INFO 474 Assignment 4 EC

In [1]:
```
import pandas as pd
import numpy as np
```

In [2]:
```
fertility_df = pd.read_csv("fertility_rates.csv")
life_df = pd.read_csv("life_expectancy.csv")
population_df = pd.read_csv("population.csv")
```

In [3]:
```
print(fertility_df.shape)
print(life_df.shape)
print(population_df.shape)
```

```
(3010, 8)
(6768, 8)
(9563, 8)
```

In [4]:
```
fertility_df.columns
```

Out[4]:
```
Index(['LOCATION', 'INDICATOR', 'SUBJECT', 'MEASURE', 'FREQUENCY', '
TIME',
       'Value', 'Flag Codes'],
      dtype='object')
```

In [5]:
```
life_df.columns
```

Out[5]:
```
Index(['LOCATION', 'INDICATOR', 'SUBJECT', 'MEASURE', 'FREQUENCY', '
TIME',
       'Value', 'Flag Codes'],
      dtype='object')
```

In [6]:
```
population_df.columns
```

Out[6]:
```
Index(['LOCATION', 'INDICATOR', 'SUBJECT', 'MEASURE', 'FREQUENCY', '
TIME',
       'Value', 'Flag Codes'],
      dtype='object')
```

In [7]:
```
f_df = fertility_df[fertility_df['SUBJECT'] == 'TOT'].drop(columns=['INDICATOR', 'SUBJECT', 'MEASURE', 'FREQUENCY', 'Flag Codes'])
l_df = life_df[life_df['SUBJECT'] == 'TOT'].drop(columns=['INDICATOR', 'SUBJECT', 'MEASURE', 'FREQUENCY', 'Flag Codes'])
p_df = population_df[population_df['SUBJECT'] == 'TOT'][population_df['MEASURE'] == 'MLN_PER'].drop(columns=['INDICATOR', 'SUBJECT', 'MEASURE', 'FREQUENCY', 'Flag Codes'])
```

```
/usr/local/lib/python3.7/site-packages/ipykernel_launcher.py:3: UserWarning: Boolean Series key will be reindexed to match DataFrame index.
  This is separate from the ipykernel package so we can avoid doing imports until
```

In [8]:
```
f_df.columns = ['location', 'time', 'fertility_rate']
l_df.columns = ['location', 'time', 'life_expectancy']
p_df.columns = ['location', 'time', 'pop_mlns']
```

In [9]:
```
df = f_df.merge(l_df, on=['location', 'time']).merge(p_df, on=['location', 'time'])
```

In [10]:
```
df = df.round({'pop_mlns': 5})
```

In [11]:
```
df.isna().sum()
```

Out[11]:
```
location           0
time               0
fertility_rate     0
life_expectancy    0
pop_mlns           0
dtype: int64
```

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

Out[12]:
```
(1991, 5)
```

In [13]:
```
df.to_csv('data.csv')
```