

V. EDA – DATA VISUALIZATION

AIM:

To explore and understand the underlying patterns, distributions, and relationships within the dataset through visual representations, which aids in uncovering insights, detecting anomalies, and guiding further analysis.

PROCEDURE:

1. Import matplotlib, pandas, and numpy for data handling and visualization.
2. Load your dataset using pandas and check first few rows for review.
3. Create a line plot using numpy's generated x values and plot a sine wave for demo visualization.
4. Build a bar plot to display values grouped by categories, enhancing categorical insight.
5. Draw a histogram to show data distribution and frequency of a continuous variable.

PROGRAM:

```
import matplotlib.pyplot as plt
```

```
import pandas as pd
```

```
import numpy as np
```

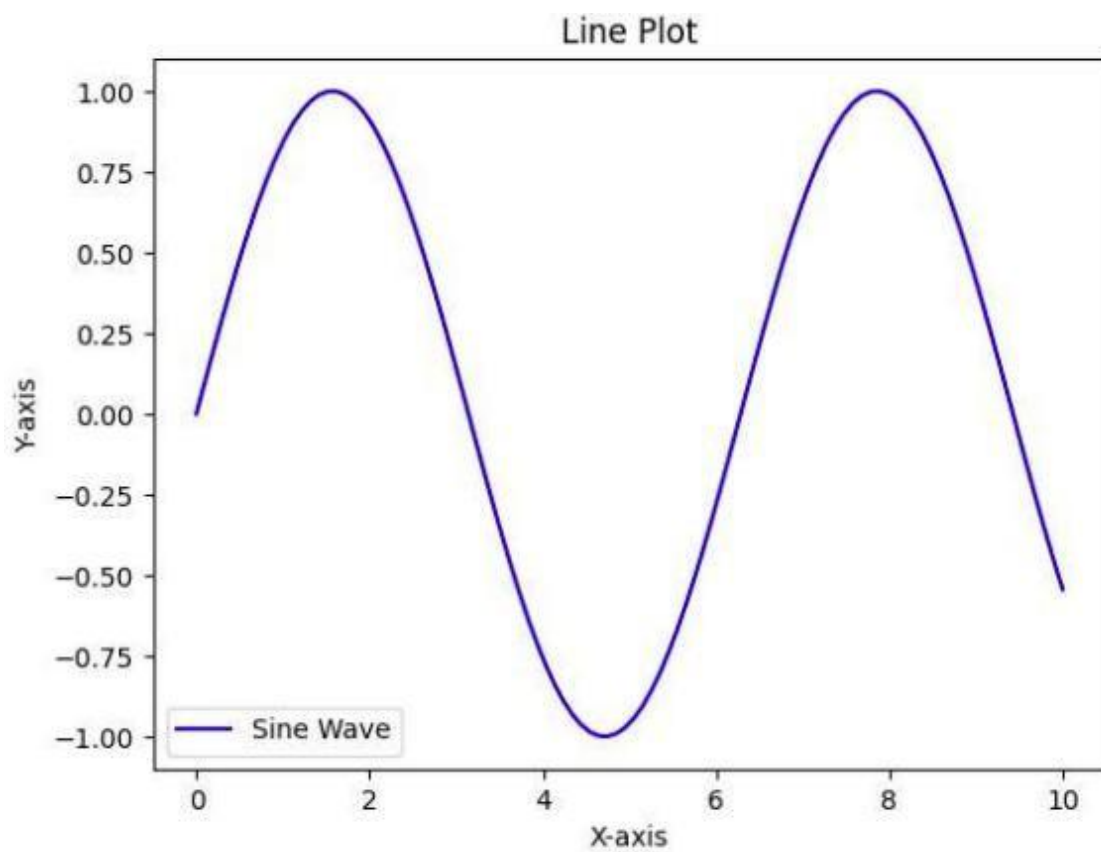
```
df = pd.read_csv("/content/data.csv") # Replace with your file
```

```
df.head()
```

	Make	Model	Year	Engine Type	Fuel Type	Engine HP	Engine Cylinders	Transmission Type	Driven_wheels	Number of Doors	Market Category	Vehicle Size	Vehicle Style	highway MPG	city mpg	Popularity	MSRP
0	BMW	1 Series M	2011	premium unleaded (required)		335.0	6.0	MANUAL	rear wheel drive	2.0	Factory Tuner,Luxury,High-Performance	Compact	Coupe	26	19	3916	46135
1	BMW	1 Series	2011	premium unleaded (required)		300.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,Performance	Compact	Convertible	28	19	3916	40650
2	BMW	1 Series	2011	premium unleaded (required)		300.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,High-Performance	Compact	Coupe	28	20	3916	36350
3	BMW	1 Series	2011	premium unleaded (required)		230.0	6.0	MANUAL	rear wheel drive	2.0	Luxury,Performance	Compact	Coupe	28	18	3916	29450
4	BMW	1 Series	2011	premium unleaded (required)		230.0	6.0	MANUAL	rear wheel drive	2.0	Luxury	Compact	Convertible	28	18	3916	34500

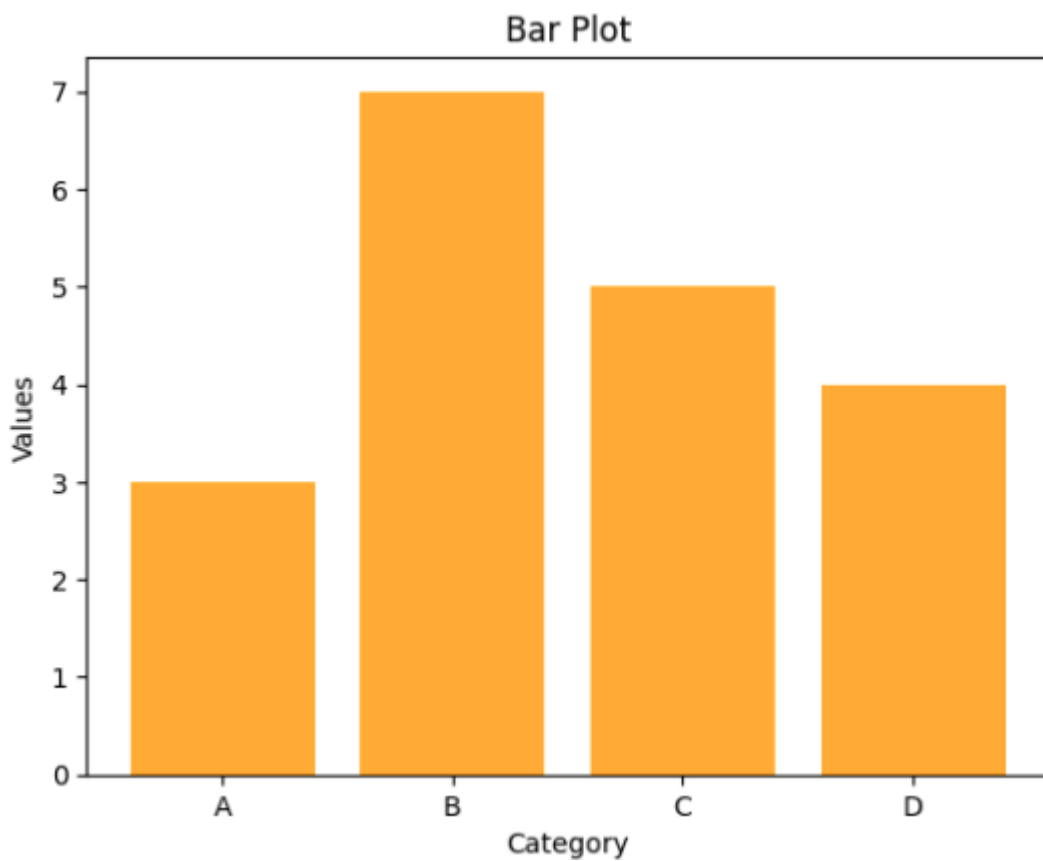
LINE CHART:

```
x = np.linspace(0, 10, 100)
y = np.sin(x)
plt.plot(x, y, color='blue', label='Sine Wave')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.title('Line Plot')
plt.legend()
plt.show()
```



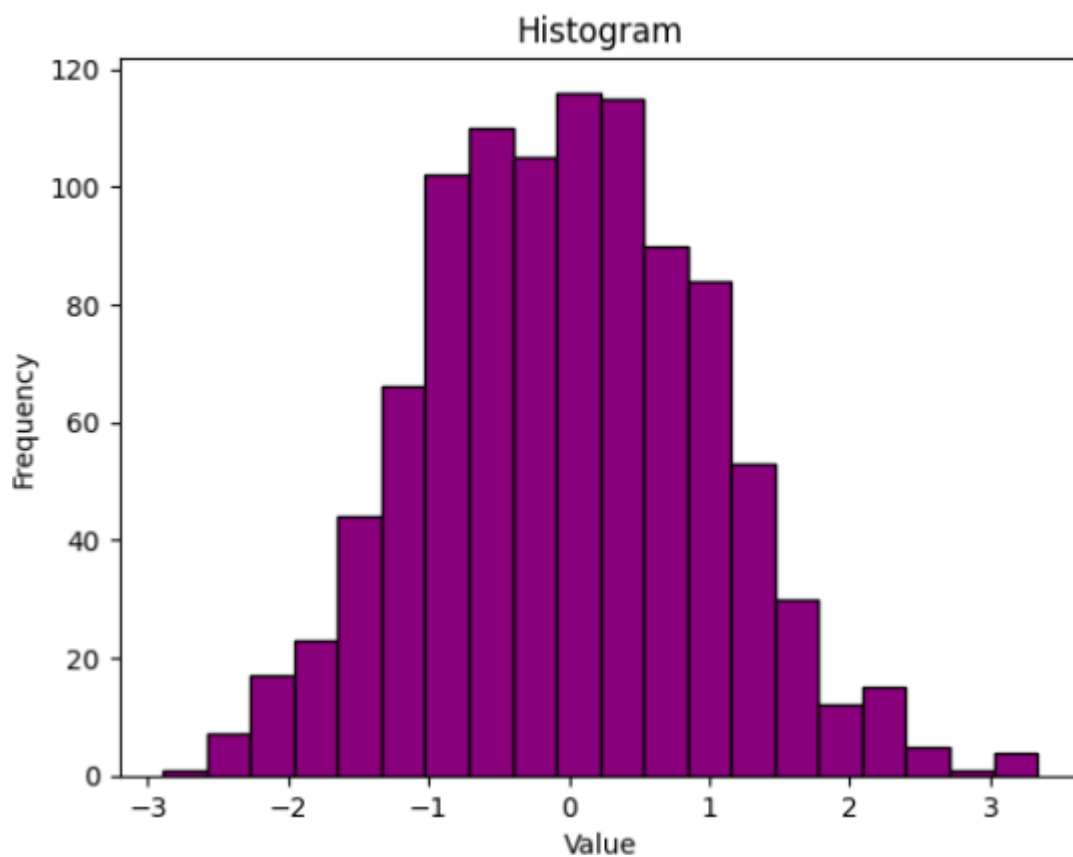
BAR CHART:

```
categories = ['A', 'B', 'C', 'D']  
values = [3, 7, 5, 4]  
plt.bar(categories, values, color='orange')  
plt.xlabel('Category')  
plt.ylabel('Values')  
plt.title('Bar Plot')  
plt.show()
```



HISTOGRAM:

```
data = np.random.randn(1000)
plt.hist(data, bins=20, color='purple', edgecolor='black')
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.title('Histogram')
plt.show()
```



RESULT:

Thus, the given program was written and executed successfully.