Skills Network

Hands-on Lab: Guided Practice Project

Estimated Effort: 60 mins

Project Scenario

You have been employed as a data analyst by a Healthcare consultancy firm which has been conducting a survey on the state of global happiness annually. The World Happiness Report offers valuable insights into factors influencing happiness across countries. The firm wants you to produce a report to find out whether there are demographic, regional, and/or economic characteristics that lead to a better life.

The project tasks are data preparation, analysis, visualization, and dashboarding. Based on the data set, you must write prompts to generate the Python codes for performing specific tasks. You can access a JupyterLite-based testing environment to test the generated codes using the Generative AI classroom prompts.

The tasks assigned to you are as follows:

1. Check the correctness of the data types in the dataset.

2. There might be a few missing values in the dataset. Data cleaning will be a part of the assignment.

3. You have to perform exploratory data analysis to draw insights on the data:

    - Identify the GDP per capita and Healthy Life Expectancy of the top 10 countries and represent it as a bar chart
    - Find the correlation between the Economy (GDP per Capita), Family, Health (Life Expectancy), Freedom, Trust (Government Corruption), Generosity, and Happiness Score
    - Create a scatter plot to identify the effect of GDP per Capita on Happiness Score in various Regions
    - Create a pie chart to present Happiness Score by region
    - Create a map to display GDP per capita of countries and include Healthy Life Expectancy to be shown as a tooltip

4. Create a dashboard with at least four of the above visualizations

5. Generate the narrative to present the dashboard

You decide to use Generative AI to create python codes that can help you analyze the data, determine the best features and create the visualization as per requirement.

Disclaimer: This is a fictitious scenario created for the purpose of this project. The dataset being used is publicly available.

About the data set

The World Happiness Report is a landmark survey of the state of global happiness. The reports review the state of happiness in the world today and show how the new science of happiness explains personal and national variations in happiness. This is a public dataset available on the Kaggle website as World Happiness Report under the CC0: Public Domain license.

For this guided practice project we will work on the **year 2016** data, which has been slightly modified for the purpose of this guided practice project.

You can download it from here: 2016

Attributes of this dataset have been explained below.

Variable	Description
Country	Name of the country
Region	Region the country belongs to
Happiness Rank	Rank of the country based on the Happiness Score
Happiness Score	A metric measured in 2016 by asking the sampled people the question: "How would you rate your happiness?"
Lower Confidence Interval	Lower Confidence Interval of the Happiness Score
Upper Confidence Interval	Upper Confidence Interval of the Happiness Score
Economy (GDP per Capita)	The extent to which GDP contributes to the calculation of the Happiness Score
Family	The extent to which Family contributes to the calculation of the Happiness Score
Health (Life Expectancy)	The extent to which Life expectancy contributes to the calculation of the Happiness Score
Freedom	The extent to which Freedom contributes to the calculation of the Happiness Score
Trust (Government Corruption)	The extent to which Trust contributes to the calculation of the Happiness Score
Generosity	The extent to which Generosity contributes to the calculation of the Happiness Score
Dystopia Residual	Dystopia is an imaginary country that has the world's least-happy people. The residuals, or unexplained components, differ for each country, reflecting the extent to which the six variables either over- or under-explain average 2014-2016 life evaluations. These residuals have an average value of approximately zero over the whole set of countries

Code execution environment

To test the prompt-generated code, open the Jupyter Notebook that you have been provided with in the succeeding lab in the course.

The data set for this lab is available in the following URL.

```
URL = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMSkillsNetwork-AI0272EN-SkillsNetwork/labs/dataset/2016.csv"
```

Complete the setup in the Jupyter Notebook and then proceed further.

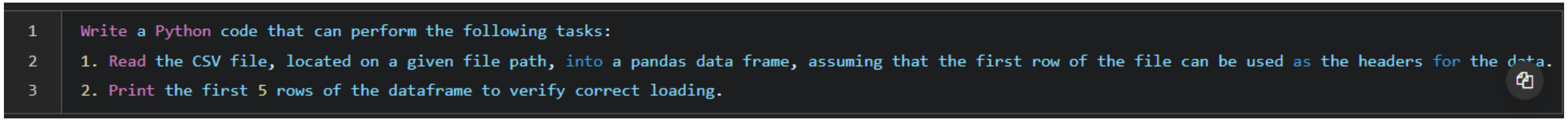
> *Important Note: All prompts that are made available have been hidden and the users are encouraged to first try to write their own prompts to create the solutions. Also, the prompts given as solutions have also been maintained as ones which will create generic code structures which you can modify according to the question at hand.*

Importing the Dataset

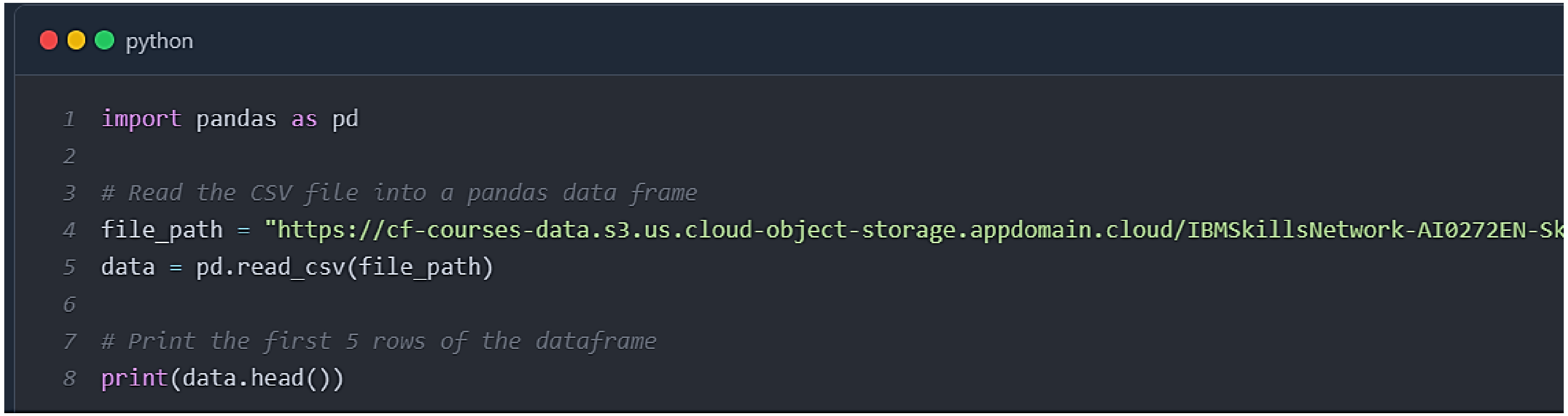
You can begin by using the Generative AI model to create a python script that can load the dataset to a pandas dataframe. The dataset file already has the headers in the first row.

> NOTE: Write the prompt to generate the said code and test it in the JupyterLite environment. For verification of appropriate loading, include a step for printing the first 5 values of the loaded dataframe.

▶ Click here for the prompt



```
Write a Python code that can perform the following tasks:
1. Read the CSV file, located on a given file path, into a pandas data frame, assuming that the first row of the file can be used as the headers for the data.
2. Print the first 5 rows of the dataframe to verify correct loading.
```



```python
import pandas as pd

# Read the CSV file into a pandas data frame
file_path = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMSkillsNetwork-AI0272EN-Sk
data = pd.read_csv(file_path)

# Print the first 5 rows of the dataframe
print(data.head())
```

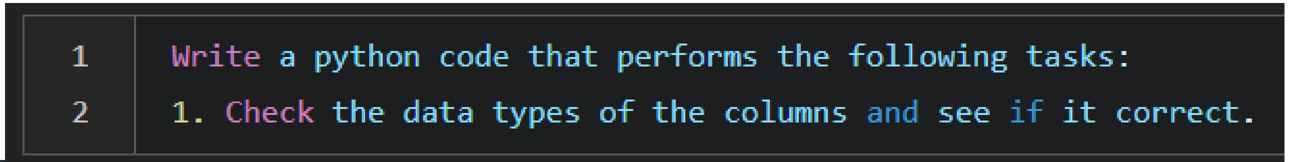
Data Preparation

Data Cleaning - Checking for correct data types

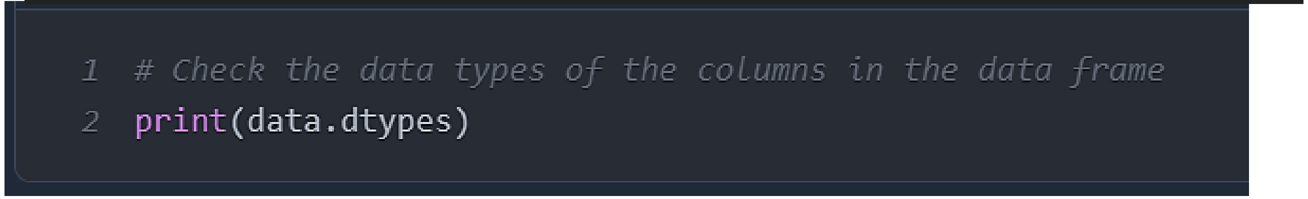
You need to check the columns in the dataset for the right data type. If the data type is not appropriate, it may lead to misinterpretation of the data. Write a prompt that performs the following tasks:

1. List the data types of the columns and check if there is any column type that is unsuitable.

▶ Click here for the prompt



```
Write a python code that performs the following tasks:
1. Check the data types of the columns and see if it correct.
```



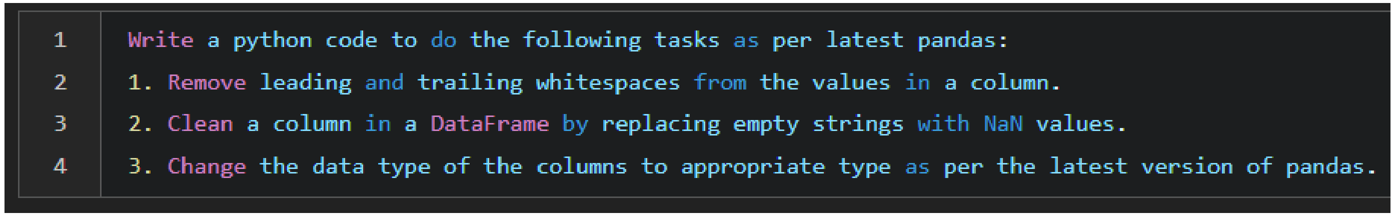
```
# Check the data types of the columns in the data frame
print(data.dtypes)
```

Data Cleaning - Change the data types

1. Change the data type to an appropriate type.

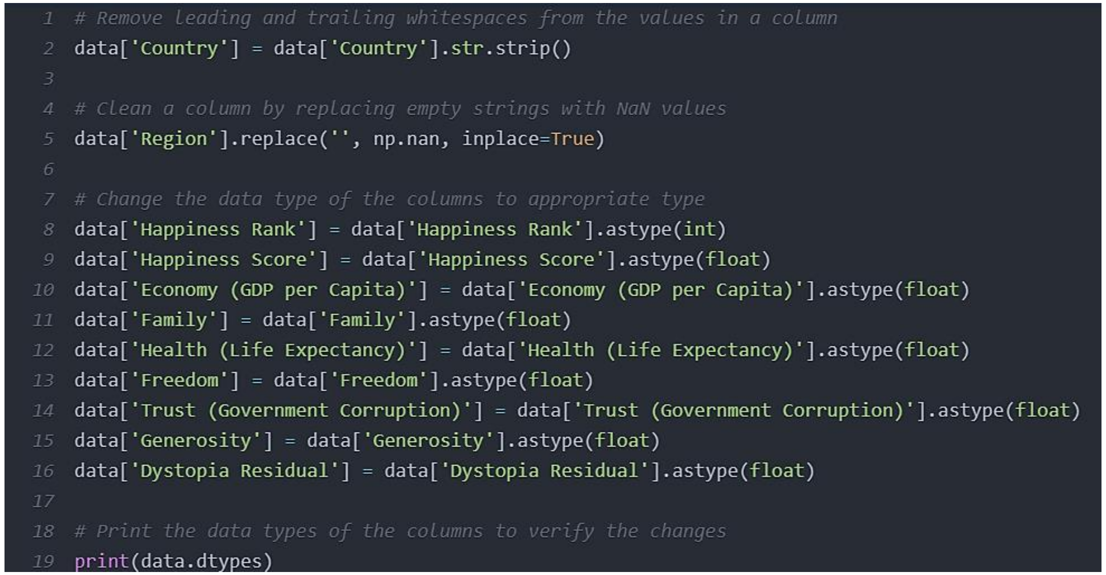
   If you need specific types, you can specify so.

▶ Click here for the prompt



```
Write a python code to do the following tasks as per latest pandas:
1. Remove leading and trailing whitespaces from the values in a column.
2. Clean a column in a DataFrame by replacing empty strings with NaN values.
3. Change the data type of the columns to appropriate type as per the latest version of pandas.
```

Please note as of the future version 3 of pandas it is recommended to use `df.method({col: value}, inplace=True)` instead of `df[col].method(value, inplace=True)`. You may see warning message in this regard, when you generate code and execute it in the notebook.



```python
# Remove leading and trailing whitespaces from the values in a column
data['Country'] = data['Country'].str.strip()

# Clean a column by replacing empty strings with NaN values
data['Region'].replace('', np.nan, inplace=True)

# Change the data type of the columns to appropriate type
data['Happiness Rank'] = data['Happiness Rank'].astype(int)
data['Happiness Score'] = data['Happiness Score'].astype(float)
data['Economy (GDP per Capita)'] = data['Economy (GDP per Capita)'].astype(float)
data['Family'] = data['Family'].astype(float)
data['Health (Life Expectancy)'] = data['Health (Life Expectancy)'].astype(float)
data['Freedom'] = data['Freedom'].astype(float)
data['Trust (Government Corruption)'] = data['Trust (Government Corruption)'].astype(float)
data['Generosity'] = data['Generosity'].astype(float)
data['Dystopia Residual'] = data['Dystopia Residual'].astype(float)

# Print the data types of the columns to verify the changes
print(data.dtypes)
```

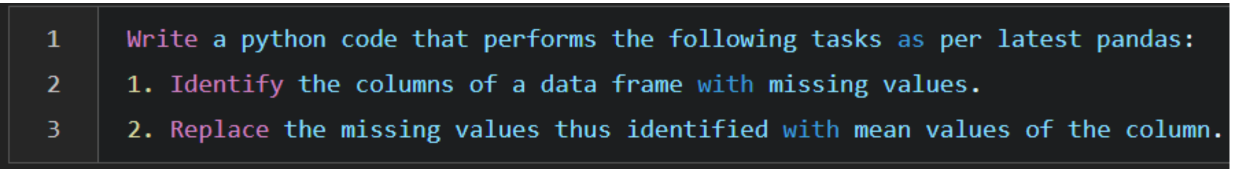
Data Cleaning - Checking for missing values

At this stage, you need to clean up the data. As has been shared earlier, the data may have missing values. Write a prompt that performs the following tasks:

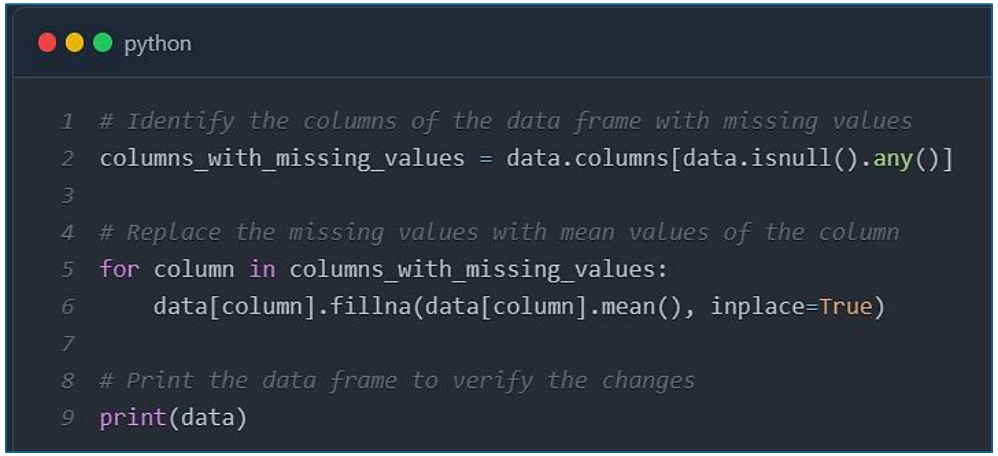
1. Identify the columns with missing values and fill the blank cells with mean value of the columns.

   Please ensure that you have changed all the column to the appropriate data type before you do this.

▶ Click here for the prompt



```
Write a python code that performs the following tasks as per latest pandas:
1. Identify the columns of a data frame with missing values.
2. Replace the missing values thus identified with mean values of the column.
```



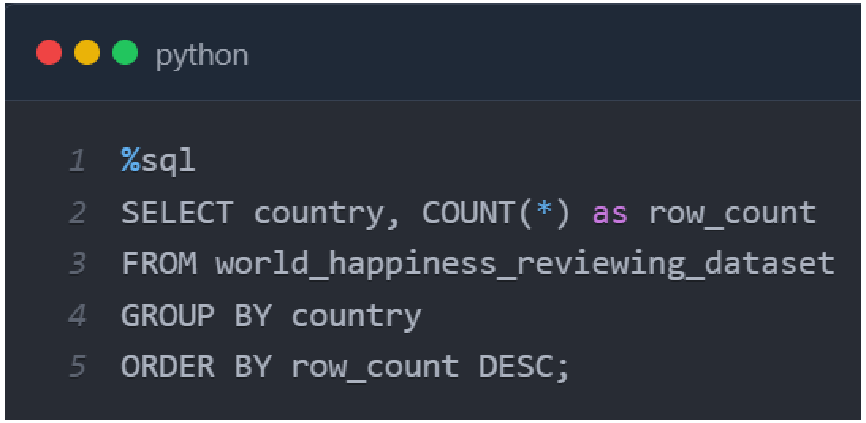
python

```python
# Identify the columns of the data frame with missing values
columns_with_missing_values = data.columns[data.isnull().any()]

# Replace the missing values with mean values of the column
for column in columns_with_missing_values:
    data[column].fillna(data[column].mean(), inplace=True)

# Print the data frame to verify the changes
print(data)
```

Database Querying (optional)

You may choose to query the dataset using dbsensei. Use “Text to SQL“ option for generating SQL queries.

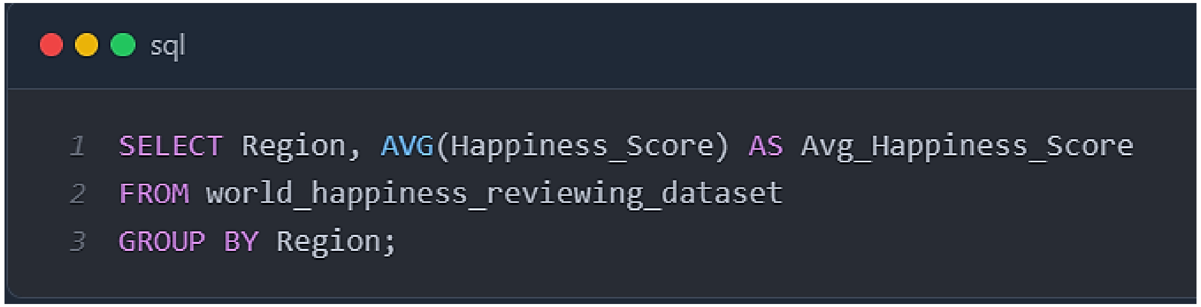
1. Generate SQL query to Count the number of rows for each country



```
%sql
SELECT country, COUNT(*) as row_count
FROM world_happiness_reviewing_dataset
GROUP BY country
ORDER BY row_count DESC;
```

2. Generate SQL query to Calculate AVG `Happiness score`, `Region` wise for comparison

   Please note the dbsensei application portal is still in beta. It may be down and unavailable sometimes.



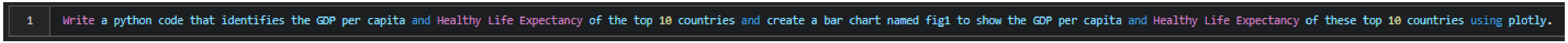
```
SELECT Region, AVG(Happiness_Score) AS Avg_Happiness_Score
FROM world_happiness_reviewing_dataset
GROUP BY Region;
```

Data Insights and Visualization

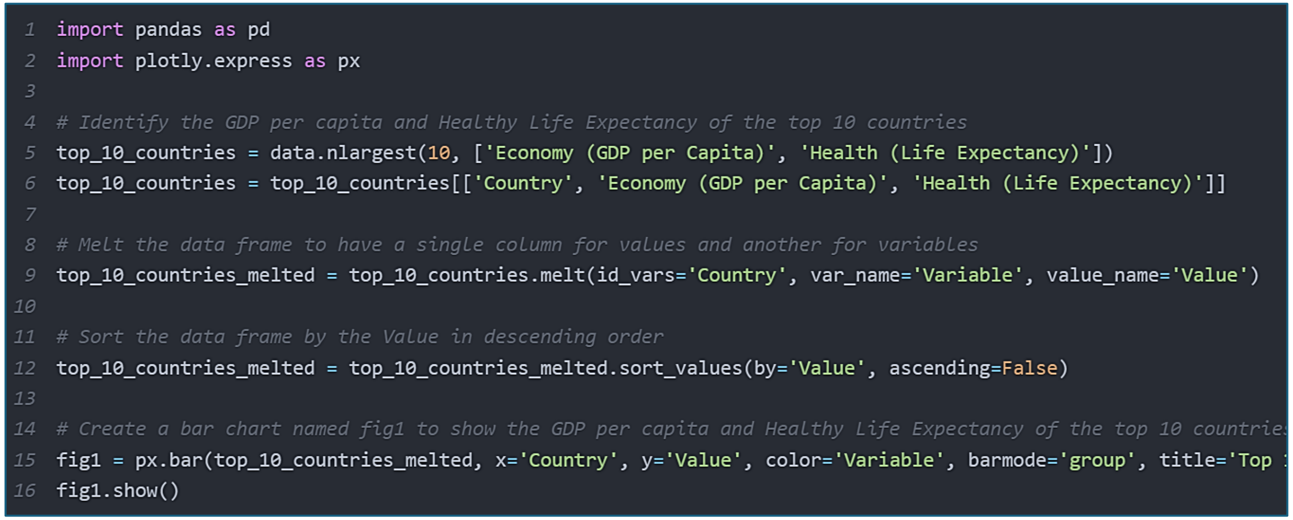
Write prompts that generate codes to perform the following actions:

1. Identify the GDP per capita and Healthy Life Expectancy of the top 10 countries.

▶ Click here for the prompt



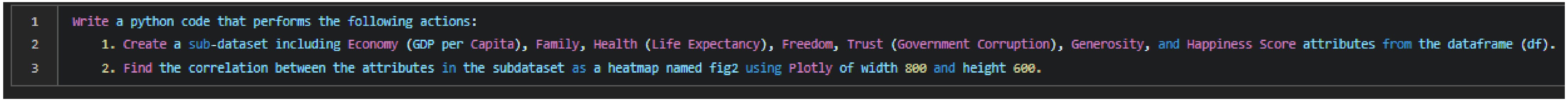
```
Write a python code that identifies the GDP per capita and Healthy Life Expectancy of the top 10 countries and create a bar chart named fig1 to show the GDP per capita and Healthy Life Expectancy of these top 10 countries using plotly.
```



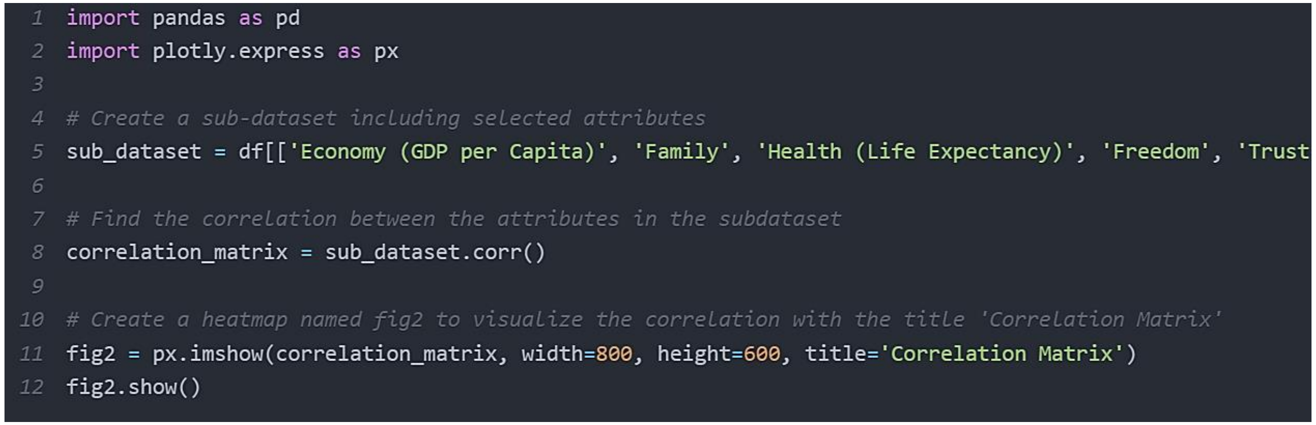
```python
import pandas as pd
import plotly.express as px

# Identify the GDP per capita and Healthy Life Expectancy of the top 10 countries
top_10_countries = data.nlargest(10, ['Economy (GDP per Capita)', 'Health (Life Expectancy)'])
top_10_countries = top_10_countries[['Country', 'Economy (GDP per Capita)', 'Health (Life Expectancy)']]

# Melt the data frame to have a single column for values and another for variables
top_10_countries_melted = top_10_countries.melt(id_vars='Country', var_name='Variable', value_name='Value')

# Sort the data frame by the Value in descending order
top_10_countries_melted = top_10_countries_melted.sort_values(by='Value', ascending=False)

# Create a bar chart named fig1 to show the GDP per capita and Healthy Life Expectancy of the top 10 countrie
fig1 = px.bar(top_10_countries_melted, x='Country', y='Value', color='Variable', barmode='group', title='Top 
fig1.show()
```

2. Find the correlation between the `Economy (GDP per Capita)`, `Family`, `Health (Life Expectancy)`, `Freedom`, `Trust (Government Corruption)`, `Generosity` and `Happiness score`. You may like to represent the correlation as a heatmap of a readable, visually appealing size.

▶ Click here for the prompt



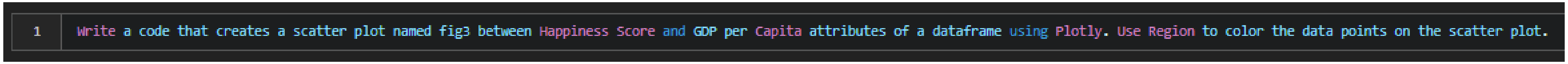
```
Write a python code that performs the following actions:
    1. Create a sub-dataset including Economy (GDP per Capita), Family, Health (Life Expectancy), Freedom, Trust (Government Corruption), Generosity, and Happiness Score attributes from the dataframe (df).
    2. Find the correlation between the attributes in the subdataset as a heatmap named fig2 using Plotly of width 800 and height 600.
```



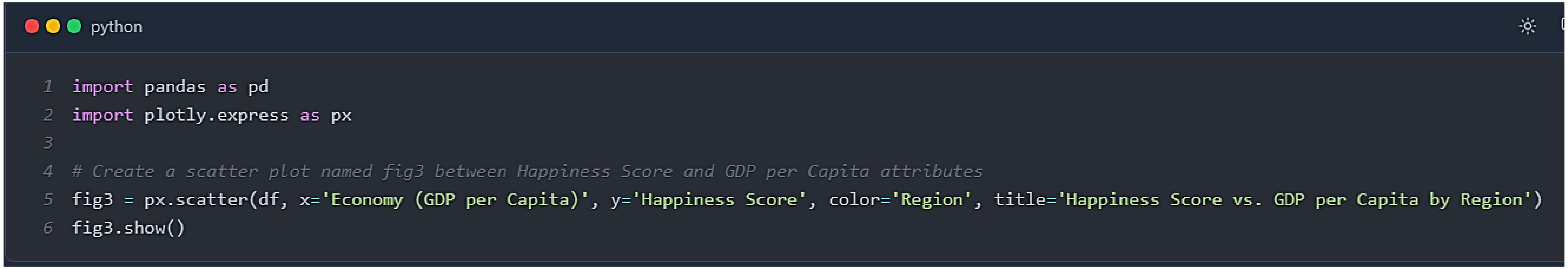
```python
import pandas as pd
import plotly.express as px

# Create a sub-dataset including selected attributes
sub_dataset = df[['Economy (GDP per Capita)', 'Family', 'Health (Life Expectancy)', 'Freedom', 'Trust

# Find the correlation between the attributes in the subdataset
correlation_matrix = sub_dataset.corr()

# Create a heatmap named fig2 to visualize the correlation with the title 'Correlation Matrix'
fig2 = px.imshow(correlation_matrix, width=800, height=600, title='Correlation Matrix')
fig2.show()
```

3. Create a scatter plot to identify the effect of `GDP per Capita` on `Happiness Score` in various `Regions`. Use plotly for creating the plot.

▶ Click here for the prompt



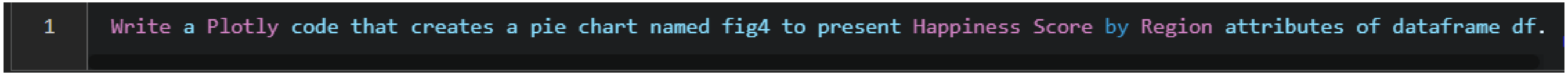
```
Write a code that creates a scatter plot named fig3 between Happiness Score and GDP per Capita attributes of a dataframe using Plotly. Use Region to color the data points on the scatter plot.
```



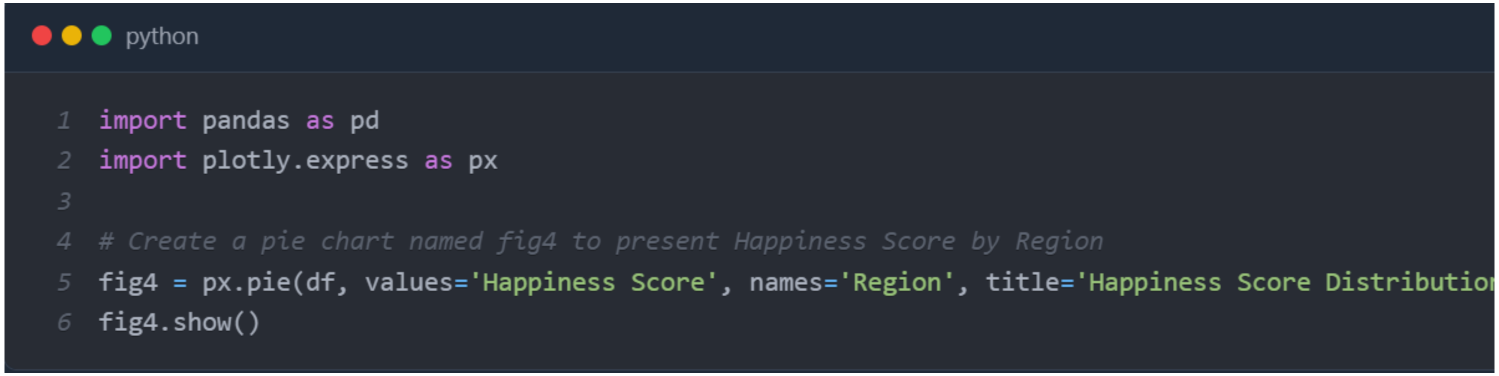
```python
import pandas as pd
import plotly.express as px

# Create a scatter plot named fig3 between Happiness Score and GDP per Capita attributes
fig3 = px.scatter(df, x='Economy (GDP per Capita)', y='Happiness Score', color='Region', title='Happiness Score vs. GDP per Capita by Region')
fig3.show()
```

4. Create a pie chart to present `Happiness Score` by `Regions`

▶ Click here for the prompt



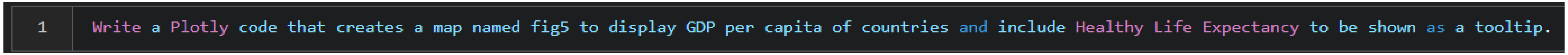
```
Write a Plotly code that creates a pie chart named fig4 to present Happiness Score by Region attributes of dataframe df.
```



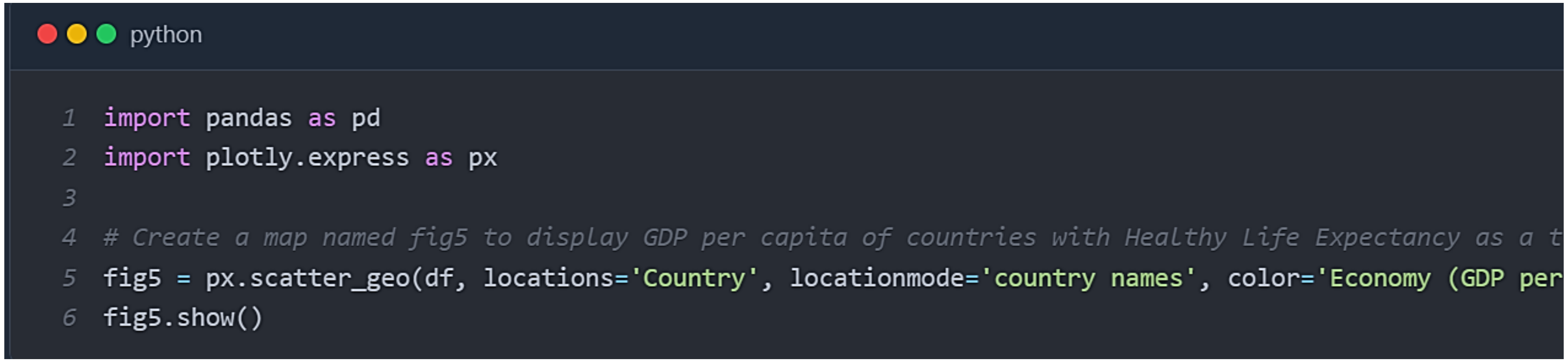
```python
import pandas as pd
import plotly.express as px

# Create a pie chart named fig4 to present Happiness Score by Region
fig4 = px.pie(df, values='Happiness Score', names='Region', title='Happiness Score Distributio
fig4.show()
```

5. Create a map to display `GDP per capita` of `countries` and include `Healthy life expectancy` to be shown as a tooltip

▶ Click here for the prompt



```
Write a Plotly code that creates a map named fig5 to display GDP per capita of countries and include Healthy Life Expectancy to be shown as a tooltip.
```



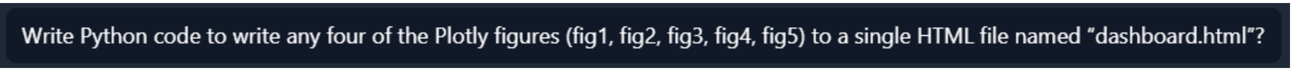
```python
import pandas as pd
import plotly.express as px

# Create a map named fig5 to display GDP per capita of countries with Healthy Life Expectancy as a t
fig5 = px.scatter_geo(df, locations='Country', locationmode='country names', color='Economy (GDP per
fig5.show()
```

Dashboarding and Storytelling

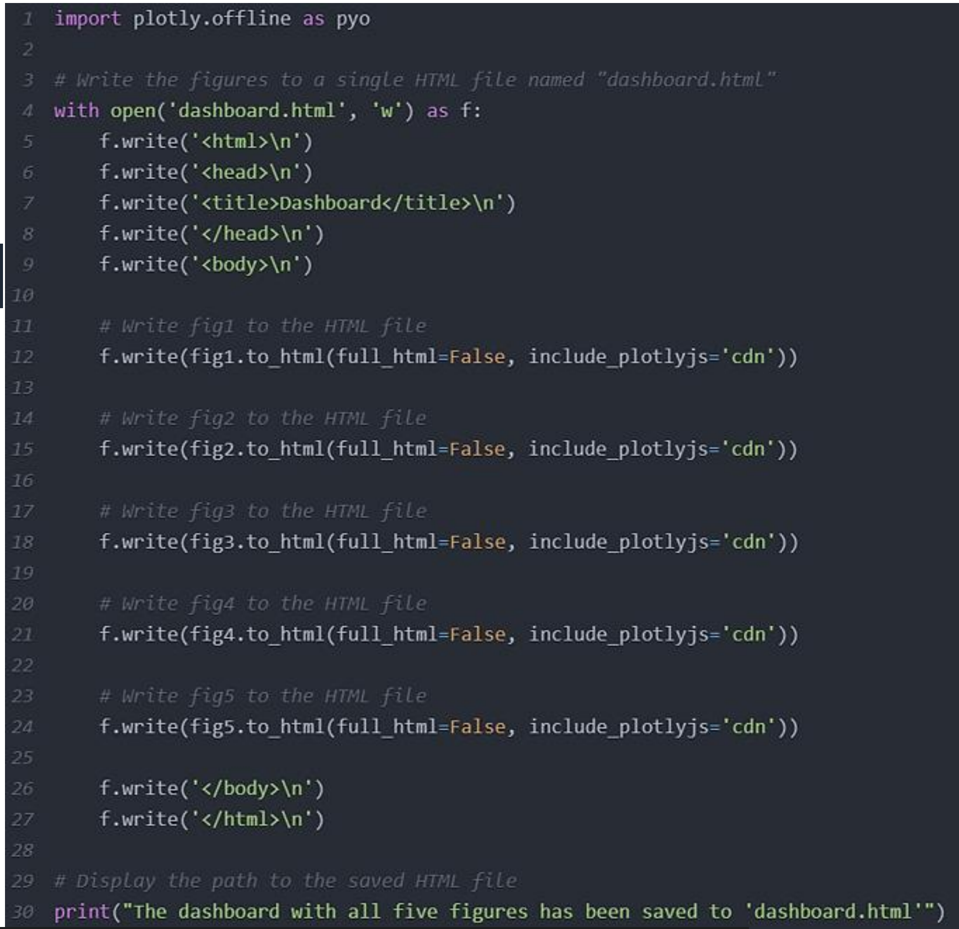
1. Write prompts that generate codes to write at least four of the graph plots generated in the previous steps into a HTML page.

▶ Click here for the prompt

NOTE: Please ensure that the code has the plotly codes generated for creating various visualizations in the previous section.



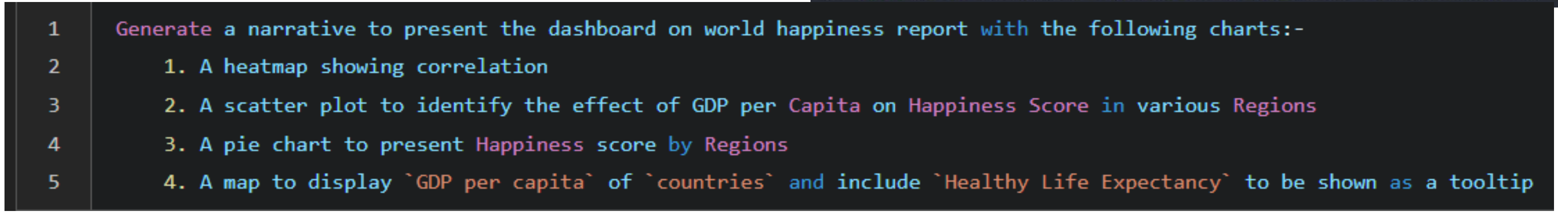
Write Python code to write any four of the Plotly figures (fig1, fig2, fig3, fig4, fig5) to a single HTML file named "dashboard.html"?



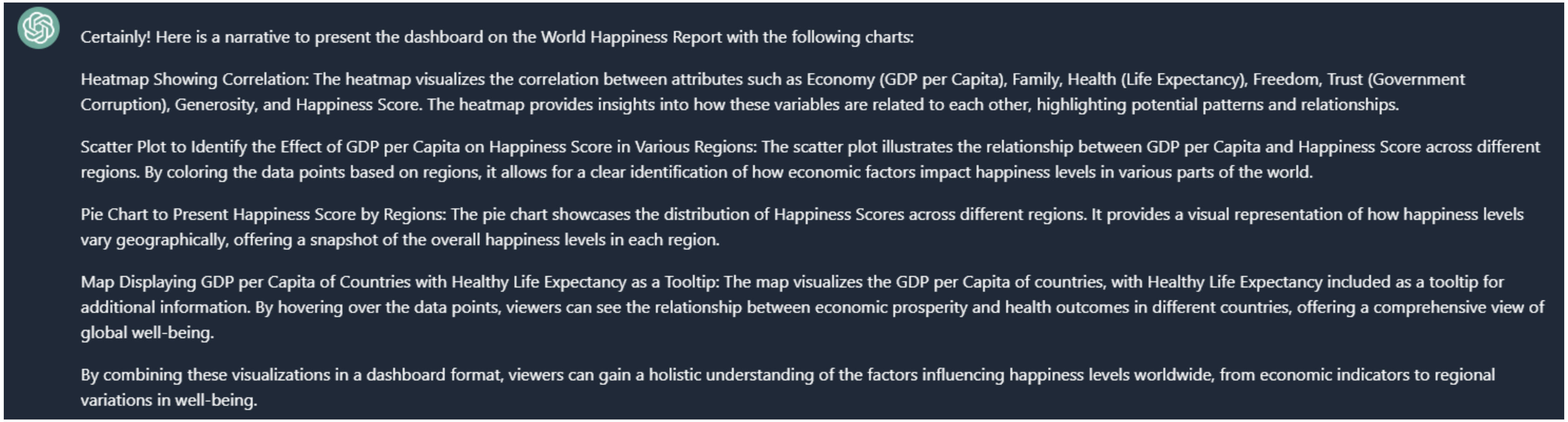
```python
import plotly.offline as pyo

# Write the figures to a single HTML file named "dashboard.html"
with open('dashboard.html', 'w') as f:
    f.write('<html>\n')
    f.write('<head>\n')
    f.write('<title>Dashboard</title>\n')
    f.write('</head>\n')
    f.write('<body>\n')

    # Write fig1 to the HTML file
    f.write(fig1.to_html(full_html=False, include_plotlyjs='cdn'))

    # Write fig2 to the HTML file
    f.write(fig2.to_html(full_html=False, include_plotlyjs='cdn'))

    # Write fig3 to the HTML file
    f.write(fig3.to_html(full_html=False, include_plotlyjs='cdn'))

    # Write fig4 to the HTML file
    f.write(fig4.to_html(full_html=False, include_plotlyjs='cdn'))

    # Write fig5 to the HTML file
    f.write(fig5.to_html(full_html=False, include_plotlyjs='cdn'))

    f.write('</body>\n')
    f.write('</html>\n')

# Display the path to the saved HTML file
print("The dashboard with all five figures has been saved to 'dashboard.html'")
```

2. Generate the narrative to present the dashboard

▶ Click here for the prompt



```
Generate a narrative to present the dashboard on world happiness report with the following charts:-
    1. A heatmap showing correlation
    2. A scatter plot to identify the effect of GDP per Capita on Happiness Score in various Regions
    3. A pie chart to present Happiness score by Regions
    4. A map to display `GDP per capita` of `countries` and include `Healthy Life Expectancy` to be shown as a tooltip
```



Certainly! Here is a narrative to present the dashboard on the World Happiness Report with the following charts:

Heatmap Showing Correlation: The heatmap visualizes the correlation between attributes such as Economy (GDP per Capita), Family, Health (Life Expectancy), Freedom, Trust (Government Corruption), Generosity, and Happiness Score. The heatmap provides insights into how these variables are related to each other, highlighting potential patterns and relationships.

Scatter Plot to Identify the Effect of GDP per Capita on Happiness Score in Various Regions: The scatter plot illustrates the relationship between GDP per Capita and Happiness Score across different regions. By coloring the data points based on regions, it allows for a clear identification of how economic factors impact happiness levels in various parts of the world.

Pie Chart to Present Happiness Score by Regions: The pie chart showcases the distribution of Happiness Scores across different regions. It provides a visual representation of how happiness levels vary geographically, offering a snapshot of the overall happiness levels in each region.

Map Displaying GDP per Capita of Countries with Healthy Life Expectancy as a Tooltip: The map visualizes the GDP per Capita of countries, with Healthy Life Expectancy included as a tooltip for additional information. By hovering over the data points, viewers can see the relationship between economic prosperity and health outcomes in different countries, offering a comprehensive view of global well-being.

By combining these visualizations in a dashboard format, viewers can gain a holistic understanding of the factors influencing happiness levels worldwide, from economic indicators to regional variations in well-being.

You can use this narrative to showcase your dashboard in a business meeting!

Conclusion

Congratulations! You have completed this guided project on using Generative AI for different data analytics tasks.

By the end of this project, you are now capable of using Generative AI for the tasks of:

- Data preparation
- Data analysis
- Dashboarding
- Storytelling

Author(s)

Dr. Pooja