Intended Learning Outcome

1. Perform descriptive and correlation analysis to to analyze the dataset.
2. Interpret the results of descriptive and correlation analysis

Resources

- Personal Computer
- Jupyter Notebook
- Internet Connection

Instruction

1. Gather a dataset regarding your identified problem for the ASEAN Data Science Explorer. Make sure that the dataset includes multiple variables.
2. Load the dataset into pandas dataframe.
3. Prepare the data by applying appropriate data preprocessing techniques.
4. Analyze the data using descriptive analysis.
5. Perform correlation analysis.
6. Interpret the results based on the descriptive and correlation analysis.
7. Submit the PDF file.

```
 1 import pandas as pd
 2 import seaborn as sns
 3
 4 df = pd.read_csv('/content/data/NUTRITION.csv')
 5 print("Shape:\n", df.shape)
 6 print("\n\nColumns\n:", df.columns)
 7 print("\n\nFirst 5 rows:\n", df.head())
 8
 9 df.fillna(df.mean(), inplace=True)
10 print("\n\nMissing values:\n", df.isnull().sum())
```

```
Shape:
 (37977, 83)


Columns
```

```
: Index(['REF_AREA', 'Geographic area', 'INDICATOR', 'Indicator', 'SEX', 'Sex',
       'AGE', 'Current age', 'WEALTH_QUINTILE', 'Wealth Quintile', 'RESIDENCE',
       'Residence', 'MATERNAL_EDU_LVL', 'Mother's Education Level',
       'HEAD_OF_HOUSE', 'Head of House', 'REPORTING_LVL', 'Reporting level',
       'INDICATOR_METADATA', 'UNIT_MULTIPLIER', 'Unit multiplier',
       'UNIT_MEASURE', 'Unit of measure', 'SOURCE_LINK', 'SERIES_FOOTNOTE',
       'CUSTODIAN', 'PUBLICATION_DATE', '1975', '1976', '1977', '1978', '1979',
       '1980', '1981', '1982', '1983', '1984', '1985', '1986', '1987', '1988',
       '1989', '1990', '1991', '1992', '1993', '1994', '1995', '1996', '1997',
       '1998', '1999', '2000', '2000-05-31', '2000-12-31', '2001', '2002',
       '2003', '2004', '2005', '2005-12-07', '2005-12-31', '2006', '2007',
       '2008', '2008-03-31', '2008-11-18', '2009', '2010', '2011', '2012',
       '2013', '2014', '2015', '2016', '2017', '2018', '2019', '2020',
       '2020-01-01', '2020-07-02', '2021', '2022'],
      dtype='object')


First 5 rows:
   REF_AREA    Geographic area       INDICATOR           Indicator SEX     Sex  \
0       BRN  Brunei Darussalam  NT_ANT_BAZ_NE2  BMI-for-age <-2 SD   F  Female
1       BRN  Brunei Darussalam  NT_ANT_BAZ_NE3  BMI-for-age <-3 SD   F  Female
2       BRN  Brunei Darussalam  NT_ANT_BAZ_PO2  BMI-for-age >+2 SD   F  Female
3       BRN  Brunei Darussalam  NT_ANT_BAZ_PO2  BMI-for-age >+2 SD   F  Female
4       BRN  Brunei Darussalam  NT_ANT_BAZ_PO2  BMI-for-age >+2 SD   F  Female

       AGE          Current age WEALTH_QUINTILE Wealth Quintile  ... 2015 2016  \
0    Y0T4    Under 5 years old              _T           Total  ...  NaN  NaN
1    Y0T4    Under 5 years old              _T           Total  ...  NaN  NaN
2    M0T5   Under 6 months old              _T           Total  ...  NaN  NaN
3  M12T23  12 to 23 months old              _T           Total  ...  NaN  NaN
4  M24T35  24 to 35 months old              _T           Total  ...  NaN  NaN

  2017 2018 2019 2020 2020-01-01 2020-07-02 2021  2022
0  NaN  NaN  NaN  NaN        NaN        NaN  NaN   NaN
1  NaN  NaN  NaN  NaN        NaN        NaN  NaN   NaN
2  NaN  NaN  NaN  NaN        NaN        NaN  NaN   NaN
3  NaN  NaN  NaN  NaN        NaN        NaN  NaN   NaN
4  NaN  NaN  NaN  NaN        NaN        NaN  NaN   NaN

[5 rows x 83 columns]


Missing values:
 REF_AREA           0
Geographic area    0
INDICATOR          0
Indicator          0
SEX                0
                  ..
2020               0
2020-01-01         0
```

```
1 print("Description:\n", df.describe())
```

Description:

	UNIT_MULTIPLIER	PUBLICATION_DATE	1975	1976
count	37977.000000	0.0	37977.000000	37977.000000
mean	0.001581	NaN	6.785000	6.790000
std	0.068828	NaN	0.249249	0.247551
min	0.000000	NaN	0.300000	0.300000
25%	0.000000	NaN	6.785000	6.790000
50%	0.000000	NaN	6.785000	6.790000
75%	0.000000	NaN	6.785000	6.790000
max	3.000000	NaN	21.200000	21.200000

	1977	1978	1979	1980	1981
count	37977.000000	37977.000000	37977.000000	37977.000000	37977.00000
mean	6.801667	6.818333	6.833333	6.861667	6.89500
std	0.246141	0.244020	0.242239	0.240638	0.23897
min	0.300000	0.400000	0.400000	0.400000	0.40000
25%	6.801667	6.818333	6.833333	6.861667	6.89500
50%	6.801667	6.818333	6.833333	6.861667	6.89500
75%	6.801667	6.818333	6.833333	6.861667	6.89500
max	21.200000	21.200000	21.200000	21.300000	21.40000

	1982	...	2015	2016	2017
count	37977.000000	...	37977.000000	37977.000000	37977.000000
mean	6.946667	...	13.021137	16.060056	24.213289
std	0.237275	...	49.004443	49.527904	48.432628
min	0.400000	...	-1.900000	-1.800000	-1.800000
25%	6.946667	...	13.021137	16.060056	24.213289
50%	6.946667	...	13.021137	16.060056	24.213289
75%	6.946667	...	13.021137	16.060056	24.213289
max	21.500000	...	7991.800000	7858.900000	7688.000000

	2018	2019	2020	2020-01-01	2020-07-02
count	37977.000000	37977.000000	37977.000000	37977.00000	37977.000000
mean	65.459660	17.465729	310.399414	16.23125	15.550000
std	46.084929	46.314807	42.940982	0.10287	0.028665
min	-1.600000	-1.500000	0.900000	6.20000	11.600000
25%	65.459660	17.465729	310.399414	16.23125	15.550000
50%	65.459660	17.465729	310.399414	16.23125	15.550000
75%	65.459660	17.465729	310.399414	16.23125	15.550000
max	7517.300000	7363.900000	7212.900000	25.40000	19.500000

	2021	2022
count	37977.000000	37977.000000
mean	13.955687	389.702041
std	43.808971	40.466928
min	-1.400000	0.800000
25%	13.955687	389.702041
50%	13.955687	389.702041
75%	13.955687	389.702041
max	7048.700000	6896.600000

[8 rows x 58 columns]

```
1 start = df.columns.get_loc('1975')
2 end = df.columns.get_loc('2016')
3
4 df = pd.concat([df.iloc[:, :start], df.iloc[:, end:]], axis=1)
5 print(df.head())
6 print(df.columns)
```

```
  REF_AREA    Geographic area       INDICATOR           Indicator SEX     Sex  \
0      BRN  Brunei Darussalam  NT_ANT_BAZ_NE2  BMI-for-age <-2 SD   F  Female
1      BRN  Brunei Darussalam  NT_ANT_BAZ_NE3  BMI-for-age <-3 SD   F  Female
2      BRN  Brunei Darussalam  NT_ANT_BAZ_PO2  BMI-for-age >+2 SD   F  Female
3      BRN  Brunei Darussalam  NT_ANT_BAZ_PO2  BMI-for-age >+2 SD   F  Female
4      BRN  Brunei Darussalam  NT_ANT_BAZ_PO2  BMI-for-age >+2 SD   F  Female

       AGE          Current age WEALTH_QUINTILE Wealth Quintile  ...  \
0    Y0T4    Under 5 years old              _T           Total  ...
1    Y0T4    Under 5 years old              _T           Total  ...
2    M0T5   Under 6 months old              _T           Total  ...
3  M12T23  12 to 23 months old              _T           Total  ...
4  M24T35  24 to 35 months old              _T           Total  ...

  PUBLICATION_DATE       2016       2017      2018       2019        2020  \
0              NaN  16.060056  24.213289  65.45966  17.465729  310.399414
1              NaN  16.060056  24.213289  65.45966  17.465729  310.399414
2              NaN  16.060056  24.213289  65.45966  17.465729  310.399414
3              NaN  16.060056  24.213289  65.45966  17.465729  310.399414
4              NaN  16.060056  24.213289  65.45966  17.465729  310.399414

  2020-01-01 2020-07-02       2021        2022
0   16.23125      15.55  13.955687  389.702041
1   16.23125      15.55  13.955687  389.702041
2   16.23125      15.55  13.955687  389.702041
3   16.23125      15.55  13.955687  389.702041
4   16.23125      15.55  13.955687  389.702041

[5 rows x 36 columns]
Index(['REF_AREA', 'Geographic area', 'INDICATOR', 'Indicator', 'SEX', 'Sex',
       'AGE', 'Current age', 'WEALTH_QUINTILE', 'Wealth Quintile', 'RESIDENCE',
       'Residence', 'MATERNAL_EDU_LVL', 'Mother's Education Level',
       'HEAD_OF_HOUSE', 'Head of House', 'REPORTING_LVL', 'Reporting level',
       'INDICATOR_METADATA', 'UNIT_MULTIPLIER', 'Unit multiplier',
       'UNIT_MEASURE', 'Unit of measure', 'SOURCE_LINK', 'SERIES_FOOTNOTE',
       'CUSTODIAN', 'PUBLICATION_DATE', '2016', '2017', '2018', '2019', '2020',
       '2020-01-01', '2020-07-02', '2021', '2022'],
      dtype='object')
```

```
1 df['Geographic area'].unique()
```

```
array(['Brunei Darussalam', 'Indonesia', 'Cambodia',
       "Lao People's Democratic Republic", 'Myanmar', 'Malaysia',
       'Philippines', 'Singapore', 'Thailand', 'Viet Nam'], dtype=object)
```

```
1 corr_matrix = df.corr()
```

```
<ipython-input-33-b69b4b6a6184>:1: FutureWarning: The default value of numeric_only in
  corr_matrix = df.corr()
```

```
import matplotlib.pyplot as plt
plt.figure(figsize=(10, 8))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title('Correlation Matrix')
plt.show()
```