Visualization Library Documentation

1. Introduction

Matplotlib

Matplotlib is a widely used visualization library in Python that provides a variety of static, animated, and interactive plots. It is highly customizable and serves as the foundation for many other visualization tools. It is particularly useful for generating publication-quality figures and supports various backends for rendering graphs.

Use Cases:
- Exploratory Data Analysis (EDA)
- Scientific and engineering plotting
- Customizable reports and dashboards

Seaborn

Seaborn is a statistical data visualization library based on Matplotlib. It provides high-level functions for drawing attractive and informative statistical graphics. Seaborn is particularly useful for working with Pandas DataFrames and simplifies complex visualization tasks.

Use Cases:
- Statistical data visualization
- Heatmaps and correlation matrices
- Time series and categorical data analysis

2. Graph Types and Examples

Matplotlib Graphs

1. Line Plot
Use Case:
Visualizing trends over time, such as stock prices or temperature variations.

```
CODE BLOCK START
import matplotlib.pyplot as plt
import numpy as np

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

```
plt.legend()
plt.show()
```
CODE BLOCK START

2. Scatter Plot
Use Case:
Analyzing relationships between two continuous variables.

CODE BLOCK START
```
x = np.random.rand(50)
y = np.random.rand(50)
plt.scatter(x, y, color='red')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.title('Scatter Plot Example')
plt.show()
```
CODE BLOCK START

3. Bar Chart
Use Case:
Comparing categorical data, such as sales figures across different regions.

CODE BLOCK START
```
categories = ['A', 'B', 'C', 'D']
values = [4, 7, 1, 8]
plt.bar(categories, values, color='blue')
plt.xlabel('Categories')
plt.ylabel('Values')
plt.title('Bar Chart Example')
plt.show()
```
CODE BLOCK START

4. Histogram
Use Case:
Displaying the distribution of a dataset.

CODE BLOCK START
```
data = np.random.randn(1000)
plt.hist(data, bins=30, color='green', alpha=0.7)
plt.xlabel('Value')
plt.ylabel('Frequency')
plt.title('Histogram Example')
plt.show()
```
CODE BLOCK START

Seaborn Graphs

1. Line Plot
Use Case:
Similar to Matplotlib, used for trend visualization.

```
CODE BLOCK START
import seaborn as sns
import pandas as pd
import numpy as np

data = pd.DataFrame({'x': np.arange(100), 'y': np.sin(np.linspace(0, 10, 100))})
sns.lineplot(x='x', y='y', data=data)
plt.title('Seaborn Line Plot')
plt.show()
CODE BLOCK START
```

2. Scatter Plot
Use Case:
Useful for visualizing relationships between two variables with additional aesthetics like color and size.

```
CODE BLOCK START
tips = sns.load_dataset("tips")
sns.scatterplot(x='total_bill', y='tip', hue='sex', data=tips)
plt.title('Seaborn Scatter Plot')
plt.show()
CODE BLOCK START
```

3. Bar Plot
Use Case:
Comparing categorical data with added statistical estimations.

```
CODE BLOCK START
sns.barplot(x='day', y='total_bill', data=tips)
plt.title('Seaborn Bar Plot')
plt.show()
CODE BLOCK START
```

4. Heatmap
Use Case:
Visualizing correlation matrices or frequency distributions.

```
CODE BLOCK START
```

```
corr = tips.corr()
sns.heatmap(corr, annot=True, cmap='coolwarm')
plt.title('Seaborn Heatmap')
plt.show()
CODE BLOCK START
```

3. Comparison of Matplotlib and Seaborn

Feature	Matplotlib	Seaborn
Ease of Use	Requires more code for customization	Simpler, built-in aesthetic defaults
Customization	Highly customizable	Limited but effective styling
Interactivity	Static and requires extra libraries for interactivity	Limited, mainly for static plots
Performance	Efficient but can be verbose	Optimized for statistical plots
Best for	General-purpose plotting	Statistical data visualization

4. Resources

- [Matplotlib Quick Start Guide](https://matplotlib.org/stable/users/explain/quick_start.html#quick-start)
- [Seaborn Introduction](https://seaborn.pydata.org/tutorial/introduction.html)
- [Plotly Documentation](https://plotly.com/python/distplot/)
- [Bokeh Documentation](https://docs.bokeh.org/en/latest/docs/user_guide/basic.html)
- [Pandas Visualization Guide](https://pandas.pydata.org/docs/user_guide/index.html)