Hands-on Activity 10.1 Data Analysis using Python

Intended Learning Outcomes:

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

Resources:

- Personal Computer
- Jupyter Notebook
- Internet Connection

Instructions:

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.

Dataset and Problem: Wastewater Sanitation

Source:

https://ourworldindata.org/sdgs/clean-water-sanitation

```
import pandas as pd
import numpy as np


# read csv file

water = pd.read_csv('wastewater safely treated.csv')
water
```

	Entity	Code	Year	6.3.1 - Proportion of safely treated domestic wastewater flows (%) - EN_WWT_WWDS
0	Algeria	DZA	2020	76.17
1	Algeria	DZA	2022	76.19
2	American Samoa	ASM	2020	69.01
3	American Samoa	ASM	2022	77.50
4	Andorra	AND	2020	100.00
...	...	...	...	...
285	World	OWID_WRL	2022	57.79
286	Yemen	YEM	2020	34.40
287	Yemen	YEM	2022	28.11
288	Zimbabwe	ZWE	2020	22.99
289	Zimbabwe	ZWE	2022	54.78

Next steps: Generate code with water   View recommended plots

```
# create new variable countries and select only the ASEAN countries from the list

countries =  ['Vietnam', 'Indonesia', 'Philippines', 'Thailand', 'Myanmar', 'Cambodia', 'Malaysia', 'Lao PDR', 'Singapore', 'Brunei Darussal
water = water[water['Entity'].isin(countries)]
```

```
# check the entity values

water['Entity'].unique()
```

```
array(['Cambodia', 'Malaysia', 'Myanmar', 'Philippines', 'Singapore',
       'Thailand', 'Vietnam'], dtype=object)
```

```
# info of dataframe

water.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 11 entries, 38 to 283
Data columns (total 4 columns):
 #   Column                                                                Non-Null Count  Dtype  
---  ------                                                                --------------  -----  
 0   Entity                                                                11 non-null     object 
 1   Code                                                                  11 non-null     object 
 2   Year                                                                  11 non-null     int64  
 3   6.3.1 - Proportion of safely treated domestic wastewater flows (%) - EN_WWT_WWDS  11 non-null     float64
dtypes: float64(1), int64(1), object(2)
memory usage: 440.0+ bytes
```

```
# descriptive statistics

water.describe()
```

	Year	6.3.1 - Proportion of safely treated domestic wastewater flows (%) - EN_WWT_WWDS
count	11.000000	11.000000
mean	2021.272727	57.956364
std	1.009050	31.989907
min	2020.000000	15.120000
25%	2020.000000	32.265000
50%	2022.000000	46.790000
75%	2022.000000	88.570000
max	2022.000000	100.000000

```
# check missing values

water.isnull().sum()
```

```
Entity                                                                     0
Code                                                                       0
Year                                                                       0
6.3.1 - Proportion of safely treated domestic wastewater flows (%) - EN_WWT_WWDS    0
dtype: int64
```

```
# change the name of column Entity to Country

water.rename(columns={'Entity': 'Country'}, inplace=True)
water.head()
```

```
<ipython-input-95-552ee7ece125>:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_
  water.rename(columns={'Entity': 'Country'}, inplace=True)
```

	Country	Code	Year	6.3.1 - Proportion of safely treated domestic wastewater flows (%) - EN_WWT_WWDS
38	Cambodia	KHM	2022	46.79
164	Malaysia	MYS	2020	87.82
165	Malaysia	MYS	2022	89.32
183	Myanmar	MMR	2022	15.12
210	Philippines	PHL	2020	42.95

Next steps: Generate code with water   View recommended plots

```
# change the name of column 6.3.1 - Proportion of safely treated domestic wastewater flows (%) - EN_WWT_WWDS to Proportion of safely wastewa

water.rename(columns={'6.3.1 - Proportion of safely treated domestic wastewater flows (%) - EN_WWT_WWDS': 'Proportion of safely wastewater f
water.head()
```

```
<ipython-input-96-db3490c247e9>:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_
  water.rename(columns={'6.3.1 - Proportion of safely treated domestic wastewater flows
```

	Country	Code	Year	Proportion of safely wastewater flows
38	Cambodia	KHM	2022	46.79
164	Malaysia	MYS	2020	87.82
165	Malaysia	MYS	2022	89.32
183	Myanmar	MMR	2022	15.12
210	Philippines	PHL	2020	42.95

Next steps: Generate code with water   View recommended plots

```
# mean of proportion

water['Proportion of safely wastewater flows'].mean()
```

```
57.95636363636363
```

```
# check code values

water['Code'].unique()
```

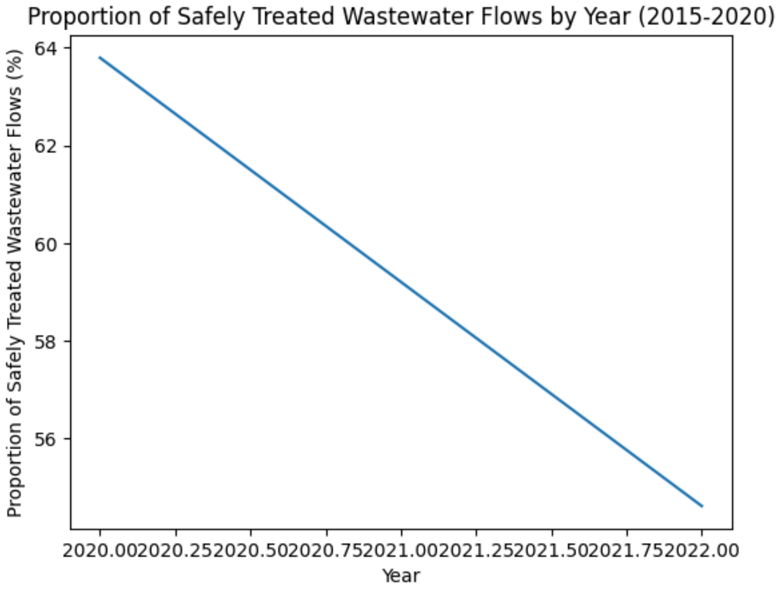
```
array(['KHM', 'MYS', 'MMR', 'PHL', 'SGP', 'THA', 'VNM'], dtype=object)
```

```
# check year values

water['Year'].unique()
```

```
array([2022, 2020])
```

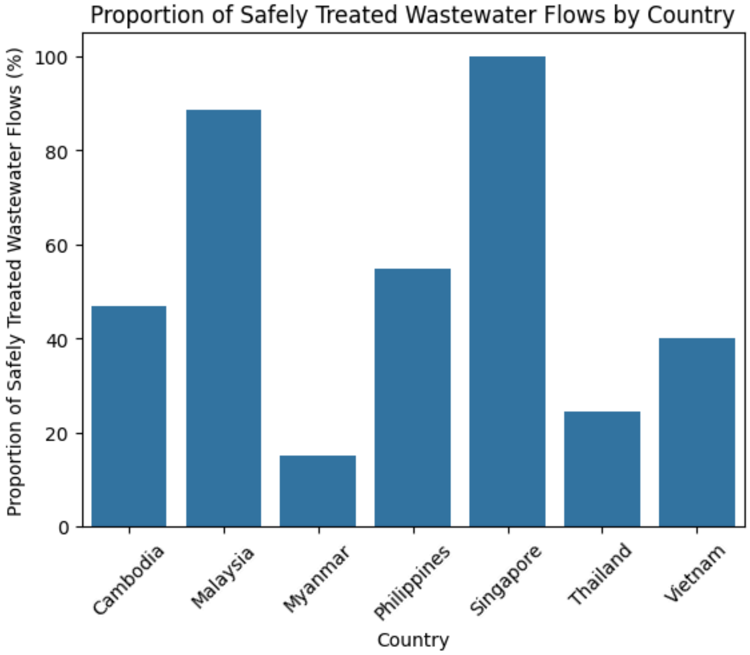
```
# create a lineplot for proportion by year

import matplotlib.pyplot as plt
import seaborn as sns

grouped_data = water.groupby('Year')['Proportion of safely wastewater flows'].mean().reset_index()

sns.lineplot(x='Year', y='Proportion of safely wastewater flows', data=grouped_data)

plt.title('Proportion of Safely Treated Wastewater Flows by Year (2015-2020)')
plt.xlabel('Year')
plt.ylabel('Proportion of Safely Treated Wastewater Flows (%)')
```

```
Text(0, 0.5, 'Proportion of Safely Treated Wastewater Flows (%)')
```



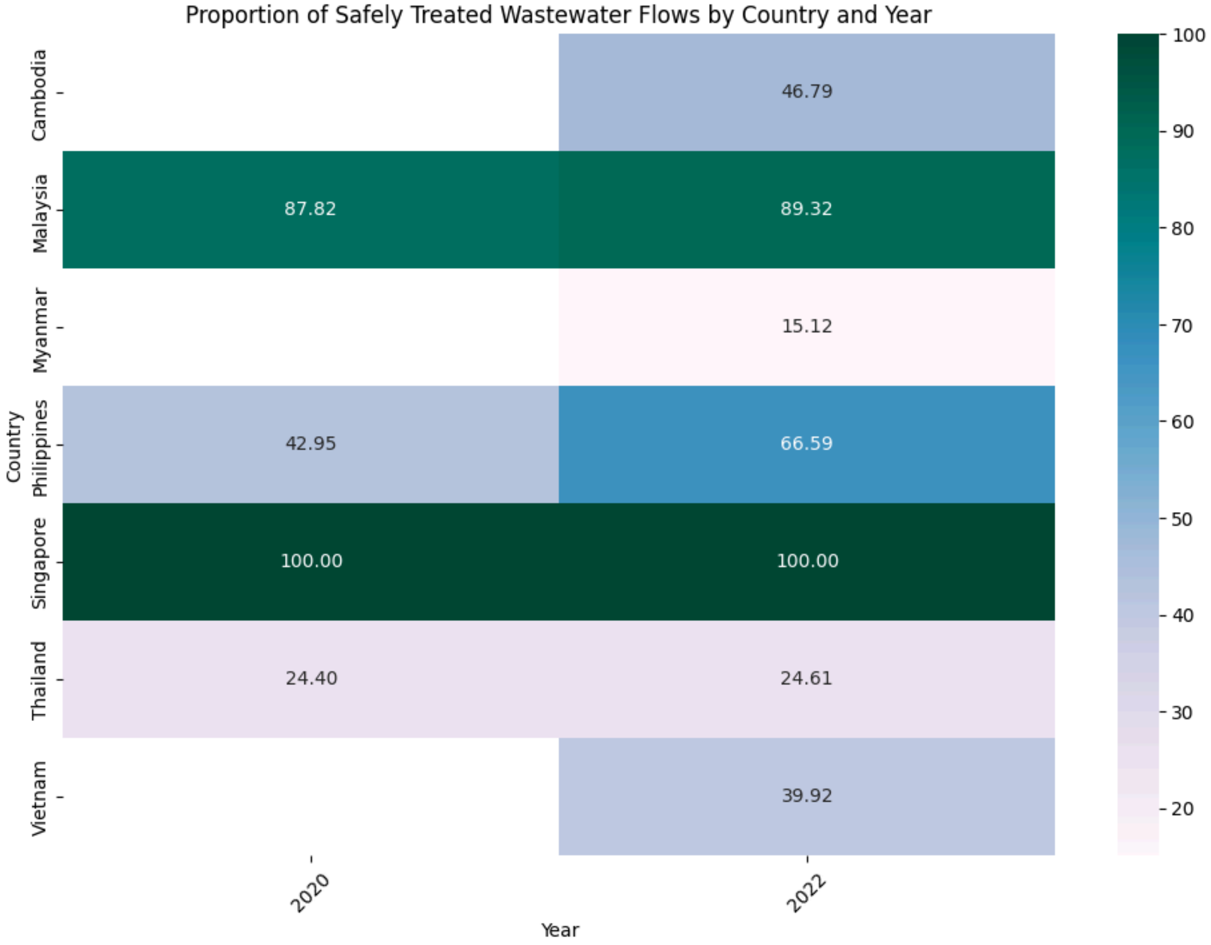
```
# create a barplot for proportion by country

import matplotlib.pyplot as plt
import seaborn as sns

grouped_data = water.groupby('Country')['Proportion of safely wastewater flows'].mean().reset_index()

sns.barplot(x='Country', y='Proportion of safely wastewater flows', data=grouped_data)

plt.title('Proportion of Safely Treated Wastewater Flows by Country')
plt.xlabel('Country')
plt.ylabel('Proportion of Safely Treated Wastewater Flows (%)')

plt.xticks(rotation=45)
```

```
([0, 1, 2, 3, 4, 5, 6],
 [Text(0, 0, 'Cambodia'),
  Text(1, 0, 'Malaysia'),
  Text(2, 0, 'Myanmar'),
  Text(3, 0, 'Philippines'),
  Text(4, 0, 'Singapore'),
  Text(5, 0, 'Thailand'),
  Text(6, 0, 'Vietnam')])
```



```
# visualize a heatmap for proportion by country and year

import matplotlib.pyplot as plt
import seaborn as sns

proportion_data = water.pivot_table(values='Proportion of safely wastewater flows', index='Country', columns='Year')

plt.figure(figsize=(12, 8))
sns.heatmap(proportion_data, cmap='PuBuGn', annot=True, fmt=".2f")

plt.title('Proportion of Safely Treated Wastewater Flows by Country and Year')
plt.xlabel('Year')
plt.ylabel('Country')

plt.xticks(rotation=45)
```

```
(array([0.5, 1.5]), [Text(0.5, 0, '2020'), Text(1.5, 0, '2022')])
```


Proportion of Safely Treated Wastewater Flows by Country and Year
Cambodia
Malaysia
Myanmar
Philippines
Singapore
Thailand
Vietnam
Country
Year