Introduction to Matplotlib

Course Code: CPE 031	**Program**: Computer Engineering
Course Title: Visualization and Data Analysis	**Date Performed**: 10 / 22 / 2024
Section: CPE21S4	**Date Submitted**: 10 / 22 / 2024
Name: ROALLOS, Jean Gabriel Vincent G.	**Instructor**: Prof. Maria Rizette Sayo

Intended Learning Outcomes (ILO):

By the end of this laboratory session, learners will be able to:

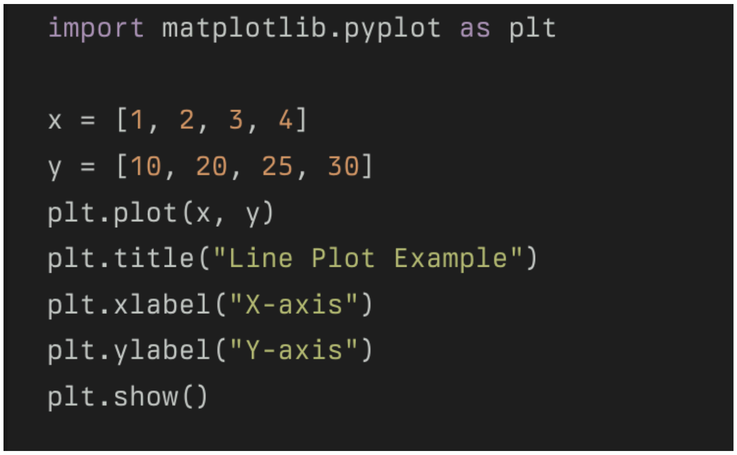
1. Utilize Matplotlib's pyplot interface to create a variety of visualizations, including line plots, scatter plots, histograms, and box plots, demonstrating an understanding of the library's syntax and functionality.

2. Customize visual elements such as titles, labels, and legends to enhance the clarity and aesthetics of their plots, applying best practices in data visualization.

3. Analyze and interpret visual data representations to extract meaningful insights, effectively communicating findings through well-structured graphical presentations.

Part 1: Perform the following codes, and understand the difference between line plot, scatter plot, histogram, bar chart, box plot, and pie chart using matplotlib's pyplot sub-module. **(Provide a screenshot of your output.)**

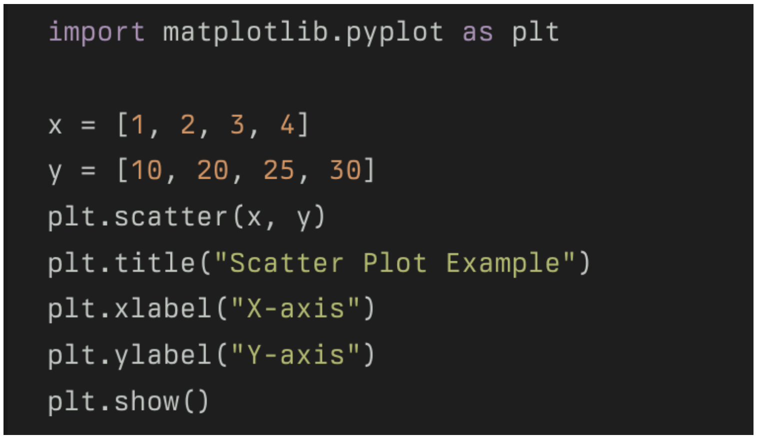
1. Line Plot



```python
import matplotlib.pyplot as plt

x = [1, 2, 3, 4]
y = [10, 20, 25, 30]
plt.plot(x, y)
plt.title("Line Plot Example")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.show()
```

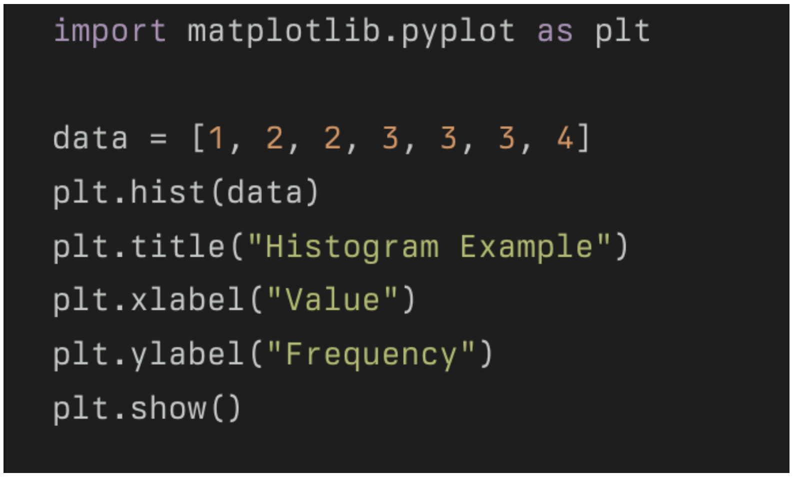
2. Scatter Plot



```
import matplotlib.pyplot as plt

x = [1, 2, 3, 4]
y = [10, 20, 25, 30]
plt.scatter(x, y)
plt.title("Scatter Plot Example")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.show()
```

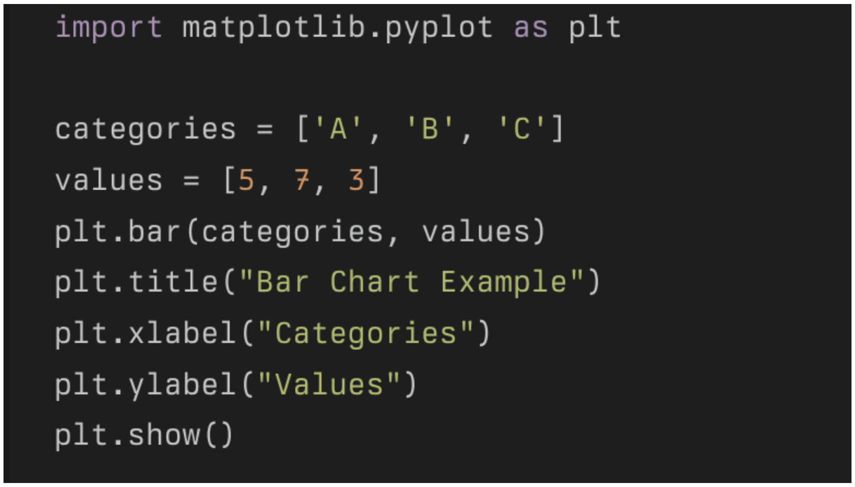
3. Histogram



```
import matplotlib.pyplot as plt

data = [1, 2, 2, 3, 3, 3, 4]
plt.hist(data)
plt.title("Histogram Example")
plt.xlabel("Value")
plt.ylabel("Frequency")
plt.show()
```

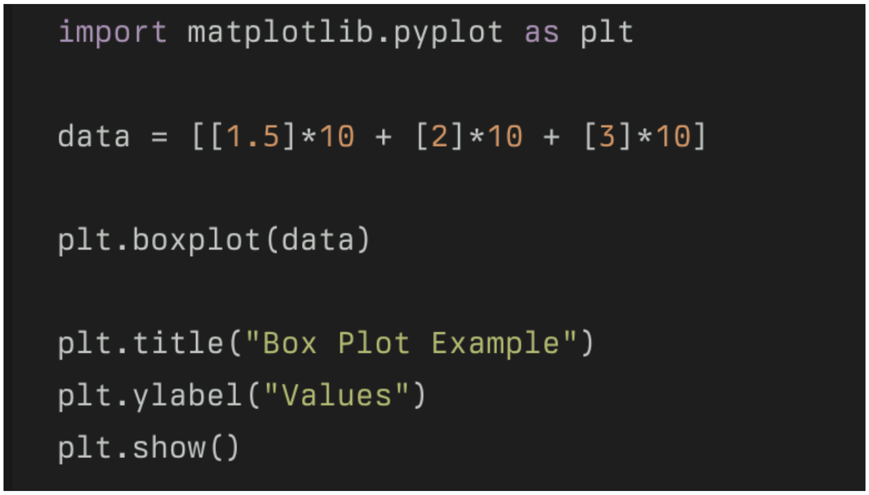
4. Bar Chart



```
import matplotlib.pyplot as plt

categories = ['A', 'B', 'C']
values = [5, 7, 3]
plt.bar(categories, values)
plt.title("Bar Chart Example")
plt.xlabel("Categories")
plt.ylabel("Values")
plt.show()
```

5. Box plot



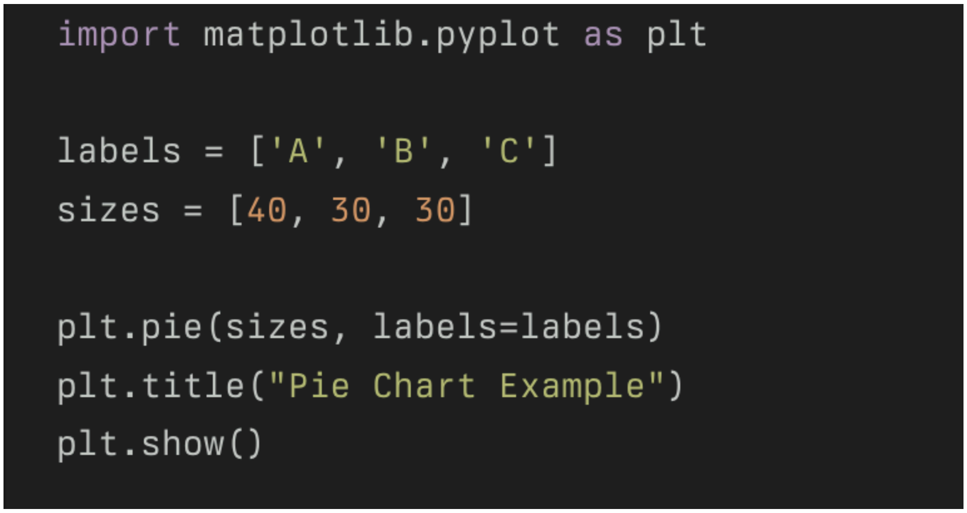
```
import matplotlib.pyplot as plt

data = [[1.5]*10 + [2]*10 + [3]*10]

plt.boxplot(data)

plt.title("Box Plot Example")
plt.ylabel("Values")
plt.show()
```

6. Pie chart



```
import matplotlib.pyplot as plt

labels = ['A', 'B', 'C']
sizes = [40, 30, 30]

plt.pie(sizes, labels=labels)
plt.title("Pie Chart Example")
plt.show()
```

Part 2: Refer to the instructions below.

1. **Find a dataset for this activity**: Please visit Kaggle and look for a new dataset that would allow you to perform visualization and analysis using matplotlib.

2. **Creating a dataframe from your CSV file:** Once you have successfully loaded your dataset, you need to create a dataframe from your uploaded CSV file

3. **Import the matplotlib.pyplot**

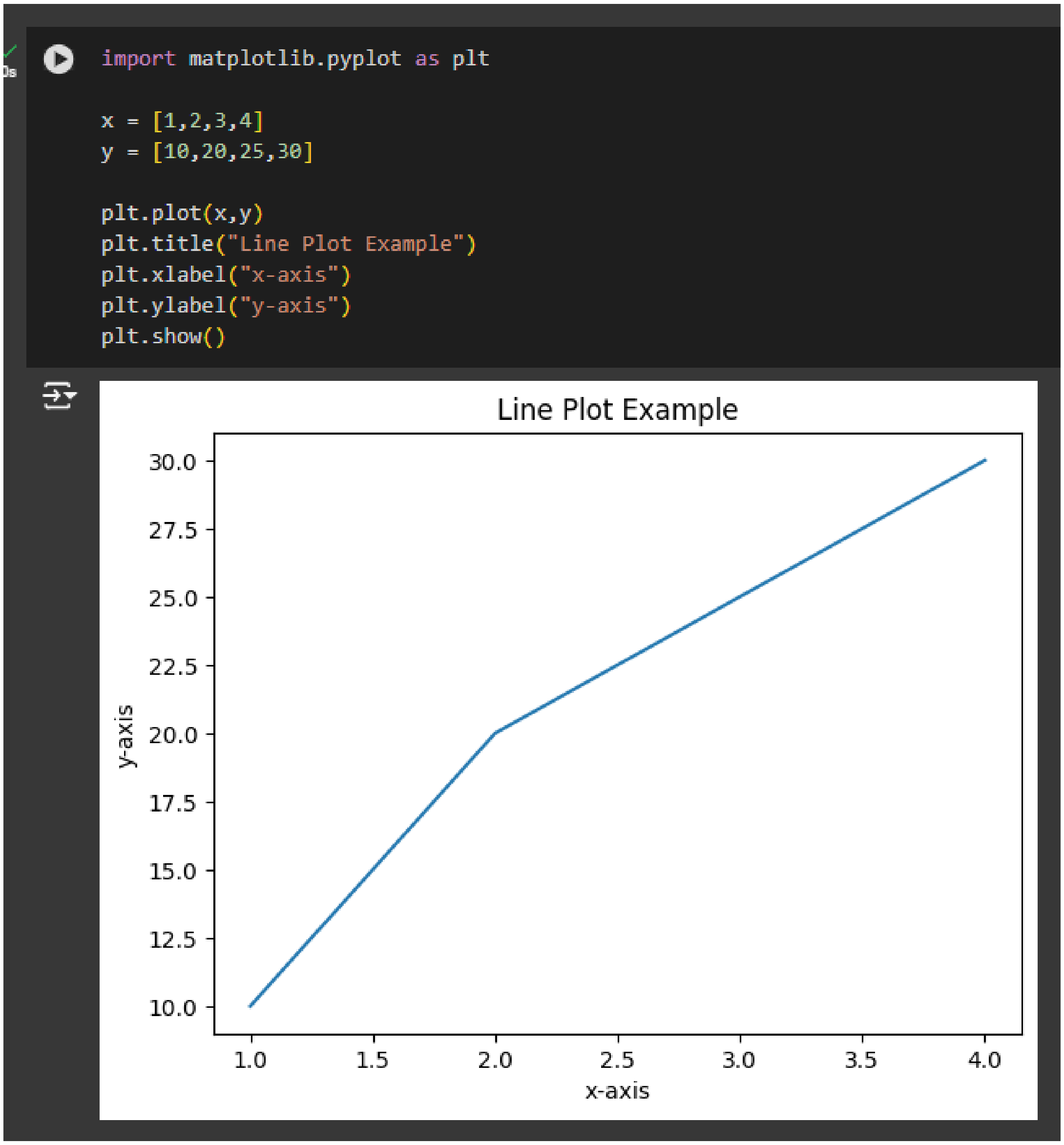
4. **Based on your chosen dataset, you will develop three questions that you will answer using pyplot visualizations. This means that you will need to produce at least three pyplot visualizations. You are also required to make certain customizations on your data vizes.**

5. Provide observations for each of your data viz, then **produce one insight not longer than five sentences given your three observations**. Your output shall follow this outline:
    a. Introduction (Describe your dataset)
    b. Questions
    c. Visualization and Observation
    d. Insight

6. Your grade will depend on the quality of the question, difficulty/complexity of the visualization, and value-add of the insight that you will generate.

Part 1 Output:

1.

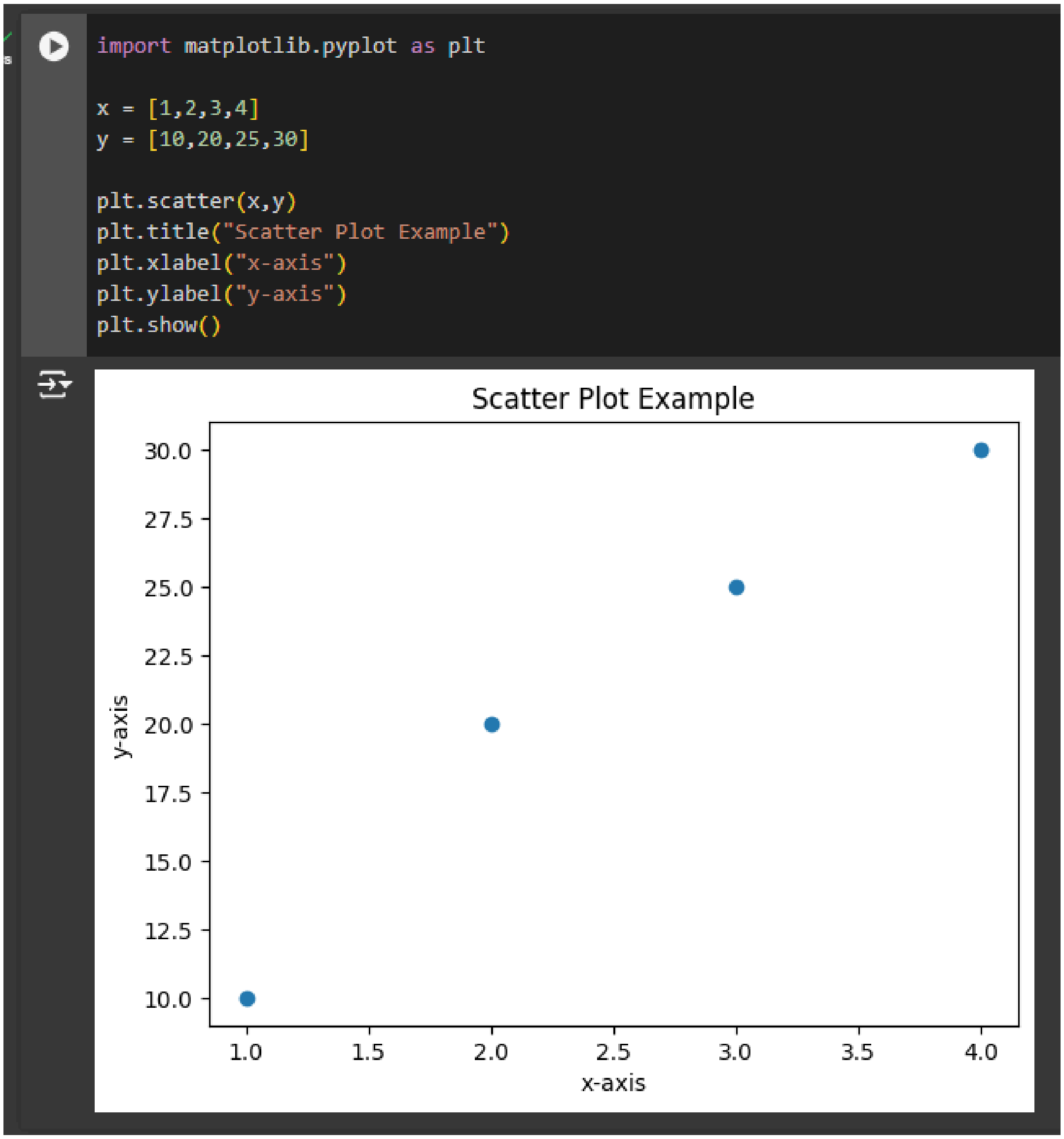


```
import matplotlib.pyplot as plt

x = [1,2,3,4]
y = [10,20,25,30]

plt.plot(x,y)
plt.title("Line Plot Example")
plt.xlabel("x-axis")
plt.ylabel("y-axis")
plt.show()
```

2.

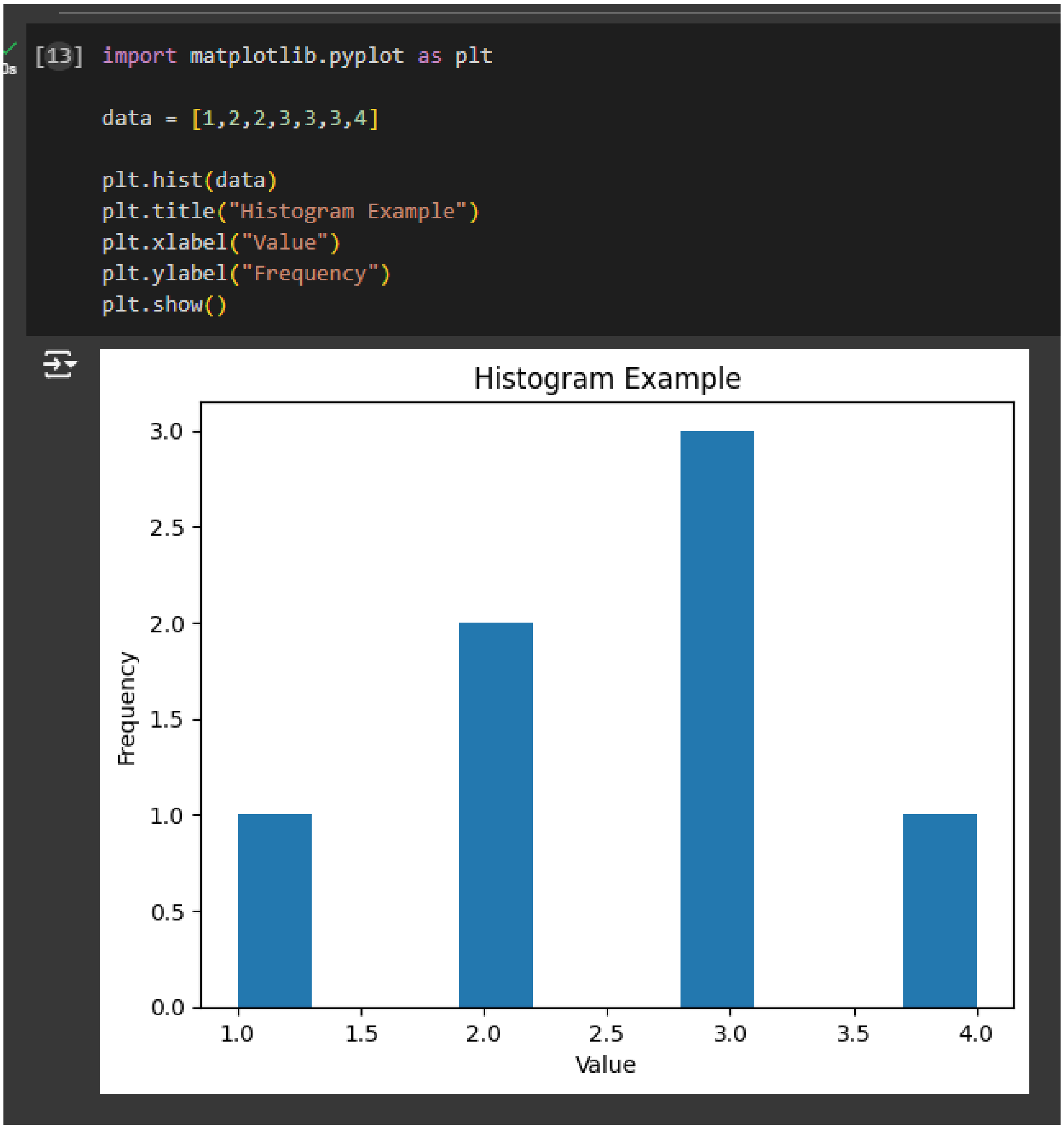


```
import matplotlib.pyplot as plt

x = [1,2,3,4]
y = [10,20,25,30]

plt.scatter(x,y)
plt.title("Scatter Plot Example")
plt.xlabel("x-axis")
plt.ylabel("y-axis")
plt.show()
```

3.

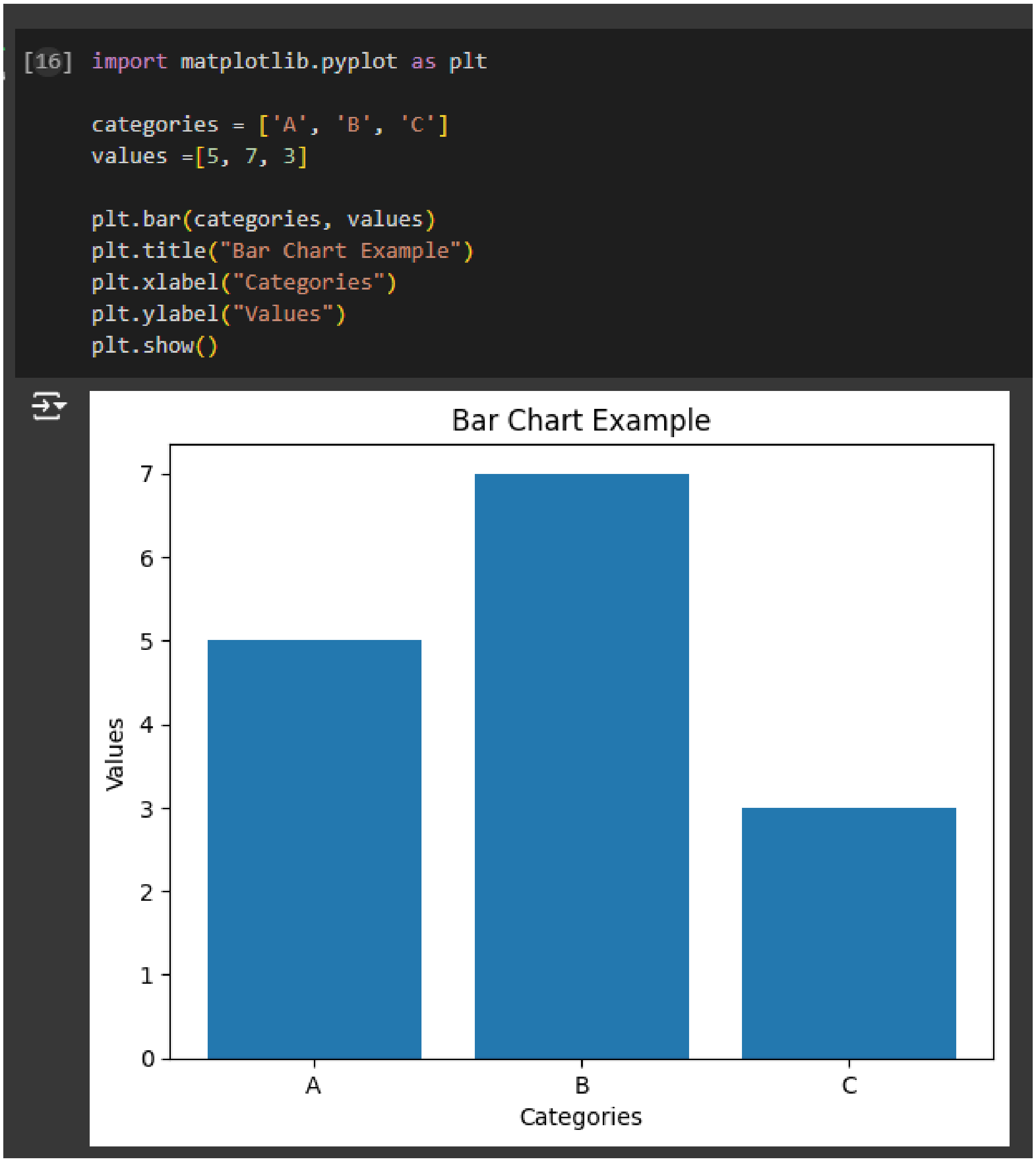


```
[13] import matplotlib.pyplot as plt

     data = [1,2,2,3,3,3,4]

     plt.hist(data)
     plt.title("Histogram Example")
     plt.xlabel("Value")
     plt.ylabel("Frequency")
     plt.show()
```

4.

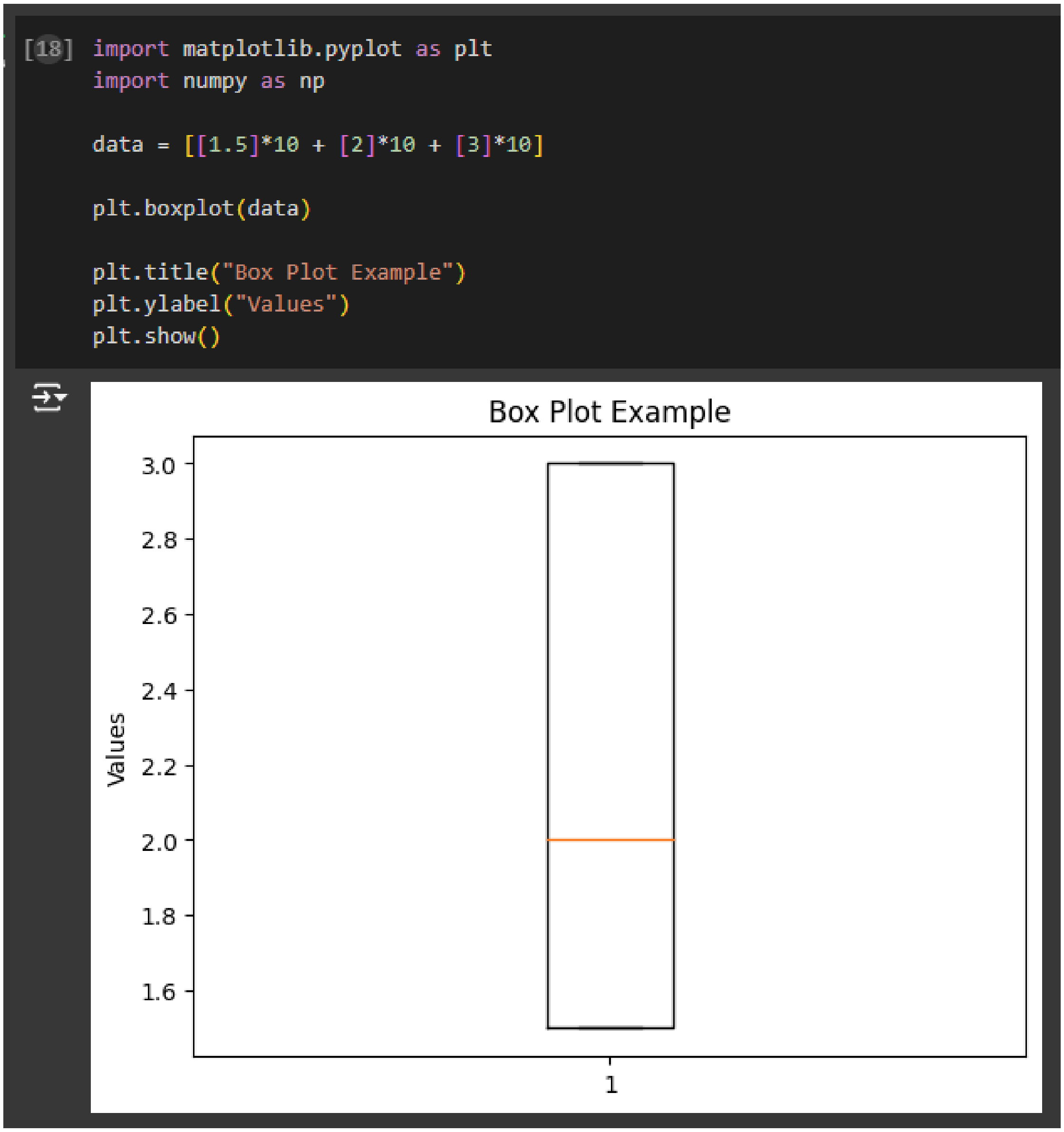


```
[16] import matplotlib.pyplot as plt

     categories = ['A', 'B', 'C']
     values =[5, 7, 3]

     plt.bar(categories, values)
     plt.title("Bar Chart Example")
     plt.xlabel("Categories")
     plt.ylabel("Values")
     plt.show()
```

5.



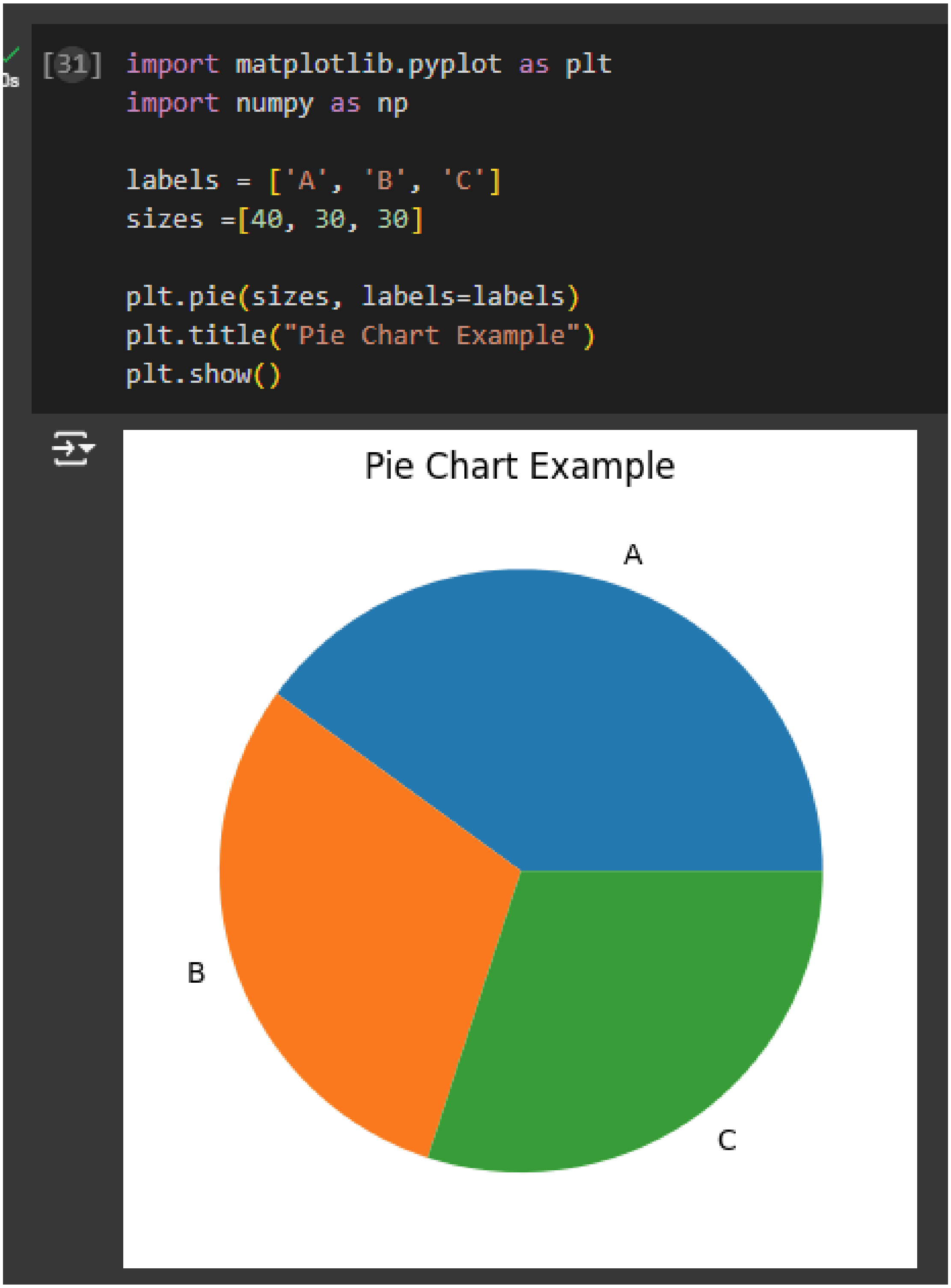
```
[18] import matplotlib.pyplot as plt
     import numpy as np

     data = [[1.5]*10 + [2]*10 + [3]*10]

     plt.boxplot(data)

     plt.title("Box Plot Example")
     plt.ylabel("Values")
     plt.show()
```

6.

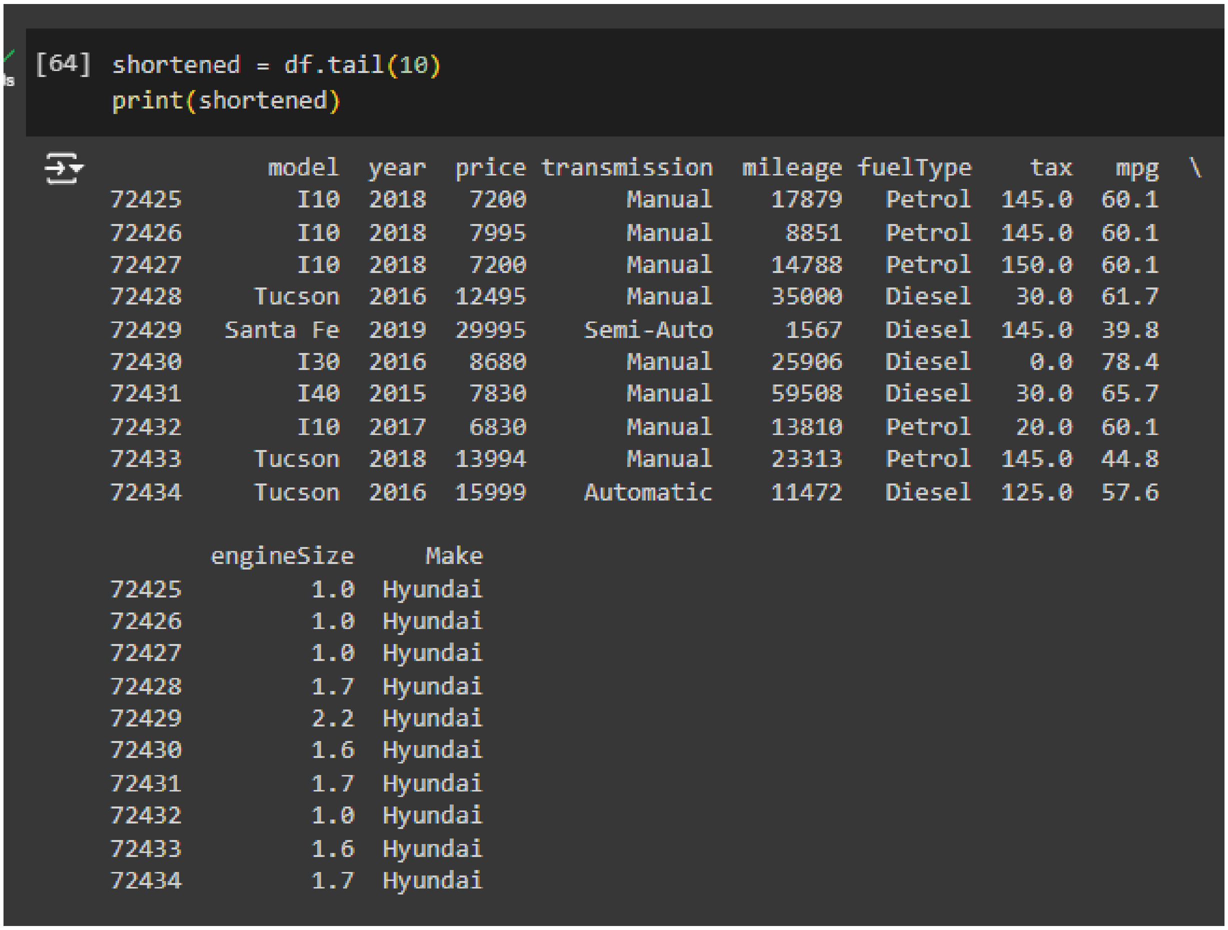


```
import matplotlib.pyplot as plt
import numpy as np

labels = ['A', 'B', 'C']
sizes =[40, 30, 30]

plt.pie(sizes, labels=labels)
plt.title("Pie Chart Example")
plt.show()
```

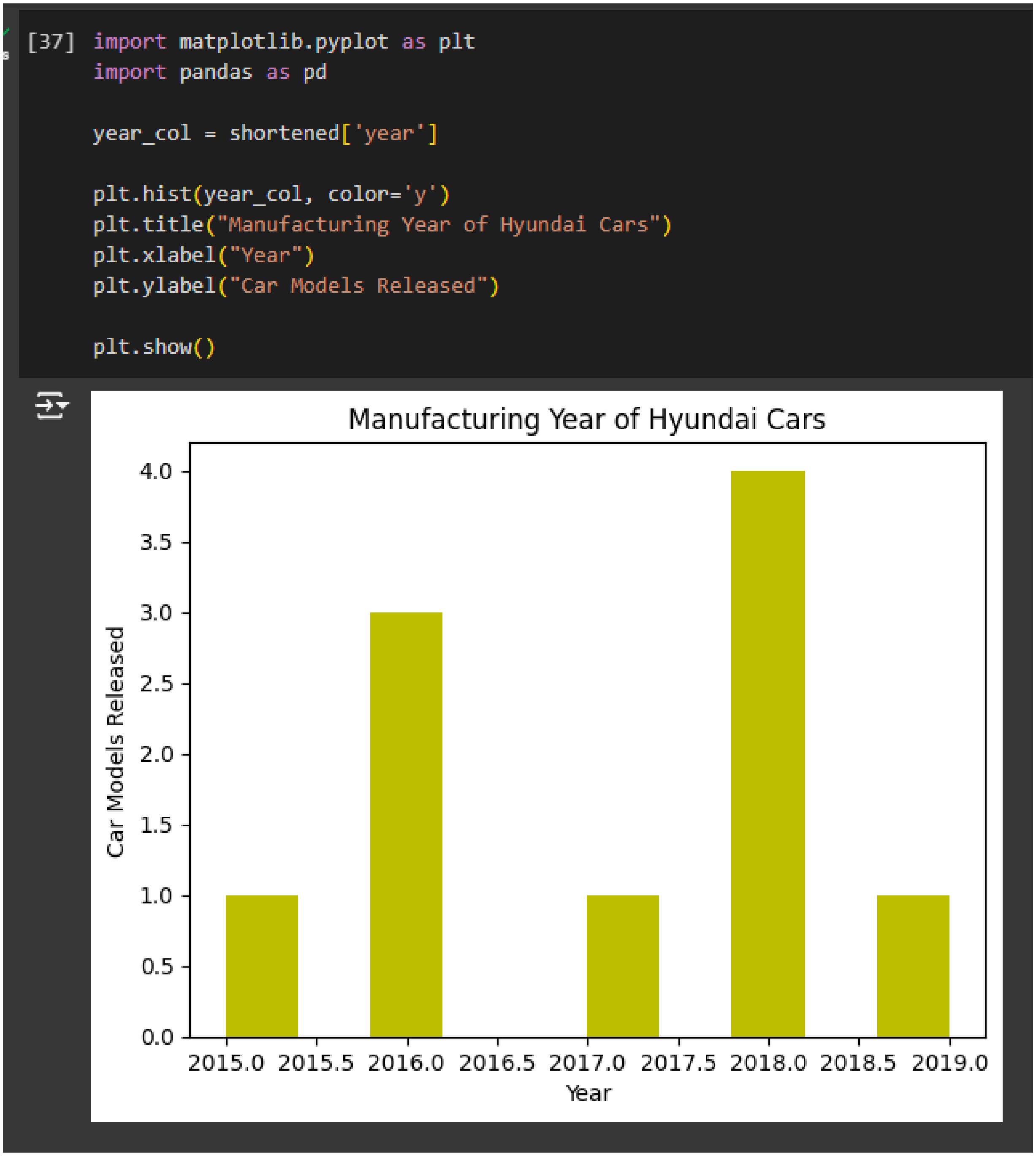
Part 2 Output:



```
shortened = df.tail(10)
print(shortened)
```

```
           model  year  price transmission  mileage fuelType    tax   mpg  \
72425        I10  2018   7200       Manual    17879   Petrol  145.0  60.1   
72426        I10  2018   7995       Manual     8851   Petrol  145.0  60.1   
72427        I10  2018   7200       Manual    14788   Petrol  150.0  60.1   
72428     Tucson  2016  12495       Manual    35000   Diesel   30.0  61.7   
72429   Santa Fe  2019  29995    Semi-Auto     1567   Diesel  145.0  39.8   
72430        I30  2016   8680       Manual    25906   Diesel    0.0  78.4   
72431        I40  2015   7830       Manual    59508   Diesel   30.0  65.7   
72432        I10  2017   6830       Manual    13810   Petrol   20.0  60.1   
72433     Tucson  2018  13994       Manual    23313   Petrol  145.0  44.8   
72434     Tucson  2016  15999    Automatic    11472   Diesel  125.0  57.6   

       engineSize     Make  
72425         1.0  Hyundai  
72426         1.0  Hyundai  
72427         1.0  Hyundai  
72428         1.7  Hyundai  
72429         2.2  Hyundai  
72430         1.6  Hyundai  
72431         1.7  Hyundai  
72432         1.0  Hyundai  
72433         1.6  Hyundai  
72434         1.7  Hyundai  
```



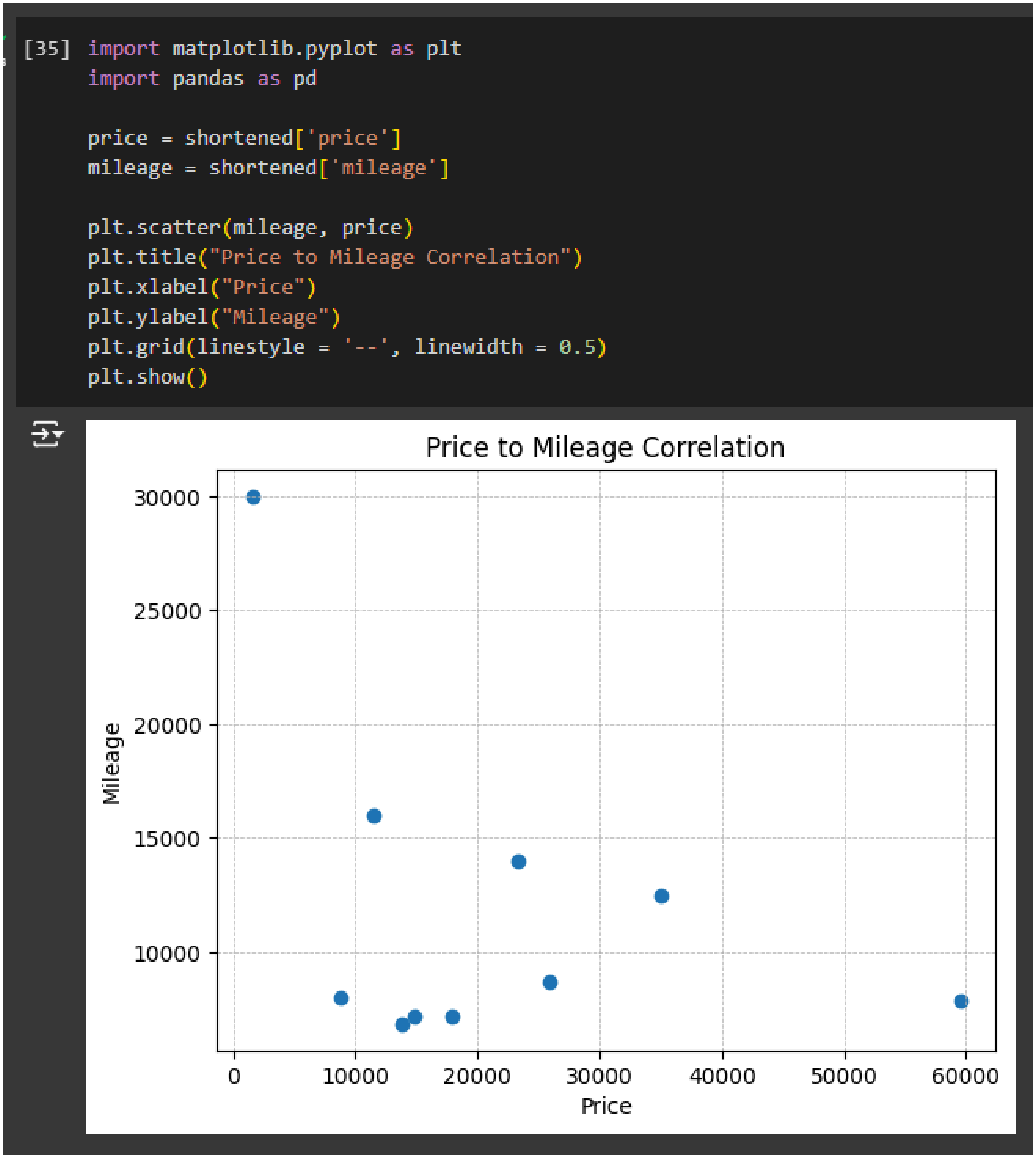
```
[37] import matplotlib.pyplot as plt
     import pandas as pd

     year_col = shortened['year']

     plt.hist(year_col, color='y')
     plt.title("Manufacturing Year of Hyundai Cars")
     plt.xlabel("Year")
     plt.ylabel("Car Models Released")

     plt.show()
```

The selected dataset has a set of Hyundai models and their following attributes. One question we can ask is: **What year did Hyundai release or update their car models the most?** By laying out the necessary data, we find out that Hyundai has released or updated their models the most in the year 2018, releasing a total of 4 new models. We easily identify this statistic due to the histogram that we used to illustrate the difference.

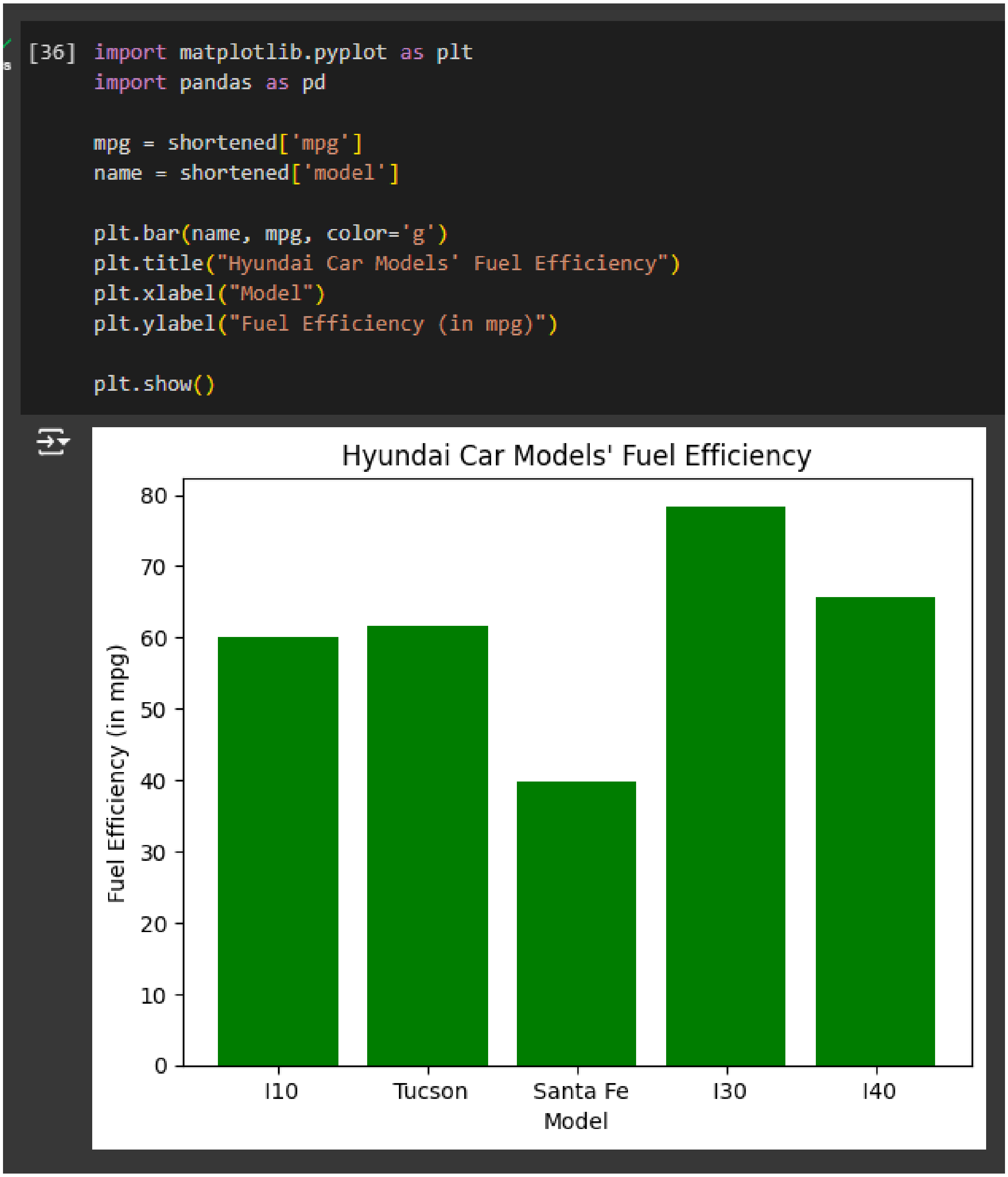


```
[35] import matplotlib.pyplot as plt
     import pandas as pd

     price = shortened['price']
     mileage = shortened['mileage']

     plt.scatter(mileage, price)
     plt.title("Price to Mileage Correlation")
     plt.xlabel("Price")
     plt.ylabel("Mileage")
     plt.grid(linestyle = '--', linewidth = 0.5)
     plt.show()
```

As the selected dataset includes the current mileage of the car and a listed price, we can identify that the cars recorded in the dataset are used. We can ask the question: **Does more mileage on a used car affect a car's resale price?** As we can see, the data has a weak negative correlation. This means that as the mileage statistic decreases, the price of the car increases. But we must note that it has a weak correlation.



```
[36] import matplotlib.pyplot as plt
     import pandas as pd

     mpg = shortened['mpg']
     name = shortened['model']

     plt.bar(name, mpg, color='g')
     plt.title("Hyundai Car Models' Fuel Efficiency")
     plt.xlabel("Model")
     plt.ylabel("Fuel Efficiency (in mpg)")

     plt.show()
```

Also included in the dataset are the fuel efficiency rating of the cars from the selected part of the dataset. We can therefore ask, **What Hyundai model has the least and best fuel efficiency rating?** By using a bar chart, We can compare the various models' fuel efficiency ratings. By doing so, we can now identify that the I30 has the best fuel efficiency rating, almost reaching 80 miles per gallon, while the Santa Fe has the worst rating, only doing 40 miles per gallon.