgincome['Years'] = years
avgincome = avgincome.pivot_table(columns = 'Country',values = ['Income'], index = 'Years')
avgincome.columns = country
avgincome
```

]:

Years	Afghanistan	Africa	Albania	Algeria	American Samoa	Andorra	Angola	Anguilla	Antigua and Barbuda	Argentina	...	Viet Nam	Wallis and Futuna Islands	
1950-1955	8047.6310	247531.3232	1422.0610	10021.6874	19.7288	11.6584	5380.3326	6.0376	54.9796	21153.0868	...	231660.7624	102.2036	1.81
1955-1960	8689.0116	277007.1242	1638.2960	11342.0114	21.3732	16.4350	5714.9650	6.3634	60.0032	22833.6864	...	242788.9740	104.7250	1.98
1960-1965	9553.0314	312540.3612	1895.3896	12922.7592	25.2148	21.9258	5856.0582	6.7718	63.9734	24662.4154	...	254581.6452	105.0016	2.13
1965-1970	10655.8486	354511.2146	2151.1782	14878.9544	28.5220	28.1816	6762.0806	7.1500	62.4714	26664.7876	...	268525.7760	108.1736	2.24
1970-1975	12095.5466	404069.2672	2412.1072	17105.5406	30.8920	33.8340	8072.8488	7.2596	61.8366	28805.7462	...	284392.3864	108.4852	2.34
1975-1980	13244.6328	463606.4482	2682.0692	19849.7116	35.1346	40.9574	9625.5816	7.3958	61.7628	31183.3482	...	297840.3158	107.1078	2.42
1980-1985	12546.8862	533762.5064	2977.5578	23103.0434	42.7230	50.4726	11461.8728	8.8322	63.6302	33519.9520	...	127772.9340	129791.7558	2.80
1985-1990	11847.9806	613695.8214	3254.2512	26385.8758	49.8086	60.6114	13516.4174	9.9056	70.1950	35652.5300	...	8739.1122	174733.3202	3.03
1990-1995	15757.5100	699691.5748	3130.6212	29234.7394	55.1634	64.5178	15887.8532	11.2366	77.0994	37681.6056	...	10062.5684	189464.5630	3.95
1995-2000	19779.8250	791913.1400	3123.7026	31447.2842	58.9160	73.1366	18796.5088	12.4432	82.7856	39686.0104	...	11646.4674	201749.7550	5.58
2000-2005	23653.9180	894550.7830	3081.3488	33676.0938	58.2418	83.2902	17695.5440	1840.8584	38.1384	8216.6682	...	13556.3236	211401.0542	7.34
2005-2010	27767.2362	802149.4406	93809.1392	15035.6206	7300.4208	19.4446	7.9116	4931.2982	5.7698	50.5266	...	15894.3602	221128.3110	9.22

12 rows × 282 columns

```
avgincome2 = avgincome
plt.figure(figsize = (14,7))
avgincome.mean()
avgincome2 = avgincome2.T
avgincome2['mean'] = avgincome.mean()

Data.mean()
      Afghanistan      7.367917
      Africa           6.140500
      Albania          4.006667
      Algeria          5.702917
      Angola           6.938500
                         ...
      World            3.848667
      Yemen            7.712500
      Zambia           6.691667
      Zimbabwe         5.898333
      World Average    4.401196
      Length: 249, dtype: float64
```

Step: 4: Next we will create a new dataframe combining our two exisisting dataframes 'Data' and 'avgincome'. We will read it from the below file: [UN_Combined_Data.csv]

```
new_df = pd.read_csv("UN_Combined_Data.csv")
reformed = new_df[new_df['Year(s)']=='1990-1995']
reformed1
```

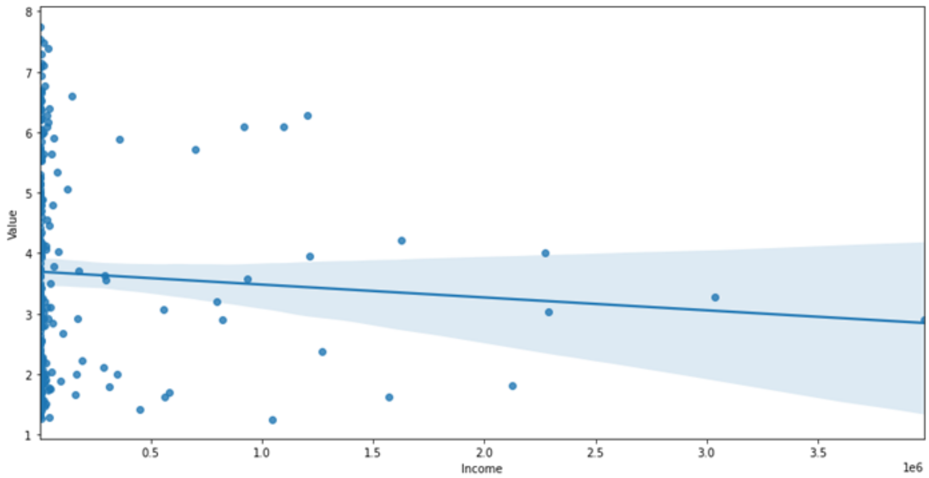
	Country or Area	Year(s)	Value	Income
3	Afghanistan	1990-1995	7.482	15757.5100

	Country or Area	Year(s)	Value	Income
15	Africa	1990-1995	5.724	699691.5748
27	Albania	1990-1995	2.786	3130.6212
39	Algeria	1990-1995	4.120	29234.7394
75	Angola	1990-1995	7.100	15887.8532
...	...	...	...	...
3315	Vanuatu	1990-1995	4.830	2664.3614
3327	Venezuela (Bolivarian Republic of)	1990-1995	3.250	10138.3074
3339	Viet Nam	1990-1995	3.227	76.3872
3363	Western Africa	1990-1995	6.395	41187.3072
3375	Western Asia	1990-1995	4.036	8.8406

245 rows × 4 columns

Step: 5: We will now plot the graph of 'Income' vs. 'Value' (i.e. Fertility Rate).
This is includes regression line. We can derive results from the slope of the regression line.
If the slope of regression line is Positive then the relation between two variables is positive.(i.e. if one value increases, another increases.) and vice-versa.

```
re2 = reformed1.sort_values(['Income'])

plt.figure(figsize = (14,7))
sns.regplot(data = re2, x = 'Income', y = 'Value')
<matplotlib.axes._subplots.AxesSubplot at 0x1485f040>
```



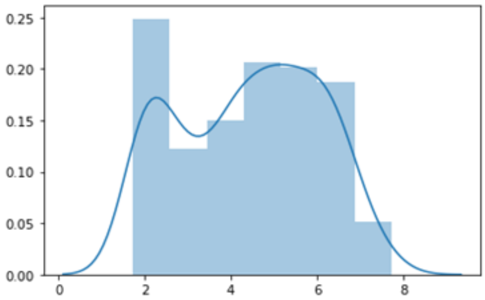
Conclusion of the above graph:
From graph, we can see that the slope of the regression line is negative. And so it proves that Income and Fertility rate are inversely proportional. i.e. the countries whose Income is high tend to have lower fertility rate and countries with low income generally have high fertility rate.

Measures of central Tendancy:

Mean:

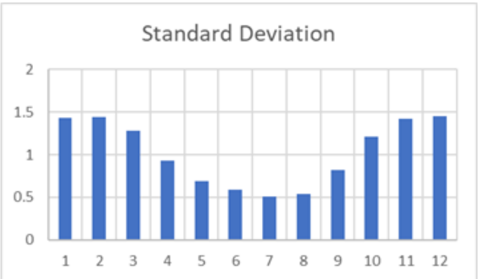
```
def my_mean_fun(xyz):
    my_sum = 0
    for i in range(0, 12):
        my_sum = my_sum + xyz.iloc[i]
#     print(my_sum)
    Mean = my_sum / 12
    print(my_sum / 12)
    return Mean

my_mean = my_mean_fun(Data)
sns.distplot(my_mean)
Afghanistan      7.367917
Africa           6.140500
Albania          4.006667
Algeria          5.702917
Angola           6.938500
                   ...
Yemen            7.712500
```

```
Zambia            6.691667
Zimbabwe          5.898333
World Average     4.401196
mean                   NaN
Length: 250, dtype: float64
<matplotlib.axes._subplots.AxesSubplot at 0x181b93b8>
```



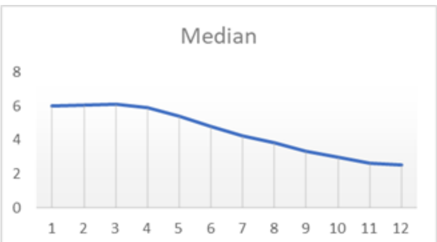
Standard Deviation:

```
import math as math
def my_std_fun(xyz):
    sq_sum = 0
    std = []
    for i in range(0,12):
        diff = my_mean[i] - xyz.iloc[i]
        sq_sum = sq_sum + diff*diff
    var = sq_sum / 12
#     print(var)
    for i in range(0,12):
        std.append(math.sqrt(var[i]))

#     std = math.sqrt(var)
    print(std)
    return std

my_std = my_std_fun(Data)
[3.532038474165607, 2.3819336693552855, 1.193136578140668, 2.1684070906460087, 3.2913903457332605, 1.6880082583353477, 2.07340
38451802714, 1.7757598701174224, 1.7190840639995537, 1.0755802527976859, 2.3017676672716605, 2.2473502031868304]

new_data_std = my_std_fun(new_data)
new_data_df = pd.DataFrame(new_data_std)
new_data_df.to_csv("new_data_std.csv")
```



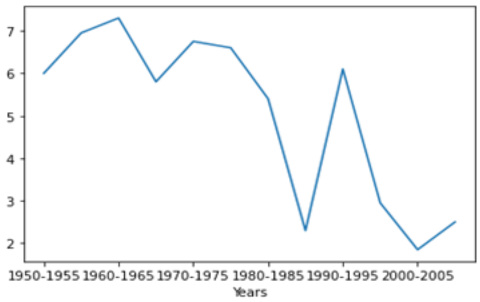
Median:

```
def my_median(xyz):
    Median = []
    for i in range(0,12):
        result = xyz.sort_values(xyz.columns[i], ascending=[1]).iloc[:,i][[125]]
        Median.append(result)
    return Median
my_data_median = my_median(new_data)
my_data_median
median_df = pd.DataFrame(my_data_median)
median_df.to_csv("median_data.csv")
```



Mode:

```
new_data.mode().iloc[0].plot()
<matplotlib.axes._subplots.AxesSubplot at 0x1850ae38>
```



Percentile, Quartile and Decile:

```
def percentile(n):
    p = n * (251) / 100
    pi = int(p)
    result = new_data.sort_values('2000-2005', ascending=[1])
    x = float(result['2000-2005'][[pi]])
#     print(x)
    y = float(result['2000-2005'][[pi+1]])
#     print(y)
    ans = x + (p - pi)*(y - x)
    return ans

p = percentile(10)
print("p10 = D1 = ", p)

p = percentile(25)
print("p25 = Q1 = ", p)

p = percentile(30)
print("p30 = D3 = ", p)

p = percentile(50)
print("p50 = Q2 = ", p)

p = percentile(65)
print("p65 = ", p)

p = percentile(75)
print("p75 = Q3 = ", p)

p = percentile(90)
print("p90 = D9 = ", p)

p = percentile(99)
print("p99 = ", p)

p10 = D1 =  1.371
p25 = Q1 =  1.8415
p30 = D3 =  1.9469
p50 = Q2 =  2.645
p65 =  3.2571500000000007
p75 = Q3 =  4.2145
p90 = D9 =  5.7984
p99 =  nan
```

Skewness:

We have our "Data" dataframe **transposed** to "new_data" dataframe as following.

```
new_data
```

Years	1950-1955	1955-1960	1960-1965	1965-1970	1970-1975	1975-1980	1980-1985	1985-1990	1990-1995	1995-2000	2000-2005	2005-2010
Afghanistan	7.450000	7.450000	7.450000	7.450000	7.45000	7.450000	7.450000	7.469000	7.482000	7.654000	7.182000	6.478000
Africa	6.573000	6.625000	6.699000	6.706000	6.70300	6.640000	6.501000	6.187000	5.724000	5.351000	5.077000	4.900000
Albania	6.230000	6.546000	6.230000	5.259000	4.60000	3.900000	3.409000	3.150000	2.786000	2.384000	1.946000	1.640000
Algeria	7.278000	7.384000	7.648000	7.648000	7.57200	7.175000	6.315000	5.302000	4.120000	2.885000	2.384000	2.724000
Angola	6.000000	6.500000	6.900000	7.300000	7.50000	7.456000	7.456000	7.400000	7.100000	6.750000	6.550000	6.350000
...	...	...	...	...	...	...	...	...	...	...	...	...
Yemen	7.800000	7.900000	8.000000	8.250000	8.50000	8.600000	8.800000	8.800000	8.200000	6.800000	5.900000	5.000000
Zambia	6.700000	6.950000	7.250000	7.300000	7.40000	7.250000	6.900000	6.600000	6.300000	6.100000	5.950000	5.600000
Zimbabwe	6.800000	7.000000	7.300000	7.400000	7.40000	7.300000	6.302000	5.373000	4.415000	3.885000	3.720000	3.885000
World Average	5.430629	5.480714	5.481782	5.265315	4.96754	4.611173	4.350528	4.067343	3.703577	3.362669	3.114859	2.978218
mean	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

250 rows × 12 columns

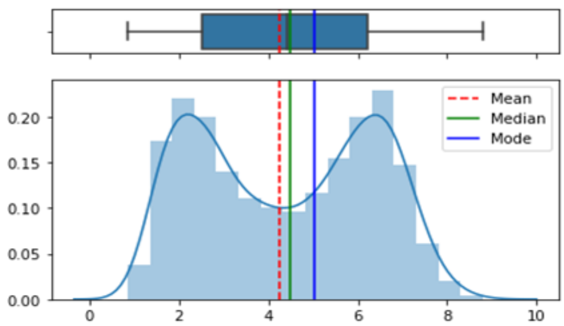
We will find **Pearson's coefficient of skewness** for our dataframe.

The equation for the Pearson's coefficient of skewness is

$$\textbf{SKp} = \frac{3(Mean - Median)}{Standard\ Deviation(\sigma)}$$

Now lets plot the distribution plot for this data frame. We have also included mean, mode and median lines in the graph.

```
plt.figure(figsize = (14,7))

mean = new_data_mean.mean()
median = new_data.median().median()
mode = new_data.mode().iloc[0].mean()

f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw= {"height_ratios": (0.2, 1)})
sns.boxplot(new_data, ax=ax_box)
ax_box.axvline(mea, color='r', linestyle='--')
ax_box.axvline(median, color='g', linestyle='-')
ax_box.axvline(mode, color='b', linestyle='-')

sns.distplot(new_data, ax=ax_hist)
ax_hist.axvline(mean, color='r', linestyle='--')
ax_hist.axvline(median, color='g', linestyle='-')
ax_hist.axvline(mode, color='b', linestyle='-')

plt.legend({'Mean':mean,'Median':median,'Mode':mode})

ax_box.set(xlabel='')
plt.show()
```



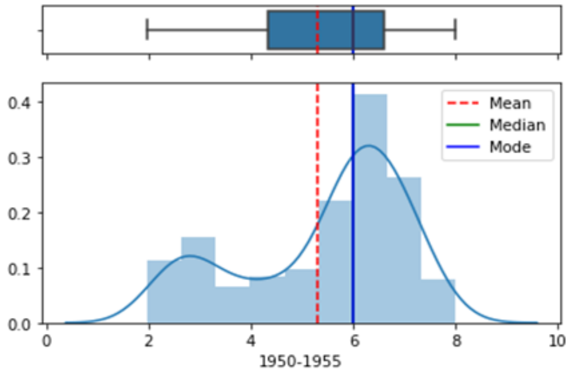
Conclusion: Here we can see that the data is equally distributed on both the sides of mean. And it is giving us Symmetric curve over the years 1950 to 2010.

But how is it being this symmetric?
For that we will make distplot for different years.
To see the change over the years, we will plot same graphs for year 1950-1955, 1980-1985 and 2005-2010. Here the total years 60 is diveded into 3 parts.

First plot the graph for year 1950-1955 and see the tendency.

```
mean5055 = new_data_mean['1950-1955']
median5055 = 6.0
```

```
mode5055 = new_data['1950-1955'].mode().iloc[0]

f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw= {"height_ratios": (0.2, 1)})
sns.boxplot(new_data['1950-1955'], ax=ax_box)
ax_box.axvline(mean5055, color='r', linestyle='--')
ax_box.axvline(median5055, color='g', linestyle='-')
ax_box.axvline(mode5055, color='b', linestyle='-')

sns.distplot(new_data['1950-1955'], ax=ax_hist)
ax_hist.axvline(mean5055, color='r', linestyle='--')
ax_hist.axvline(median5055, color='g', linestyle='-')
ax_hist.axvline(mode5055, color='b', linestyle='-')

plt.legend({'Mean': mean5055,'Median': median5055,'Mode': mode5055 })

ax_box.set(xlabel='')
plt.show()
```



Conclusion: From above graph we can see that the tail of the graph is long towards the negative edge of the x-axis. Also *mean<median<mode* . So the graph is becoming ***negativey skewed***.

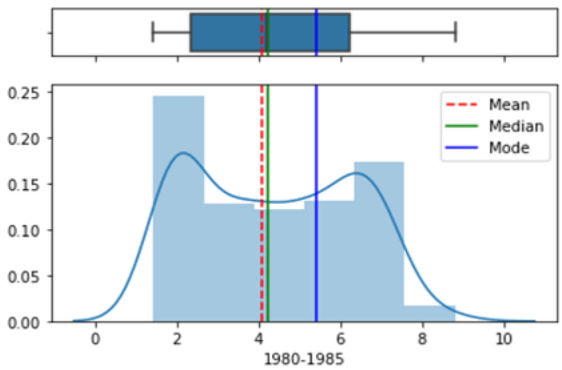
We can say that during the years 1950-1955, the fertility rate of most of the countries were higher than mean (because graph if higher to the right side of the mean and lower towards the left side of the mean).

```
std5055 = new_data_std[0]
pearsoncoeff5055 = 3*(mean5055 - median5055) / std5055
```

Pearson's coefficient of skewness for 1950-1955 = - 1.4646796041034003

Now plot graph for the year 1980-1985 and see the tendancy.

```
mean8085 = new_data_mean['1980-1985']
median8085 = 4.23
mode8085 = new_data['1980-1985'].mode().iloc[0]

f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw= {"height_ratios": (0.2, 1)})
sns.boxplot(new_data['1980-1985'], ax=ax_box)
ax_box.axvline(mean8085, color='r', linestyle='--')
ax_box.axvline(median8085, color='g', linestyle='-')
ax_box.axvline(mode8085, color='b', linestyle='-')

sns.distplot(new_data['1980-1985'], ax=ax_hist)
ax_hist.axvline(mean8085, color='r', linestyle='--')
ax_hist.axvline(median8085, color='g', linestyle='-')
ax_hist.axvline(mode8085, color='b', linestyle='-')

plt.legend({'Mean': mean8085,'Median': median8085,'Mode': mode8085})

ax_box.set(xlabel='')
plt.show()
```



Conclusion: From above graph we can see that the graph is becoming ***symmetric***. Also *mean=median<mode* . Mean, median and mode are very less far from each other.

We can say that from 1955 to 1980, fertility rate of some of the countries had started decreasing and so the negatively skewed graph started moving to get equally distributed to mean. We can say that the countries were 50% - 50% divided on *fertility rate > mean* and *fertility rate < mean*.

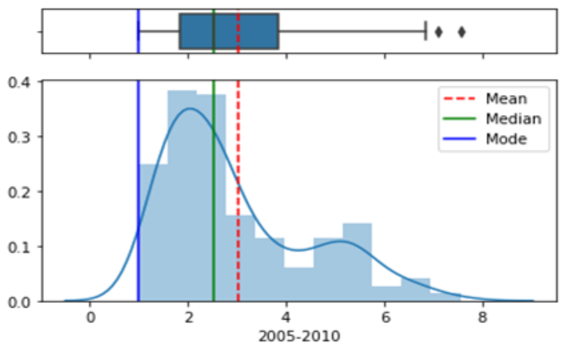
```
std8085 = new_data_std[6]
```

```
pearsoncoeff8085 = 3*(mean8085 - median8085) / std8085
```

Pearson's coefficient of skewness for 1980-1985 = - 0.9624275516198356

Now let's plot the graph for the year 2005-2010 (30 years after 1985) and see the tendency.

```
mean510 = new_data_mean['2005-2010']
median510 = 2.539
mode510 = new_data['2005-2010'].mode().iloc[0]

f, (ax_box, ax_hist) = plt.subplots(2, sharex=True, gridspec_kw= {"height_ratios": (0.2, 1)})
sns.boxplot(new_data['2005-2010'], ax=ax_box)
ax_box.axvline(mean510, color='r', linestyle='--')
ax_box.axvline(median510, color='g', linestyle='-')
ax_box.axvline(mode510, color='b', linestyle='-')

sns.distplot(new_data['2005-2010'], ax=ax_hist)
ax_hist.axvline(mean510, color='r', linestyle='--')
ax_hist.axvline(median510, color='g', linestyle='-')
ax_hist.axvline(mode510, color='b', linestyle='-')

plt.legend({'Mean':mean510,'Median':median510,'Mode':mode510})

ax_box.set(xlabel='')
plt.show()
```



Conclusion: From above graph we can see that the tail of the graph is long towards the positive edge of the x-axis. Also *mean>median>mode* . So the graph is becoming ***positively skewed*** from previously symmetric over the years 1985 to 2010.
We can say that during the years 2005-2010, the fertility rate of most of the countries have became lower than mean (because graph if higher to the left side of the mean and lower towards the right side of the mean).

```
std510 = new_data_std[11]
pearsoncoeff510 = 3*(mean510 - median510) / std510
```

Pearson's coefficient of skewness for 1980-1985 = - 0.991976218080132

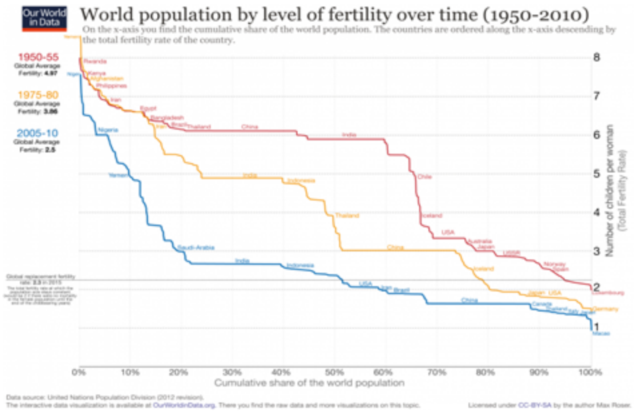
These three graph shows us the transition of the fertility rate for all countries from more than mean to become less than mean. And that's why the final graph (plotted first) is becoming symmetric and having peaks on both sides of mean.

Explanation of code (functions used in plotting):

- matplotlib library as plt.
- seaborn library as sns.
- Subplots to make it convenient to create common layouts to subplots, including enclosing figure object, in a single cell. 1st argument: ncols, nrows; 2nd argument: sharex = True (common x axis for both boxplot and distplot); 3rd argument: gridspac_kw = to create the grid the subplots are placed on.
- Axvline argument: Add vertical line across the axes. Pass data as 1st argument, can define colour, linewidth, linestyle etc.
- plt.legend: to provide legend name to the variable plotted.

The reasons for change –

- Empowerrment of women (Increasing Access to education and increasing labour participation)
- Declining Child mortality
- Rising cost of bringing up children



Conclusion: The width given to each country in this chart corresponds to the share of each country's population in the total global population. This is why China and India are so wide.

From this we can say that „

Globally, the fertility rate has been fallen to 2.5 childern per woman and low fertility rates are the norm in most of the world.

The 65% of the world's population live in countries with the fertility rate below 3 childern per woman.

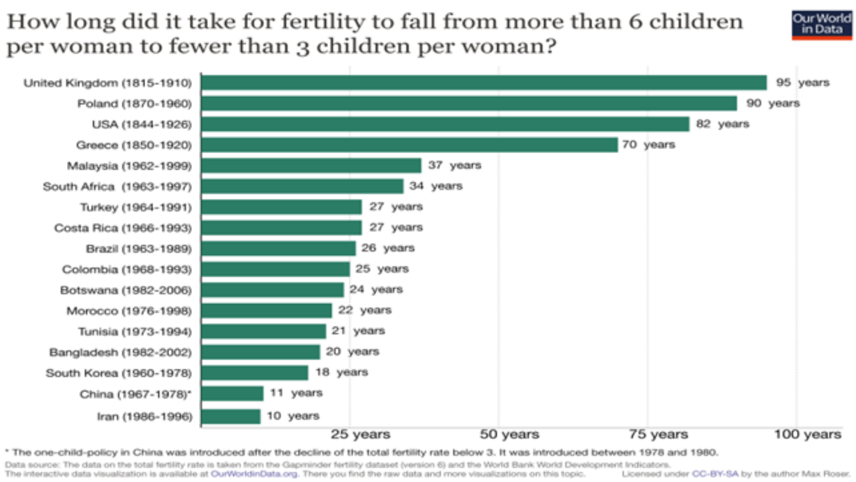
We also see convergence in fertility rates: the countries that already had low fertility rates in 1950s only slightly decreased fertility rates further, while many of the countries that had the highest fretility back than saw a rapid reduction of the number of children per woman.

It took Iran only 10 years for fertility rate to fall from more than 6 children per woman to fewer than 3 children per woman. China made this transition in 11 years – before the introduction of the one-child policy.

The speed with which countries can make transition to low fertility rate has increased over time.

Charts:

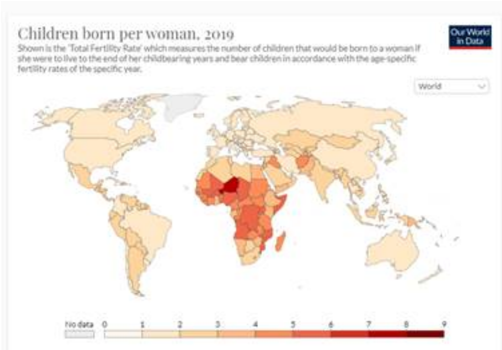
- How long did it take for fertility rate to fall from 6 children per woman to fewer than 3 children per woman.



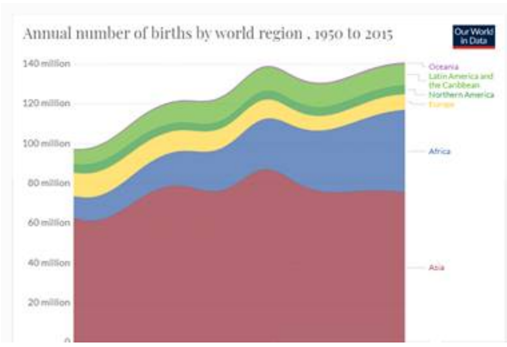
- Age of merrige of woman and Merital fertility rate

Country or Region	Mean age at first marriage	Births per married women	Percentage never married	Total fertility rate
Belgium	24.9	6.8	—	6.2
France	25.3	6.5	10	5.8
Germany	26.6	5.6	—	5.1
England	25.2	5.4	12	4.9
Netherlands	26.5	5.4	—	4.9
Scandinavia	26.1	5.1	14	4.5

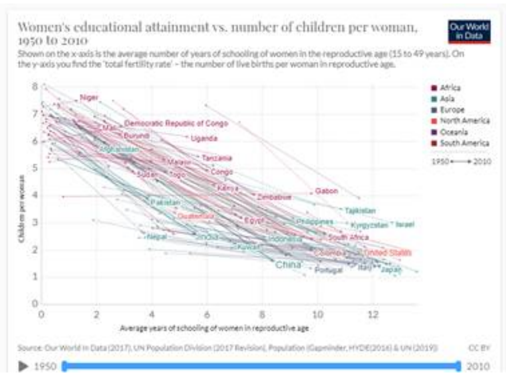
- Children borm per woman, 2019



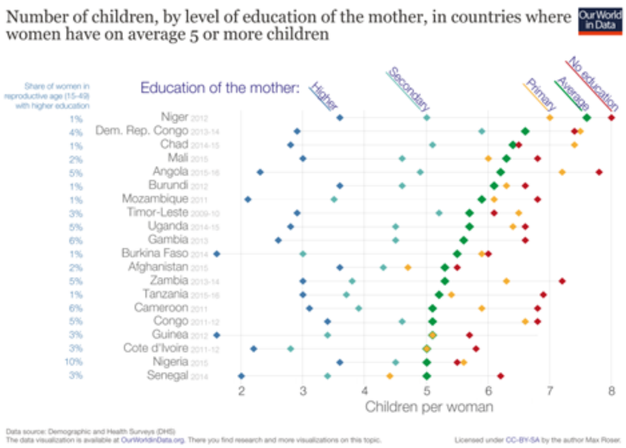
- Birth rates:



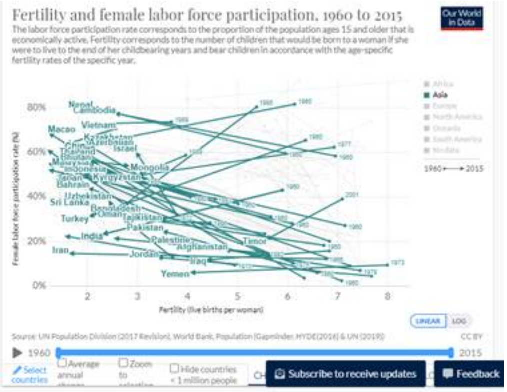
- Woman's Educational Attainment vs. No of children per woman:



- Number of children, by level of education of the mother, in countries where women have on an average 5 or more children.



- Fertility and female labour force participation:



References:

- Book: Alberto Cairo - The Functional Art_ An introduction to information graphics and visualization (2012, New Riders)
- Fertility Rate - Our World in Data