```
pip install researchpy
```

```
Requirement already satisfied: researchpy in /usr/local/lib/python3.10/dist-packages (0.3.5)
Requirement already satisfied: scipy in /usr/local/lib/python3.10/dist-packages (from researchpy) (1.11.3)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (from researchpy) (1.23.5)
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (from researchpy) (1.5.3)
Requirement already satisfied: statsmodels in /usr/local/lib/python3.10/dist-packages (from researchpy) (0.14.0)
Requirement already satisfied: patsy in /usr/local/lib/python3.10/dist-packages (from researchpy) (0.5.3)
Requirement already satisfied: python-dateutil>=2.8.1 in /usr/local/lib/python3.10/dist-packages (from pandas->researchpy) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas->researchpy) (2023.3.post1)
Requirement already satisfied: six in /usr/local/lib/python3.10/dist-packages (from patsy->researchpy) (1.16.0)
Requirement already satisfied: packaging>=21.3 in /usr/local/lib/python3.10/dist-packages (from statsmodels->researchpy) (23.2)
```

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import scipy.stats as stats
import researchpy as rp
import statistics as stat
```

```
#!pip install researchpy
```

```
data = pd.read_excel("demographics_MENA.xlsx")
#data
```

```
data.head()
#data.shape
#data.tail()
#data.dtypes
```

	Time	Time Code	Country Name	Country Code	Age dependency ratio (% of working-age population) [SP.POP.DPND]	Age dependency ratio, old [SP.POP.DPND.OL]	Age dependency ratio, young [SP.POP.DPND.YG]	Birth rate, crude (per 1,000 people) [SP.DYN.CBRT.IN]	Death rate, crude (per 1,000 people) [SP.DYN.CDRT.IN]	Ferti tot [SP.DY
0	2013	YR2013	Morocco	MAR	51.115023	8.348314	42.766709	21.141000	5.796000	
1	2013	YR2013	Algeria	DZA	49.534709	7.428022	42.106687	25.236000	4.673000	

```
#data.info()
```

Middle

```
#data.nunique() "identify the continuous and categorical columns in the data"
#remarque len(dataframe)=nombre de lignes !!!!!!!!!!!!!!!!!!!!
missing_data=data.isna().sum()
perc_missingdata=(missing_data/(len(data)))*100
#missing_data
#perc_missingdata


#Some columns or variables can be dropped if they do not add value to our analysis.
data_EDA = data.drop(['Country Name','Time Code'],axis=1)


# Middle east north africa is not a country !!!!!!!!
data_EDA.rename(columns={'Country Code': 'Region'}, inplace=True)
```

```
#abbreviations of column names
data_EDA.rename(columns={
                         "Age dependency ratio (% of working-age population) [SP.POP.DPND]":"ADR"
                         ,"Age dependency ratio, old [SP.POP.DPND.OL]":"ADR_old"
                         ,"Age dependency ratio, young [SP.POP.DPND.YG]":"ADR_young"
                         ,"Birth rate, crude (per 1,000 people) [SP.DYN.CBRT.IN]":"BRC"
                         ,"Death rate, crude (per 1,000 people) [SP.DYN.CDRT.IN]":"DRC"
                         ,"Fertility rate, total (births per woman) [SP.DYN.TFRT.IN]":"FRT"
                         ,"Life expectancy at birth, female (years) [SP.DYN.LE00.FE.IN]":"LEB_F"
                         ,"Life expectancy at birth, male (years) [SP.DYN.LE00.MA.IN]":"LEB_M"
                         ,"Life expectancy at birth, total (years) [SP.DYN.LE00.IN]":"LEB"
                         ,"Mortality rate, adult, female (per 1,000 female adults) [SP.DYN.AMRT.FE]":"MR_A_F"
                         ,"Mortality rate, adult, male (per 1,000 male adults) [SP.DYN.AMRT.MA]":"MR_A_M"
                         ,"Mortality rate, infant (per 1,000 live births) [SP.DYN.IMRT.IN]":"MR_I"
                         ,"Mortality rate, infant, female (per 1,000 live births) [SP.DYN.IMRT.FE.IN]":"MR_I_F"
                         ,"Mortality rate, infant, male (per 1,000 live births) [SP.DYN.IMRT.MA.IN]":"MR_I_M"
                         ,"Mortality rate, neonatal (per 1,000 live births) [SH.DYN.NMRT]":"MR_n"
                         ,"Net migration [SM.POP.NETM]":"NM"
                         ,"Number of infant deaths [SH.DTH.IMRT]":"NID"
                         ,"Number of infant deaths, female [SH.DTH.IMRT.FE]":"NID_F"
                         ,"Number of infant deaths, male [SH.DTH.IMRT.MA]":"NID_M"
                         ,"Population growth (annual %) [SP.POP.GROW]":"PG_a"
                         ,"Population, female [SP.POP.TOTL.FE.IN]":"PF"
                         ,"Population, male [SP.POP.TOTL.MA.IN]":"PM"
                         ,"Population, total [SP.POP.TOTL]":"PT"
                         ,"Rural population [SP.RUR.TOTL]":"RP"
                         ,"Rural population growth (annual %) [SP.RUR.TOTL.ZG]":"RPG_a"
                         ,"Urban population [SP.URB.TOTL]":"UP"
                         ,"Urban population growth (annual %) [SP.URB.GROW]":"UPG_a"
                         ,"Population ages 65 and above, total [SP.POP.65UP.TO]":"PA>=65"
                         ,"Population ages 15-64, total [SP.POP.1564.TO]":"15<PA<64"
                         ,"Population ages 0-14, total [SP.POP.0014.TO]":"0<PA<14"
                        }, inplace=True)

#data_EDA.info()

#data_EDA.describe().T

data_EDA
```

	Time	Region	ADR	ADR_old	ADR_young	BRC	DRC	FRT	LEB_F	LEB_M	...	PF	PM
0	2013	MAR	51.115023	8.348314	42.766709	21.141000	5.796000	2.594000	73.976000	70.371000	...	16812735	16990792
1	2013	DZA	49.534709	7.428022	42.106687	25.236000	4.673000	2.957000	75.840000	73.430000	...	18629532	19371095
2	2013	LBY	57.857339	7.191778	50.665561	21.886000	5.056000	2.716000	74.999000	69.983000	...	2939742	3045479
3	2013	MEA	54.832405	6.678890	46.627576	24.785522	5.057609	2.970728	74.689240	70.216334	...	203833535	218956876
4	2014	MAR	51.105229	8.583568	42.521660	20.617000	5.741000	2.548000	74.390000	70.788000	...	17027786	17220817
5	2014	DZA	50.679694	7.629553	43.050141	25.404000	4.555000	3.004000	76.467000	73.803000	...	19004433	19755734
6	2014	LBY	57.198571	7.207130	49.991441	21.908000	5.430000	2.754000	75.301000	68.237000	...	2997221	3100543
7	2014	MEA	55.305743	6.833397	46.954245	24.538066	5.007248	2.974571	74.936272	70.517108	...	208078483	223586099
8	2015	MAR	51.189767	8.871757	42.318009	20.307000	5.705000	2.531000	74.777000	71.166000	...	17235856	17444602
9	2015	DZA	51.961774	7.887966	44.073808	25.405000	4.437000	3.041000	76.824000	74.456000	...	19390923	20152232
10	2015	LBY	56.752268	7.291311	49.460957	21.214000	5.427000	2.711000	75.369000	68.500000	...	3046854	3145381
11	2015	MEA	55.717843	7.000901	47.244190	24.015800	4.956075	2.951749	75.142270	70.832811	...	212191189	228315279
12	2016	MAR	51.295387	9.207596	42.087792	19.483000	5.674000	2.451000	75.176000	71.502000	...	17442227	17665036
13	2016	DZA	53.315437	8.186477	45.128959	25.166000	4.472000	3.051000	76.803000	74.693000	...	19783015	20556314
14	2016	LBY	56.204445	7.368905	48.835540	20.576000	5.480000	2.673000	75.474000	68.501000	...	3094517	3187679
15	2016	MEA	55.906616	7.176701	47.335708	23.045558	4.898086	2.871159	75.562196	70.966163	...	216061701	232855706
16	2017	MAR	51.453494	9.588176	41.865318	19.298000	5.642000	2.451000	75.664000	71.783000	...	17646668	17881446
17	2017	DZA	54.679122	8.506600	46.172523	24.755000	4.542000	3.050000	76.821000	74.700000	...	20175233	20961313
18	2017	LBY	55.390651	7.399805	47.990846	19.898000	5.284000	2.626000	75.674000	69.630000	...	3143706	3234556
19	2017	MEA	55.970221	7.369804	47.295638	22.555495	4.778578	2.838007	76.086039	71.368280	...	219860378	237025104
20	2018	MAR	51.686189	10.003394	41.682795	18.813000	5.637000	2.415000	76.086000	72.028000	...	17840191	18087320
21	2018	DZA	55.953864	8.843219	47.110645	24.074000	4.482000	3.023000	77.205000	74.966000	...	20564404	21362603
22	2018	LBY	54.322404	7.375025	46.947378	19.282000	5.245000	2.581000	75.877000	70.017000	...	3193468	3284325
23	2018	MEA	55.964417	7.576137	47.181817	21.887915	4.773046	2.783049	76.194284	71.642446	...	223852488	241221003
24	2019	MAR	51.919106	10.443099	41.476007	18.337000	5.658000	2.382000	76.448000	72.245000	...	18022621	18281786
25	2019	DZA	57.053988	9.195569	47.858419	23.298000	4.392000	2.988000	77.760000	75.238000	...	20948465	21756903

26	2019	LBY	53.163332	7.332296	45.831036	18.732000	5.454000	2.539000	75.766000	69.496000	...	3240076	3329012
27	2019	MEA	55.858771	7.789915	46.963652	21.058566	4.793649	2.700352	76.246402	71.862269	...	227893479	245308297
28	2020	MAR	52.074887	10.875293	41.199593	17.889000	6.073000	2.353000	76.308000	71.757000	...	18213816	18474956
29	2020	DZA	57.889467	9.500951	48.388515	22.431000	5.398000	2.942000	75.912000	73.082000	...	21318767	22132899

```
#data_EDA.columns
data_EDA.dtypes
```

```
     Time             int64
     Region          object
     ADR            float64
     ADR_old        float64
     ADR_young      float64
     BRC            float64
     DRC            float64
     FRT            float64
     LEB_F          float64
     LEB_M          float64
     LEB            float64
     MR_A_F         float64
     MR_A_M         float64
     MR_I           float64
     MR_I_F         float64
     MR_I_M         float64
     MR_n           float64
     NM               int64
     NID            float64
     NID_F          float64
     NID_M          float64
     PG_a           float64
     PF               int64
     PM               int64
     PT               int64
     RP               int64
     RPG_a          float64
     UP               int64
     UPG_a          float64
     PA>=65           int64
     15<PA<64         int64
     0<PA<14          int64
     dtype: object
```

```
data_MAR=data_EDA.iloc[:36:4, :]
data_DZA=data_EDA.iloc[1:36:4, :]
data_LBY=data_EDA.iloc[2:36:4, :]
data_MEA=data_EDA.iloc[3:36:4, :]
#data_MAR
#data_DZA
#data_LBY
#data_MEA
```

```
data_MEA
```

	Time	Region	ADR	ADR_old	ADR_young	BRC	DRC	FRT	LEB_F	LEB_M	...	PF	PM	
3	2013	MEA	54.832405	6.678890	46.627576	24.785522	5.057609	2.970728	74.689240	70.216334	...	203833535	218956876	42
7	2014	MEA	55.305743	6.833397	46.954245	24.538066	5.007248	2.974571	74.936272	70.517108	...	208078483	223586099	43
11	2015	MEA	55.717843	7.000901	47.244190	24.015800	4.956075	2.951749	75.142270	70.832811	...	212191189	228315279	44
15	2016	MEA	55.906616	7.176701	47.335708	23.045558	4.898086	2.871159	75.562196	70.966163	...	216061701	232855706	44
19	2017	MEA	55.970221	7.369804	47.295638	22.555495	4.778578	2.838007	76.086039	71.368280	...	219860378	237025104	45
23	2018	MEA	55.964417	7.576137	47.181817	21.887915	4.773046	2.783049	76.194284	71.642446	...	223852488	241221003	46
27	2019	MEA	55.858771	7.789915	46.963652	21.058566	4.793649	2.700352	76.246402	71.862269	...	227893479	245308297	47
31	2020	MEA	55.800832	8.016920	46.782678	20.599688	5.285240	2.660517	75.321682	70.812576	...	231463027	248503623	47
35	2021	MEA	55.667365	8.224893	46.528262	20.142055	5.545467	2.629065	75.084944	70.640202	...	234775082	251392983	48

9 rows × 32 columns

```
#data_MAR.describe().T
#data_DZA.describe().T
#data_LBY.describe().T
#data_MEA.describe().T
```

```
data_MAR_POP=data_MAR[['Time', 'Region','ADR', 'ADR_old', 'ADR_young','NM',  'PG_a', 'PF', 'PM', 'PT', 'RP','RPG_a', 'UP', 'UPG_a', 'PA>=65', '15<PA<64', '0<
data_DZA_POP=data_DZA[['Time', 'Region','ADR', 'ADR_old', 'ADR_young','NM',  'PG_a', 'PF', 'PM', 'PT', 'RP','RPG_a', 'UP', 'UPG_a', 'PA>=65', '15<PA<64', '0<
data_LBY_POP=data_LBY[['Time', 'Region','ADR', 'ADR_old', 'ADR_young','NM',  'PG_a', 'PF', 'PM', 'PT', 'RP','RPG_a', 'UP', 'UPG_a', 'PA>=65', '15<PA<64', '0<
data_MEA_POP=data_MEA[['Time', 'Region','ADR', 'ADR_old', 'ADR_young','NM',  'PG_a', 'PF', 'PM', 'PT', 'RP','RPG_a', 'UP', 'UPG_a', 'PA>=65', '15<PA<64', '0<


data_MAR_MOR=data_MAR[['Time', 'Region', 'DRC', 'MR_A_F', 'MR_A_M', 'MR_I', 'MR_I_F', 'MR_I_M','MR_n',  'NID', 'NID_F', 'NID_M']]
data_DZA_MOR=data_DZA[['Time', 'Region', 'DRC', 'MR_A_F', 'MR_A_M', 'MR_I', 'MR_I_F', 'MR_I_M','MR_n',  'NID', 'NID_F', 'NID_M']]
data_LBY_MOR=data_LBY[['Time', 'Region', 'DRC', 'MR_A_F', 'MR_A_M', 'MR_I', 'MR_I_F', 'MR_I_M','MR_n',  'NID', 'NID_F', 'NID_M']]
data_MEA_MOR=data_MEA[['Time', 'Region', 'DRC', 'MR_A_F', 'MR_A_M', 'MR_I', 'MR_I_F', 'MR_I_M','MR_n',  'NID', 'NID_F', 'NID_M']]


data_MAR_BIR=data_MAR[['Time', 'Region', 'BRC', 'FRT','LEB_F', 'LEB_M', 'LEB']]
data_DZA_BIR=data_DZA[['Time', 'Region', 'BRC', 'FRT','LEB_F', 'LEB_M', 'LEB']]
data_LBY_BIR=data_LBY[['Time', 'Region', 'BRC', 'FRT','LEB_F', 'LEB_M', 'LEB']]
data_MEA_BIR=data_MEA[['Time', 'Region', 'BRC', 'FRT','LEB_F', 'LEB_M', 'LEB']]



print("Médiane : ",stat.median(data_MAR['FRT']))

print("Mean : ",stat.mean(data_MAR['FRT']))

print("Mode : ",stat.mode(data_MAR['FRT']))

print("Quantiles : ",stat.quantiles(data_MAR['FRT']))

print("Variance : ",stat.pvariance(data_MAR_BIR['FRT']))

print("Standar_Deviation: ",stat.stdev(data_MAR_BIR['FRT']))

print("Variation_Coefficient: ",stats.variation(data_MAR_BIR['FRT']))

print("interquartile range : ",stats.iqr(data_MAR_BIR['FRT']))

print("Etendue : ",max(data_MAR_BIR['FRT'])-min(data_MAR_BIR['FRT']))
```

```
Médiane :  2.451
Mean :  2.4503333333333335
Mode :  2.451
Quantiles :  [2.3675, 2.451, 2.5395000000000003]
Variance :  0.007449333333333333
Standar_Deviation:  0.09154507086675938
Variation_Coefficient:  0.035223583400401916
interquartile range :  0.14900000000000002
Etendue :  0.266
```

Visualization

In this part, we are going to answer 6 questions:

---

1. How does the Age Dependency Ratio vary across different countries?
2. What is the trend in Crude Birth Rate and Crude Death Rate over time?
3. Is there a correlation between Fertility Rate and Life Expectancy?
4. How does the Mortality Rate vary between male and female infants?
5. Explore the Net Migration patterns over time.
6. What is the distribution of Population Growth across urban and rural areas?

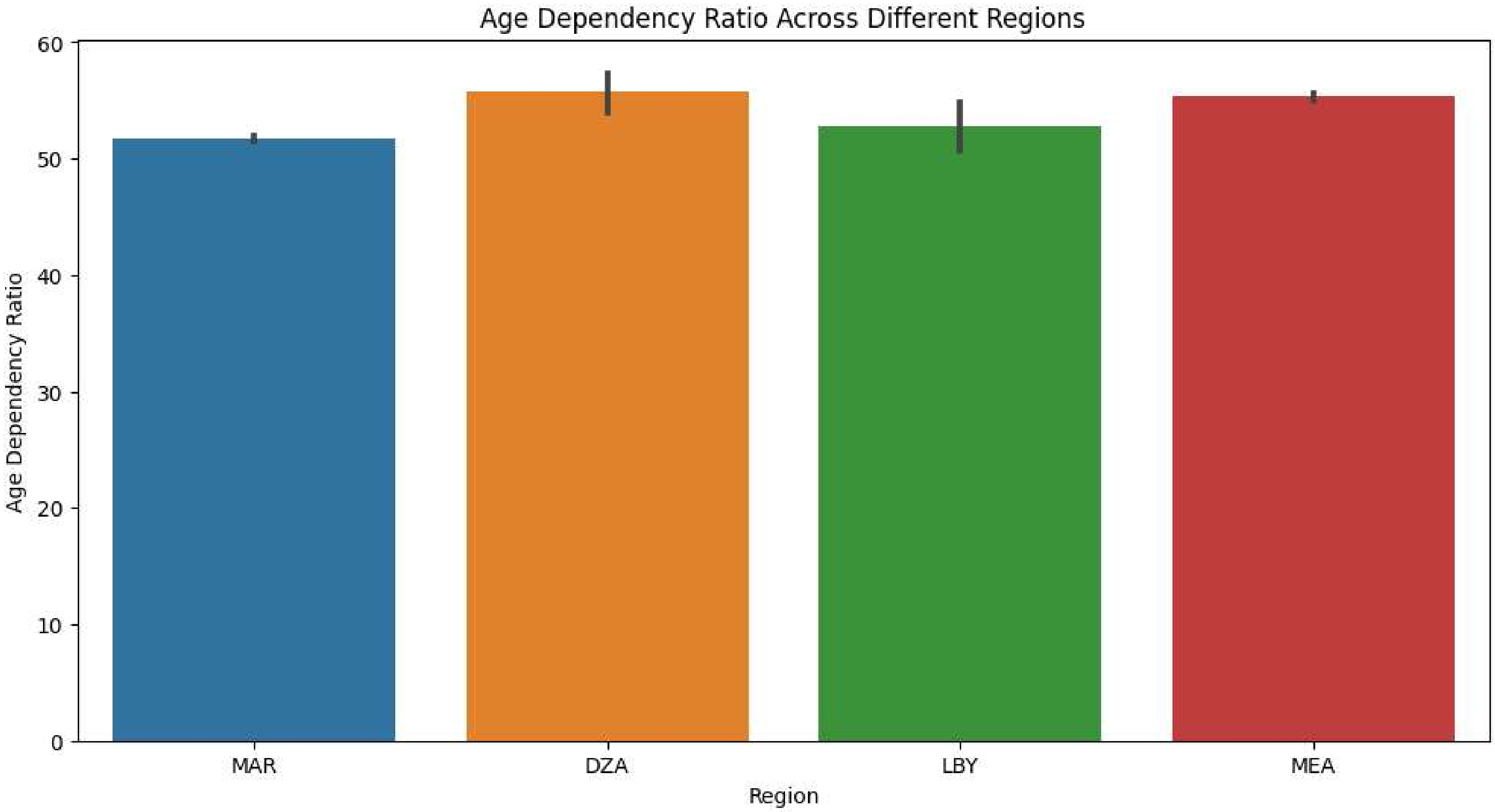
1. How does the Age Dependency Ratio vary across different countries?

> To answer this questions, we will use a **barplot**: What is a barplot, and where is it used?
>
> - A bar plot represents an aggregate or statistical estimate for a numeric variable with the height of each rectangle and indicates the uncertainty around that estimate using an error bar. Bar plots include 0 in the axis range, and they are a good choice when 0 is a meaningful value for the variable to take.
> - Used for categorical data.

> We used it in our case for:
>
> - Its simplicity and ease of interpretation make bar plots particularly suitable for displaying comparisons between different groups or categories

```
region_order = data_EDA['Region'].unique()
```

```
plt.figure(figsize=(12, 6))
sns.barplot(x='Region', y='ADR', data=data_EDA, order=region_order)
plt.xlabel('Region')
plt.ylabel('Age Dependency Ratio')
plt.title('Age Dependency Ratio Across Different Regions')
plt.show()
```



Interpretation:

> the Age Dependency Ratio across regions suggests that **DZA** has the highest demographic burden, followed by the **MEA**, **LBY**, and **MAR**, indicating **potential variations in population age structures** and associated economic implications.

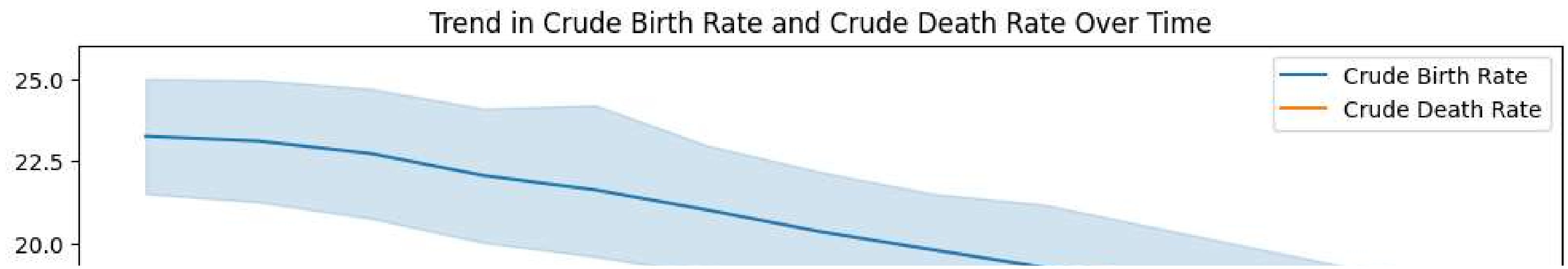
2. What is the trend in Crude Birth Rate and Crude Death Rate over time?

> To answer this questions, we will use a **lineplot**: What is a lineplot, and where is it used?
>
> - A lineplot is graphical representation that displays data points at specific time intervals, connecting them with lines to illustrate the trend or pattern of a numerical variable over time.

> A line plot visually represents the temporal trends in Crude Birth Rate and Crude Death Rate over time, **offering insights into the changing patterns** of these demographic indicators.

```
plt.figure(figsize=(12, 6))
sns.lineplot(x='Time', y='BRC', data=data_EDA, label='Crude Birth Rate')
sns.lineplot(x='Time', y='DRC', data=data_EDA, label='Crude Death Rate')

plt.xlabel('Time')
plt.ylabel('Rate per 1,000 people')
plt.title('Trend in Crude Birth Rate and Crude Death Rate Over Time')
plt.legend()
plt.show()
```



Interpretation:

> **Crude Birth Rate** line's suggests a gradual reduction in the number of births per 1,000 people over time, potentially indicating changes in fertility patterns or family planning measures.

> **Crude Death Rate** line's consistently low values reflect a relatively stable and low mortality rate per 1,000 people, implying sustained overall health and longevity in the population.

> The pic was in 2020 due to COVID 19.

3. Is there a correlation between Fertility Rate and Life Expectancy?

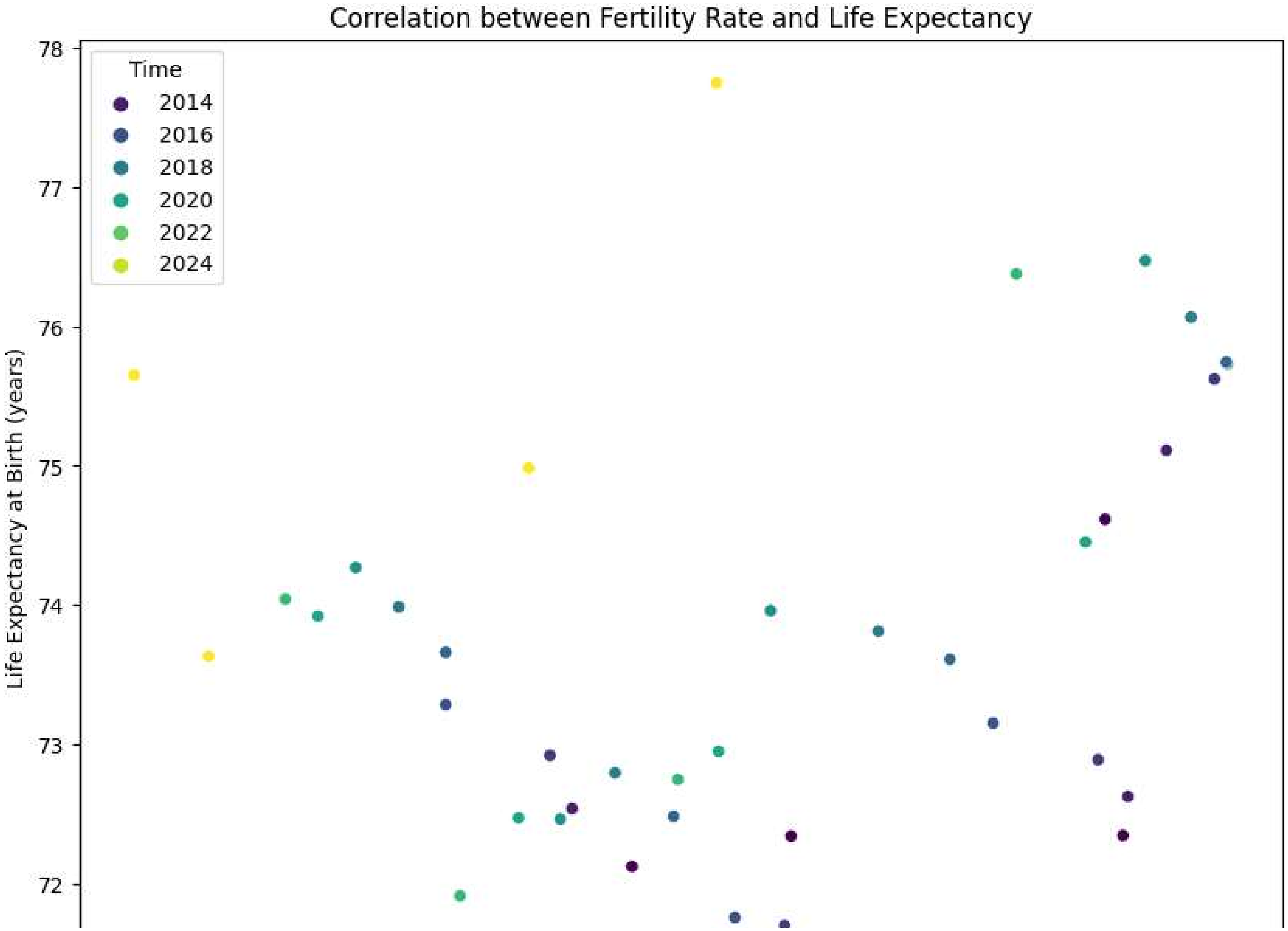
To answer this questions, we will use a **scatterplot**: What is a scatterplot, and where is it used?

- A scatterplot is

```
plt.figure(figsize=(10, 8))
sns.scatterplot(x='FRT', y='LEB', data=data_EDA, hue='Time', palette='viridis')

plt.xlabel('Fertility Rate (births per woman)')
plt.ylabel('Life Expectancy at Birth (years)')
plt.title('Correlation between Fertility Rate and Life Expectancy')

plt.show()
```



Correlation between Fertility Rate and Life Expectancy
Life Expectancy at Birth (years)
Time
2014
2016
2018
2020
2022
2024
78
77
76
75
74
73
72

```
from scipy.stats import pearsonr
plt.figure(figsize=(10, 8))

sns.regplot(x='FRT', y='LEB', data=data_EDA, scatter_kws={'s': 50}, line_kws={'color': 'red'})


plt.xlabel('Fertility Rate (births per woman)')
plt.ylabel('Life Expectancy at Birth (years)')
plt.title('Correlation between Fertility Rate and Life Expectancy')

plt.show()
correlation_coefficient, _ = pearsonr(data_EDA['FRT'].dropna(), data_EDA['LEB'].dropna())
print(correlation_coefficient)
```

Interpretation:

> The correlation coefficient of 0.2692 suggests a modest **positive relationship** between Fertility Rate and Life Expectancy. This means that, on average, as the number of births per woman increases, there is a slight tendency for life expectancy to also increase, but the connection is not very strong.

4. How does the Mortality Rate vary between male and female infants?

To answer this questions, we will use a **boxplot**: What is a boxplot, and where is it used?

- A boxplot is used to compare and contrast many groups.
- It highlights the median, and the outliers.
- It includes the interquartile range.
- the solid line in my boxplot refers to the median.

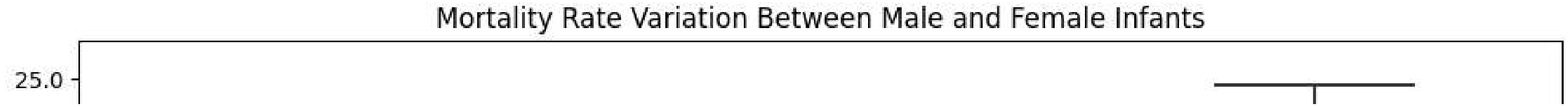
```
data_EDA.columns
#"Mortality rate, infant (per 1,000 live births) [SP.DYN.IMRT.IN]":"MR_I"
#"Mortality rate, infant, female (per 1,000 live births) [SP.DYN.IMRT.FE.IN]":"MR_I_F"
#"Mortality rate, infant, male (per 1,000 live births) [SP.DYN.IMRT.MA.IN]":"MR_I_M"
```

```
Index(['Time', 'Region', 'ADR', 'ADR_old', 'ADR_young', 'BRC', 'DRC', 'FRT',
       'LEB_F', 'LEB_M', 'LEB', 'MR_A_F', 'MR_A_M', 'MR_I', 'MR_I_F', 'MR_I_M',
       'MR_n', 'NM', 'NID', 'NID_F', 'NID_M', 'PG_a', 'PF', 'PM', 'PT', 'RP',
       'RPG_a', 'UP', 'UPG_a', 'PA>=65', '15<PA<64', '0<PA<14'],
      dtype='object')
```

```python
data_melted = data_EDA.melt(id_vars=['Time'], value_vars=['MR_I', 'MR_I_F', 'MR_I_M'],
                            var_name='Mortality Type', value_name='Mortality Rate')
plt.figure(figsize=(12, 8))
sns.boxplot(x='Mortality Type', y='Mortality Rate', data=data_melted, palette='viridis')

plt.xlabel('Mortality Type')
plt.ylabel('Mortality Rate (per 1,000 live births)')
plt.title('Mortality Rate Variation Between Male and Female Infants')

plt.show()
```



Interpretation:

> The boxplots reveal that the median Mortality Rates **differ among the age and gender groups**, with female infants having a slightly lower median rate (17.5) compared to all infants (19), while male infants show a slightly higher median rate (21). Additionally, the absence of outliers suggests a **consistent distribution of Mortality** Rates within these specified age and gender ranges.

5. Explore the Net Migration patterns over time.

To answer this questions, we will use a **lineplot**: What is a lineplot, and where is it used?

- A lineplot is employed to illustrate the temporal trends in Net Migration, to show thr representation of how this demographic indicator changes over time.

```
plt.figure(figsize=(12, 6))
sns.lineplot(x='Time', y='NM', data=data_EDA)

plt.xlabel('Time')
plt.ylabel('Net Migration')
plt.title('Net Migration Patterns Over Time')
plt.show()
```