Objective(s):

- This activity aims to demonstrate how to apply simple linear regression analysis to solve regression problem

Intended Learning Outcomes (ILOs):

- Demonstrate how to solve regression problems using simple linear regression
- Use the linear regression model to predict the target value

Resources:

- Jupyter Notebook

Files:

- Life Expectancy Data.csv

Submission Requirements:

- PDF containing initial EDA and Data Wrangling
- PDF showing demonstration of simple linear regression.
- Submit a link to the colab file through the comment section.

```
pip install pandas numpy matplotlib seaborn scikit-learn
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (2.0.3)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (1.25.2)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.10/dist-packages (3.7.1)
Requirement already satisfied: seaborn in /usr/local/lib/python3.10/dist-packages (0.13.1)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-packages (1.2.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2023.4)
Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2024.1)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.2.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (4.51.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.4.5)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (24.0)
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (3.1.2)
Requirement already satisfied: scipy>=1.3.2 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.11.4)
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.4.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (3.4.0)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
```

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

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn import metrics
```

```
df = pd.read_csv('/content/Life Expectancy Data.csv')
```

```
print(df.head())
```

```
       Country  Year      Status  Life expectancy   Adult Mortality  \
0  Afghanistan  2015  Developing              65.0            263.0
1  Afghanistan  2014  Developing              59.9            271.0
2  Afghanistan  2013  Developing              59.9            268.0
3  Afghanistan  2012  Developing              59.5            272.0
4  Afghanistan  2011  Developing              59.2            275.0

   infant deaths  Alcohol  percentage expenditure  Hepatitis B  Measles   ...  \
0             62     0.01               71.279624         65.0      1154  ...
1             64     0.01               73.523582         62.0       492  ...
2             66     0.01               73.219243         64.0       430  ...
3             69     0.01               78.184215         67.0      2787  ...
4             71     0.01                7.097109         68.0      3013  ...

   Polio  Total expenditure  Diphtheria    HIV/AIDS         GDP  Population  \
0    6.0               8.16         65.0        0.1  584.259210  33736494.0
1   58.0               8.18         62.0        0.1  612.696514    327582.0
2   62.0               8.13         64.0        0.1  631.744976  31731688.0
3   67.0               8.52         67.0        0.1  669.959000   3696958.0
4   68.0               7.87         68.0        0.1   63.537231   2978599.0

    thinness  1-19 years   thinness 5-9 years  \
0                   17.2                 17.3
1                   17.5                 17.5
2                   17.7                 17.7
3                   17.9                 18.0
4                   18.2                 18.2

   Income composition of resources  Schooling
0                            0.479       10.1
1                            0.476       10.0
2                            0.470        9.9
3                            0.463        9.8
4                            0.454        9.5

[5 rows x 22 columns]
```

```
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2938 entries, 0 to 2937
Data columns (total 22 columns):
 #   Column                           Non-Null Count  Dtype
---  ------                           --------------  -----
 0   Country                          2938 non-null   object
 1   Year                             2938 non-null   int64
```

```
 2   Status                           2938 non-null   object
 3   Life expectancy                  2928 non-null   float64
 4   Adult Mortality                  2928 non-null   float64
 5   infant deaths                    2938 non-null   int64
 6   Alcohol                          2744 non-null   float64
 7   percentage expenditure           2938 non-null   float64
 8   Hepatitis B                      2385 non-null   float64
 9   Measles                          2938 non-null   int64
 10   BMI                             2904 non-null   float64
 11  under-five deaths                2938 non-null   int64
 12  Polio                            2919 non-null   float64
 13  Total expenditure                2712 non-null   float64
 14  Diphtheria                       2919 non-null   float64
 15   HIV/AIDS                        2938 non-null   float64
 16  GDP                              2490 non-null   float64
 17  Population                       2286 non-null   float64
 18   thinness  1-19 years            2904 non-null   float64
 19   thinness 5-9 years             2904 non-null   float64
 20  Income composition of resources  2771 non-null   float64
 21  Schooling                        2775 non-null   float64
dtypes: float64(16), int64(4), object(2)
memory usage: 505.1+ KB
None
```

```
print(df.describe())
```

```
       count  2938.000000       2928.000000      2928.000000    2938.000000
       mean   2007.518720         69.224932       164.796448      30.303948
       std       4.613841          9.523867       124.292079     117.926501
       min    2000.000000         36.300000         1.000000       0.000000
       25%    2004.000000         63.100000        74.000000       0.000000
       50%    2008.000000         72.100000       144.000000       3.000000
       75%    2012.000000         75.700000       228.000000      22.000000
       max    2015.000000         89.000000       723.000000    1800.000000

                  Alcohol  percentage expenditure  Hepatitis B       Measles   \
       count  2744.000000             2938.000000  2385.000000    2938.000000
       mean      4.602861              738.251295    80.940461    2419.592240
       std       4.052413             1987.914858    25.070016   11467.272489
       min       0.010000                0.000000     1.000000       0.000000
       25%       0.877500                4.685343    77.000000       0.000000
       50%       3.755000               64.912906    92.000000      17.000000
       75%       7.702500              441.534144    97.000000     360.250000
       max      17.870000            19479.911610    99.000000  212183.000000

                   BMI   under-five deaths         Polio  Total expenditure  \
       count  2904.000000         2938.000000  2919.000000         2712.00000
       mean     38.321247           42.035739    82.550188            5.93819
       std      20.044034          160.445548    23.428046            2.49832
       min       1.000000            0.000000     3.000000            0.37000
       25%      19.300000            0.000000    78.000000            4.26000
       50%      43.500000            4.000000    93.000000            5.75500
       75%      56.200000           28.000000    97.000000            7.49250
       max      87.300000         2500.000000    99.000000           17.60000

              Diphtheria      HIV/AIDS            GDP    Population  \
       count  2919.000000  2938.000000    2490.000000  2.286000e+03
       mean     82.324084     1.742103    7483.158469  1.275338e+07
       std      23.716912     5.077785   14270.169342  6.101210e+07
       min       2.000000     0.100000       1.681350  3.400000e+01
       25%      78.000000     0.100000     463.935626  1.957932e+05
       50%      93.000000     0.100000    1766.947595  1.386542e+06
       75%      97.000000     0.800000    5910.806335  7.420359e+06
       max      99.000000    50.600000  119172.741800  1.293859e+09

               thinness  1-19 years   thinness 5-9 years  \
       count            2904.000000          2904.000000
       mean                4.839704             4.870317
       std                 4.420195             4.508882
       min                 0.100000             0.100000
       25%                 1.600000             1.500000
       50%                 3.300000             3.300000
       75%                 7.200000             7.200000
       max                27.700000            28.600000

              Income composition of resources    Schooling
       count                      2771.000000  2775.000000
       mean                          0.627551    11.992793
       std                           0.210904     3.358920
       min                           0.000000     0.000000
       25%                           0.493000    10.100000
       50%                           0.677000    12.300000
       75%                           0.779000    14.300000
       max                           0.948000    20.700000
```

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

```
       Country                             0
       Year                                0
       Status                              0
       Life expectancy                    10
       Adult Mortality                    10
       infant deaths                       0
       Alcohol                           194
       percentage expenditure              0
       Hepatitis B                       553
       Measles                             0
        BMI                               34
       under-five deaths                   0
       Polio                              19
       Total expenditure                 226
       Diphtheria                         19
        HIV/AIDS                           0
       GDP                               448
       Population                        652
        thinness  1-19 years              34
        thinness 5-9 years                34
       Income composition of resources   167
       Schooling                         163
       dtype: int64
```

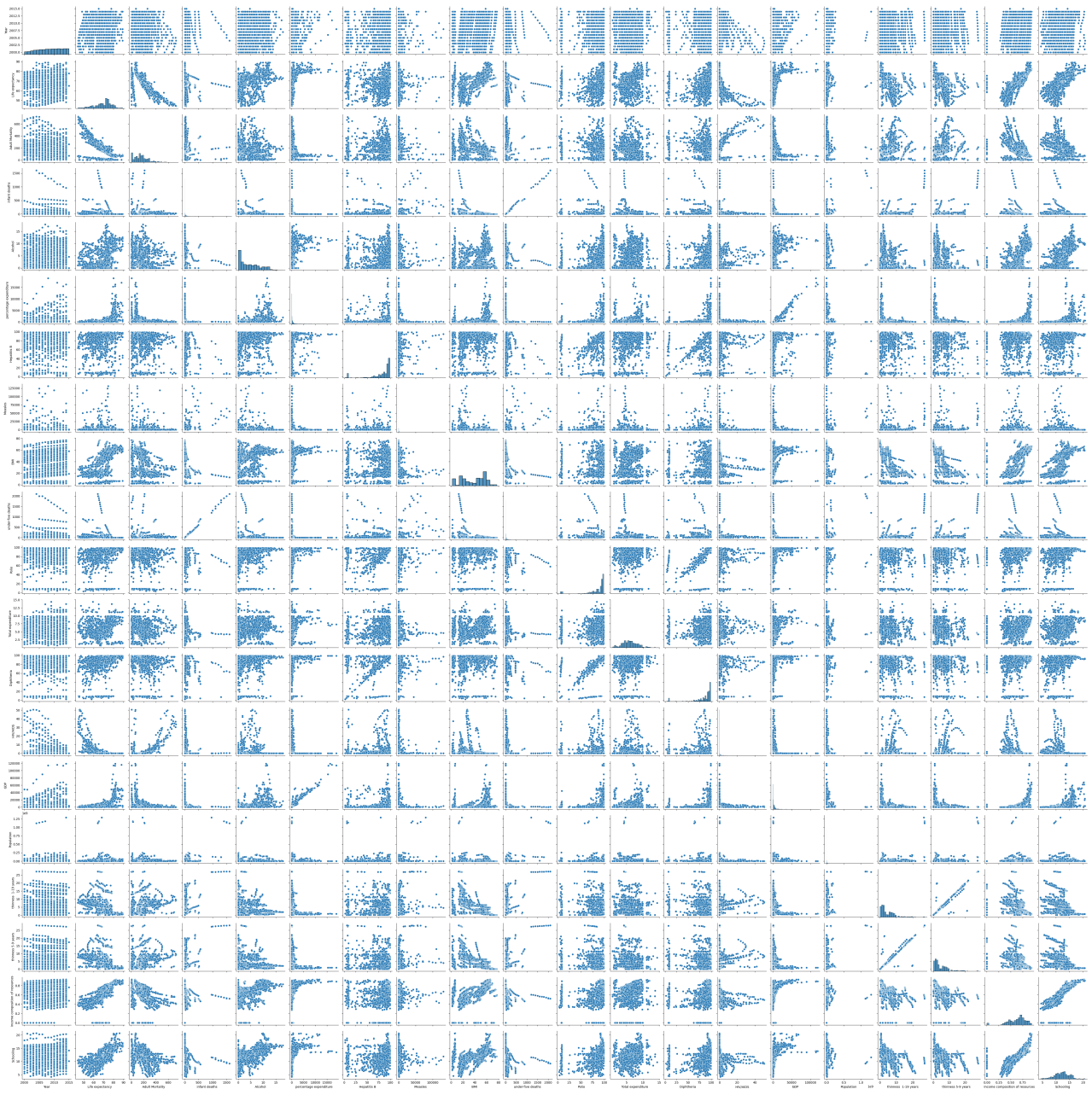
```
print(df.columns)
```

```
       Index(['Country', 'Year', 'Status', 'Life expectancy ', 'Adult Mortality',
              'infant deaths', 'Alcohol', 'percentage expenditure', 'Hepatitis B',
              'Measles ', ' BMI ', 'under-five deaths ', 'Polio', 'Total expenditure',
              'Diphtheria ', ' HIV/AIDS', 'GDP', 'Population',
              ' thinness  1-19 years', ' thinness 5-9 years',
              'Income composition of resources', 'Schooling'],
             dtype='object')
```

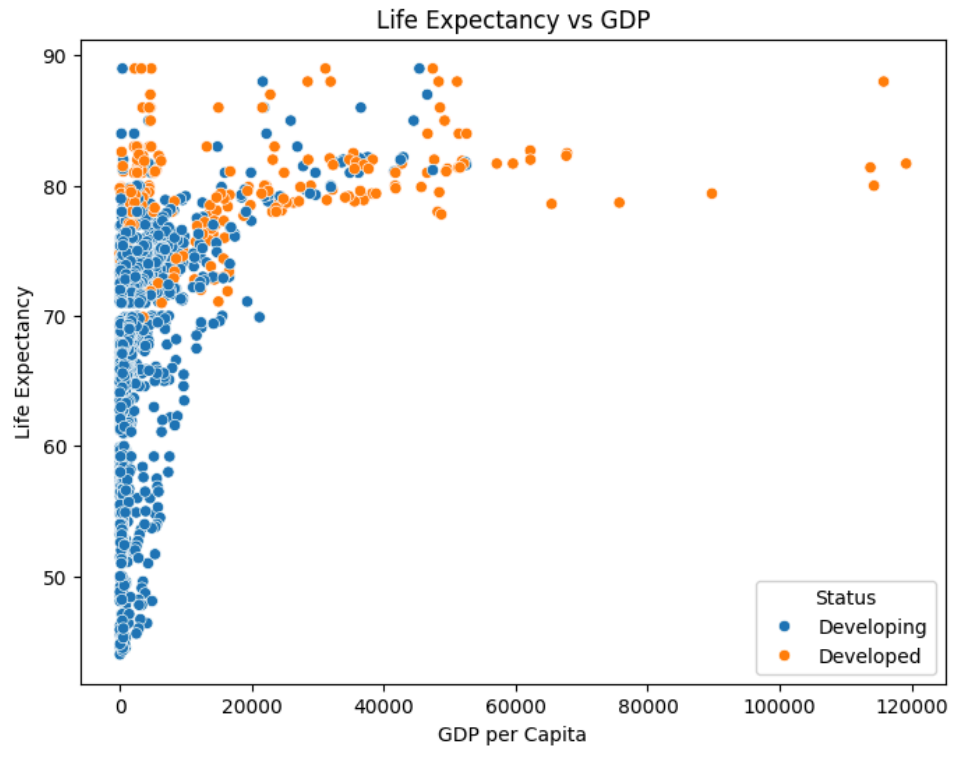
```
y = df['Life expectancy ']
X = df.drop('Adult Mortality', axis=1)
```

```
df = df.dropna()
```

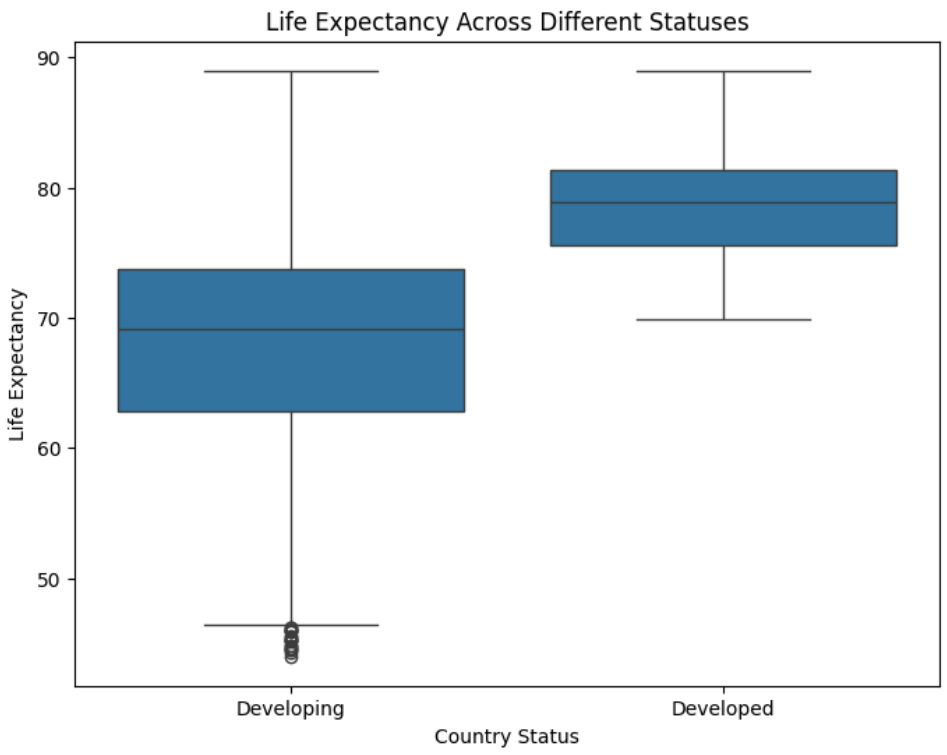
```
sns.pairplot(df)
plt.show()
```



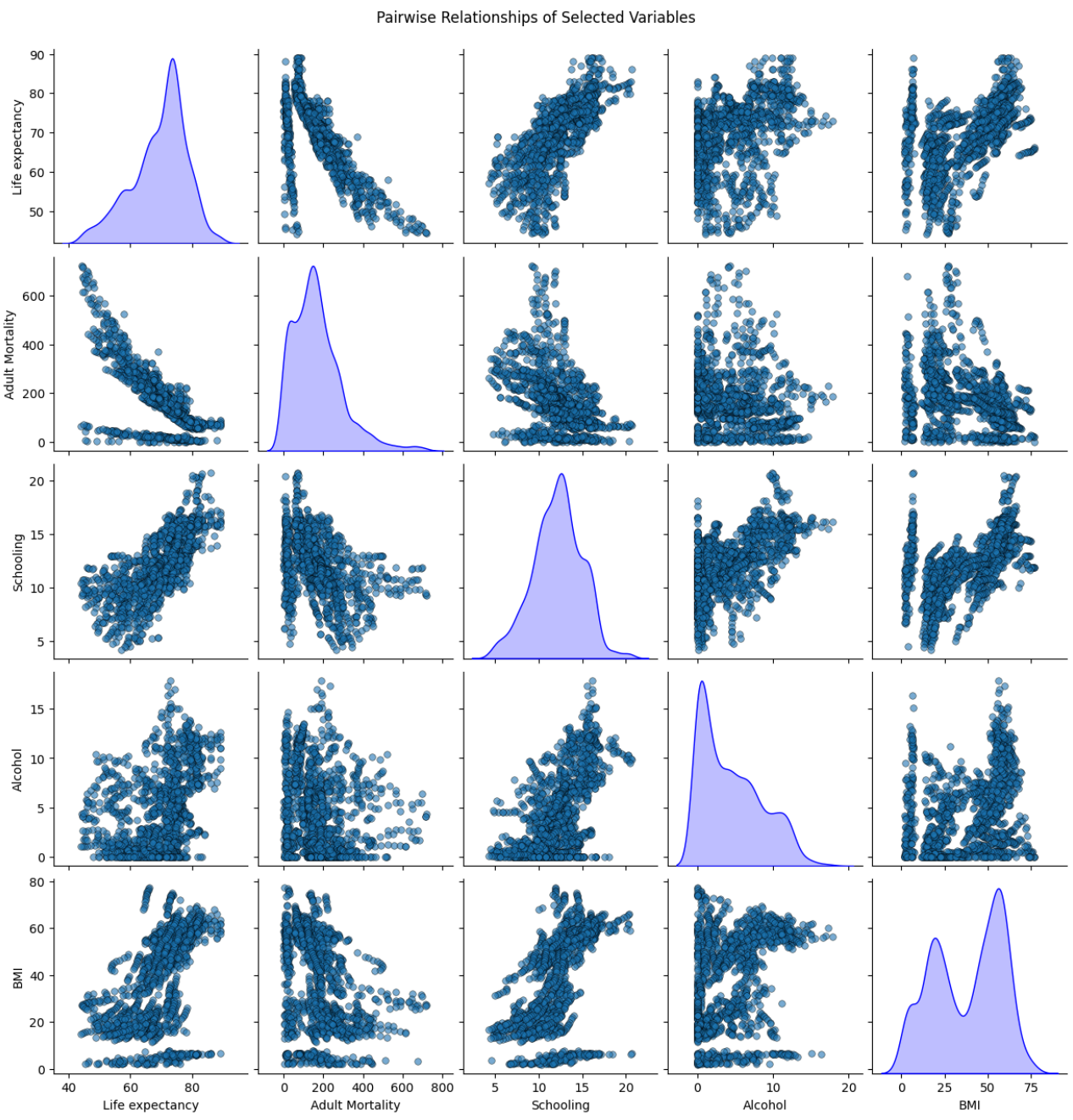

```
plt.figure(figsize=(8, 6))
sns.scatterplot(data=df, x='GDP', y='Life expectancy', hue='Status')
plt.title('Life Expectancy vs GDP')
plt.xlabel('GDP per Capita')
plt.ylabel('Life Expectancy')
plt.show()
```



```
plt.figure(figsize=(8, 6))
sns.boxplot(x='Status', y='Life expectancy', data=df)
plt.title('Life Expectancy Across Different Statuses')
plt.xlabel('Country Status')
plt.ylabel('Life Expectancy')
plt.show()
```

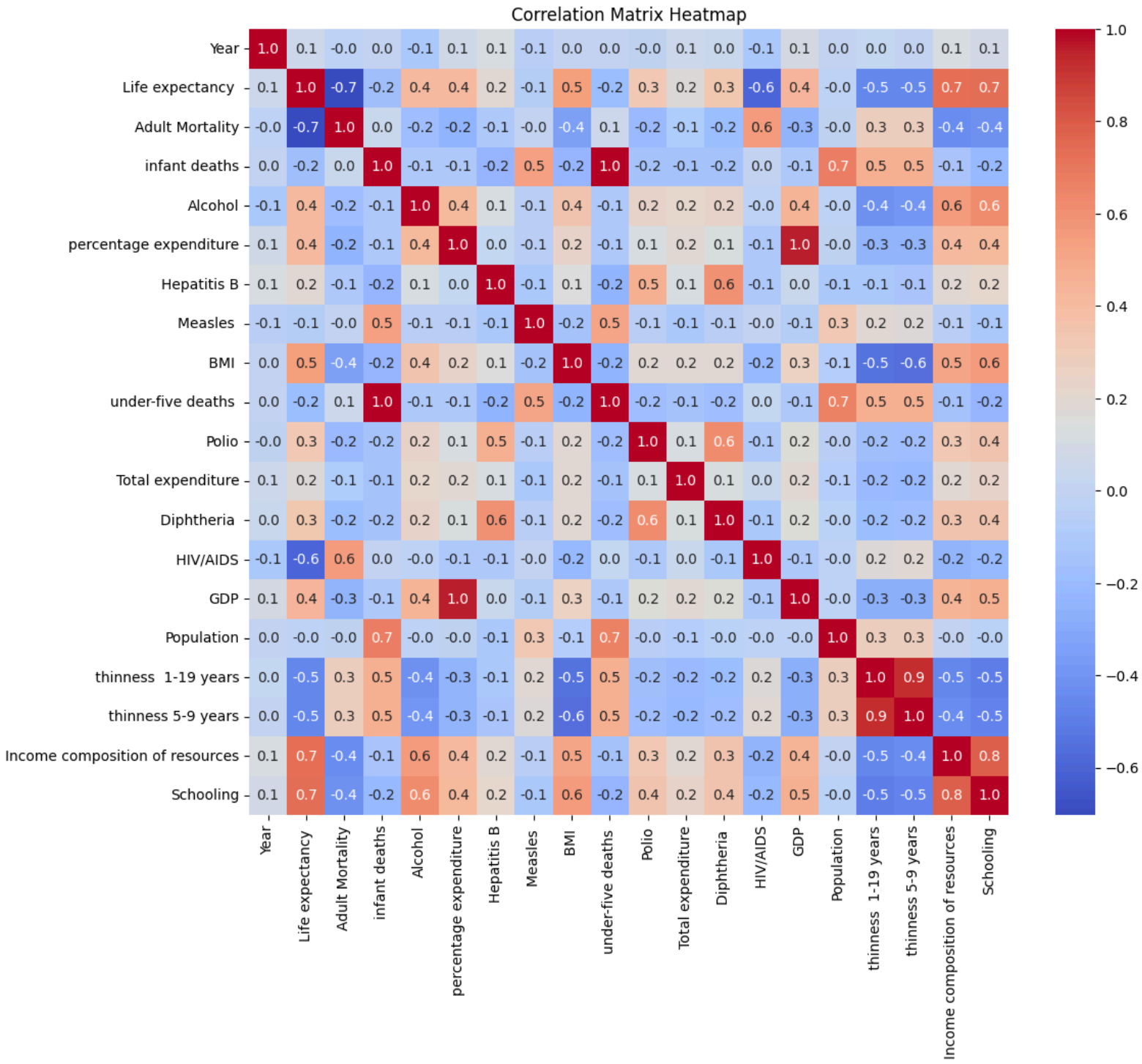



```
selected_columns = ['Life expectancy', 'Adult Mortality', 'Schooling', 'Alcohol', 'BMI']

sns.pairplot(df[selected_columns], diag_kind='kde',
             plot_kws={'alpha': 0.6, 's': 30, 'edgecolor': 'k'},
             diag_kws={'color': 'blue'})
plt.suptitle('Pairwise Relationships of Selected Variables', y=1.02)
plt.show()
```



```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```
numeric_df = df.select_dtypes(include=[np.number])
corr_matrix = numeric_df.corr()

plt.figure(figsize=(12, 10))
sns.heatmap(corr_matrix, annot=True, fmt=".1f", cmap='coolwarm', cbar=True)
plt.title('Correlation Matrix Heatmap')
plt.show()
```




```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=101)
```

```
df.columns = df.columns.str.strip()

y = df['Life expectancy']
X = df.drop(['Life expectancy', 'Adult Mortality'], axis=1).select_dtypes(include=[np.number])

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
lm = LinearRegression()
lm.fit(X_train, y_train)
```

```
▾ LinearRegression
LinearRegression()
```

```
print('Coefficients:', lm.coef_)
```

```
Coefficients: [-1.44674022e-01  1.12324315e-01 -1.72165941e-01  4.66604563e-04
 -5.40614359e-03 -9.19875192e-06  3.38352632e-02 -8.33715848e-02
  1.05688550e-02  1.26770393e-01  1.36349857e-02 -5.93219052e-01
  1.21654847e-05 -2.18271977e-10 -4.56813316e-03 -1.01343716e-01
  1.11395554e+01  1.12229857e+00]
```

```
predictions = lm.predict(X_test)
```

```
print("Predictions:")
print(predictions)
```

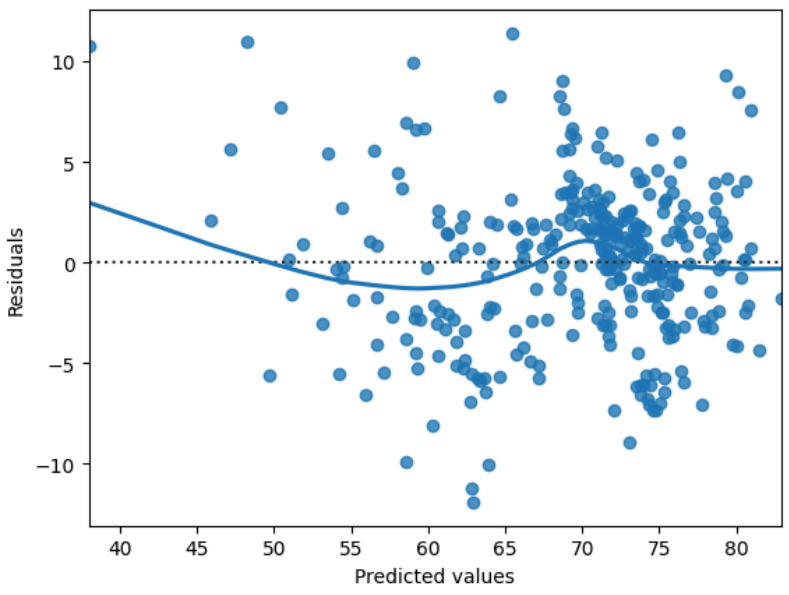
```
Predictions:
[69.71314639 72.94112919 80.80543867 56.62310998 54.19779443 53.09216089
 71.51690024 74.46003559 78.88801136 71.70663692 73.48290385 75.84375842
 77.40486436 68.63950348 67.03262903 71.76605451 78.27629797 80.96828161
 75.35415435 70.811055   79.27778183 71.80831356 70.99907764 73.99120725
 66.65940874 73.79250878 75.58879266 75.40798586 80.60893313 70.3370901
 61.80332735 63.61730885 71.41192563 80.93202329 76.19601845 78.84860508
 63.32936235 48.20318247 70.83549205 80.05081629 62.25083333 75.83947419
 72.68644102 70.59070534 76.56019865 63.72810677 73.39027407 59.0188955
 68.75018866 73.14510958 63.24710312 67.12915853 59.28468056 72.11614335
 75.16244271 56.50676163 75.24129343 75.52035548 72.26184231 62.06397474
 54.5148877  60.66394638 69.3792825  71.73606243 71.20130277 75.01336154
 60.73935894 74.62367238 78.5947477  68.54716921 71.88485555 61.09896338
 56.24089179 59.08161669 74.05241674 74.26693251 66.32903454 59.17315878
 77.97029303 81.51400162 70.09587724 73.1522761  73.4192565  71.16550609
 64.60666477 71.76106479 72.65047322 71.77125804 71.24278094 64.62163898
 74.72991688 75.04402145 75.05698013 71.56207464 79.43349754 68.63820888
 60.6426875  74.35700305 72.26125803 71.29826429 69.55529941 75.26820102
```

```
       69.57766971 61.77651642 76.28367658 73.75856218 79.77731399 75.80692178
       59.88627766 71.22640743 71.71872561 69.34528263 76.61131017 71.43944038
       73.33921269 75.03412746 47.14120226 71.71500824 70.38168542 66.81462279
       72.55447248 51.15724658 58.53552438 73.43552124 71.93687818 80.55997239
       80.05724494 61.26947937 69.1506482  54.00059824 78.20747867 72.18202492
       73.19447792 71.91184629 72.43142215 79.13517672 73.48375836 68.83422491
       73.47307502 78.56775229 76.62441262 69.25943492 70.24746886 53.46051732
       69.88670498 66.22100919 73.02202806 80.62475363 73.90923473 62.3588116
       74.65407384 78.44336383 58.03270574 70.65573512 69.36697745 76.12486941
       74.16924768 73.53921096 75.29019422 75.94149217 65.45010494 74.37733836
       54.43175354 70.512045   72.12357027 74.62916806 72.45680495 69.15177394
       72.11771045 63.81727035 68.7372489  74.88562152 72.25523895 71.85803326
       65.71182374 74.16052872 73.46448713 64.25077409 64.0161108  67.31838866
       68.92326468 73.67740249 74.83097568 76.45163458 73.65544726 66.70993095
       76.81130085 77.58622086 75.58344105 62.36651083 51.85318278 69.38119116
       57.63771466 56.64815583 78.70994173 54.3834231  55.08489766 63.93841497
       67.4657563  73.09208898 58.53889649 74.24485871 73.87779745 73.95916473
       77.03787545 58.28850088 37.95528199 74.34547663 74.79985515 59.48337308
       69.67511994 73.77856892 65.73498444 72.69930777 75.99358978 55.98146296
       75.89965073 69.75152845 61.60742113 75.33207623 62.31641379 77.7924167
       75.42332614 61.31585797 71.09221519 71.44597849 76.99493993 61.82256137
       62.16234835 62.89180681 75.64797273 75.9063062  71.56131049 63.84447319
       73.12114615 71.1090776  71.34201091 63.21147635 69.65045908 82.89656381
       73.58251749 65.97613124 80.4999062  71.61049882 73.69297011 60.60670869
       74.55554857 67.13000379 67.75875686 74.67858497 71.28688355 71.20557611
       71.37514115 73.09132958 66.68103677 74.35638376 73.06674592 70.99055407
       49.68835245 59.74746322 59.21266615 64.49997517 68.55803831 50.38795388
       74.80850266 71.52925876 63.98871907 59.16317545 69.34961972 79.05311546
       61.16226304 69.17606541 72.89312858 60.37233902 68.28282821 80.34649383
       78.44049593 66.16458635 62.82818906 79.09195203 62.80356349 65.62977863
       74.89494655 76.13948579 76.28267215 69.2005258  79.29171406 67.58333102
       76.35393536 71.74260218 67.89182564 60.33270331 71.38133769 65.33704306
       73.78068902 66.0411976  76.40298344 72.39597778 75.24893843 74.59128663
       71.49075012 50.91900998 69.26493391 71.0948048  72.07839098 78.4359072
       73.57233554 71.23142523 68.7467116  64.2158186  71.26559757 78.63040777
       73.48072948 57.13342697 45.87342065 60.57540489 71.91990335 71.51273249
       69.88816969 75.53929436 68.49513063 56.64363808 80.13877444 72.66820965
       65.52668137 76.57349199 77.90013019 62.73752734 67.96690442 58.57681249]
```

```
print('MAE:', metrics.mean_absolute_error(y_test, predictions))
print('MSE:', metrics.mean_squared_error(y_test, predictions))
print('RMSE:', np.sqrt(metrics.mean_squared_error(y_test, predictions)))
```

```
MAE: 3.0436393584103296
MSE: 15.079734201230751
RMSE: 3.883263344306017
```

```
sns.residplot(x=predictions, y=y_test, lowess=True)
plt.xlabel('Predicted values')
plt.ylabel('Residuals')
plt.show()
```



Conclusion

My data analysis that particularly through histograms, scatter plots, and pair plots. These visualizations highlighted the significant influence of economic factors, where GDP correlated positively with life expectancy, suggesting that higher economic prosperity generally leads to better health outcomes. I also examined health and lifestyle variables like 'Alcohol' consumption and 'Adult Mortality', which indicated that lifestyle choices significantly impact health. Overall, the analysis underscored the complex interplay of socioeconomic factors and personal habits in determining life expectancy, providing a comprehensive overview of the factors that contribute to health outcomes across different populations.