Data Visualization
# (Type of Charts)

Python script that demonstrates how to create various types of graphs and charts for data visualization using popular libraries like Matplotlib, Seaborn, and Plotly. These libraries offer a variety of plotting functions for different types of data visualization.

1. Line Chart

```python
import matplotlib.pyplot as plt
# Sample data
x = [1, 2, 3, 4, 5]
y = [10, 20, 25, 30, 35]

# Create line chart
plt.plot(x, y)
plt.title("Line Chart")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.show()
```
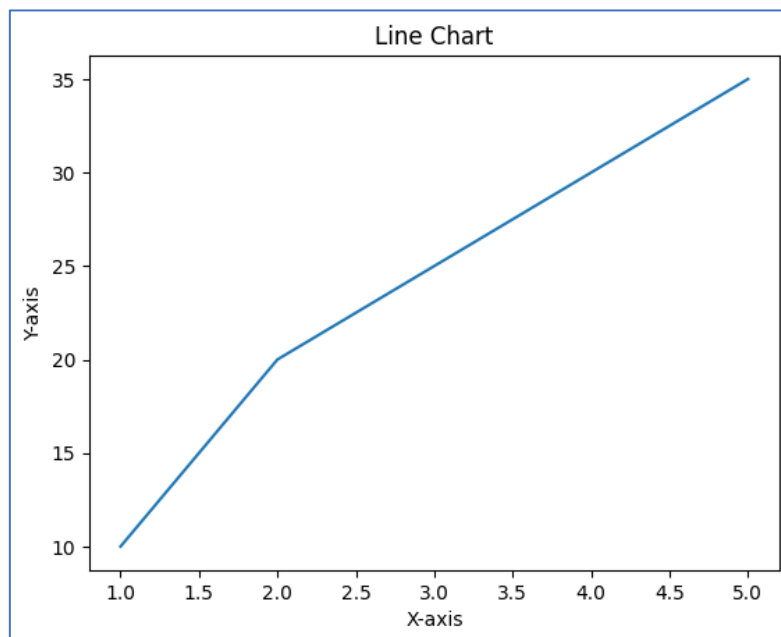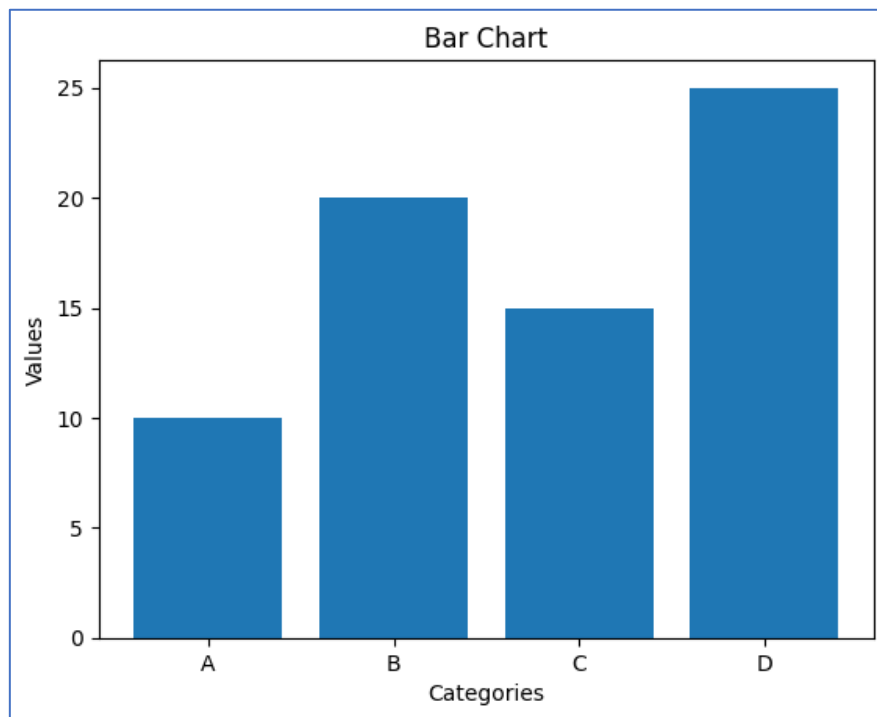
2. Bar Chart

```python
import matplotlib.pyplot as plt

# Sample data
categories = ['A', 'B', 'C', 'D']
values = [10, 20, 15, 25]


# Create bar chart
plt.bar(categories, values)
plt.title("Bar Chart")
plt.xlabel("Categories")
plt.ylabel("Values")
plt.show()
```
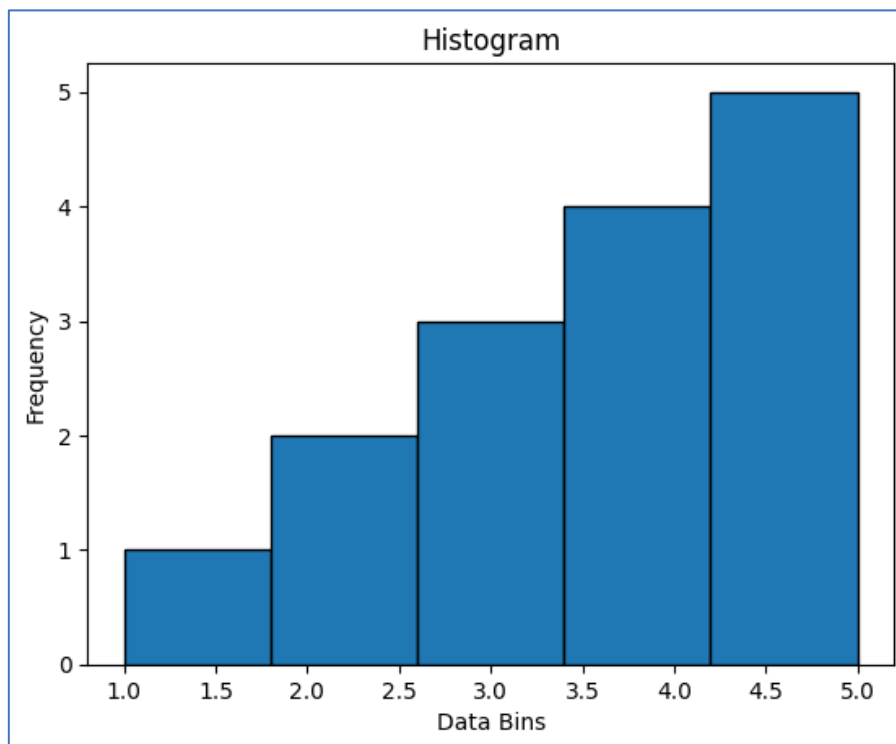
3. Histogram

```
import matplotlib.pyplot as plt
# Sample data
data = [1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5]

# Create histogram

plt.hist(data, bins=5, edgecolor='black')
plt.title("Histogram")
plt.xlabel("Data Bins")
plt.ylabel("Frequency")
plt.show()
```
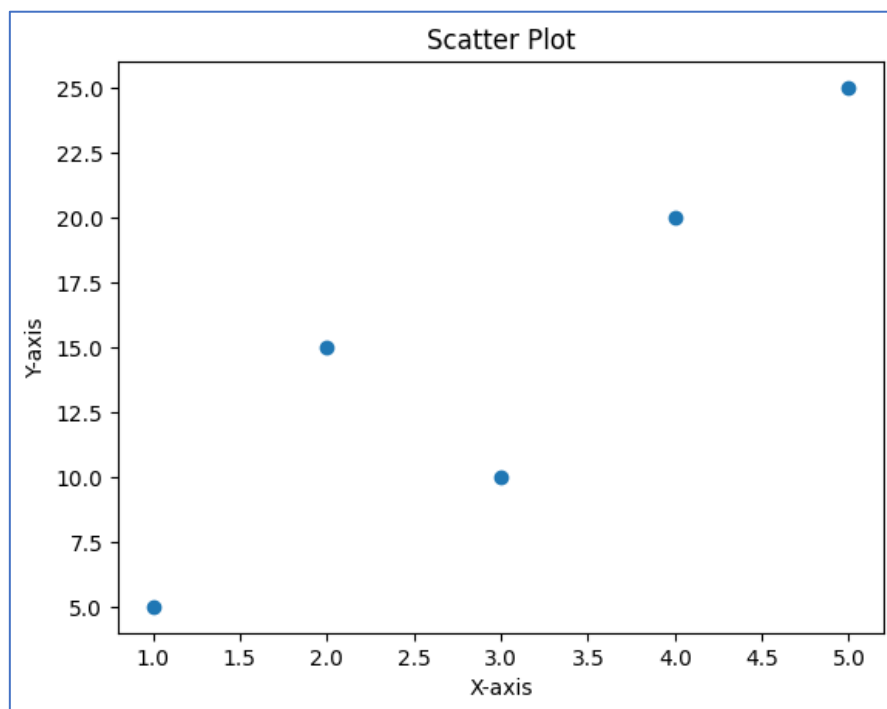
4. Scatter Plot

```python
import matplotlib.pyplot as plt

# Sample data
x = [1, 2, 3, 4, 5]
y = [5, 15, 10, 20, 25]

# Create scatter plot
plt.scatter(x, y)
plt.title("Scatter Plot")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.show()
```
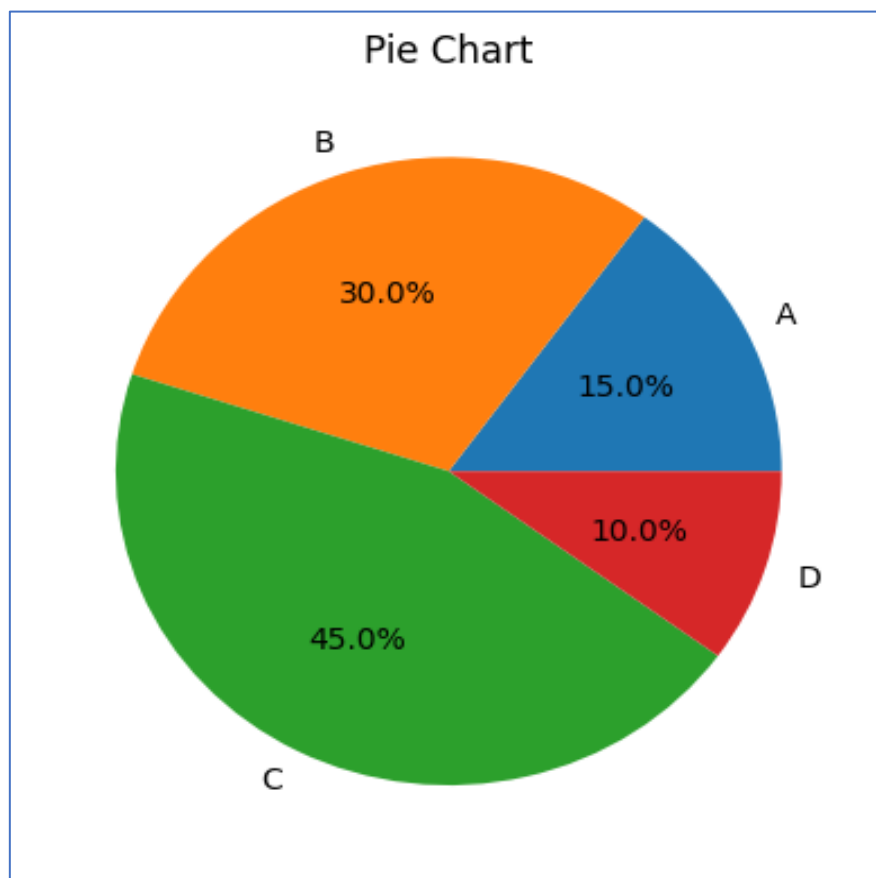
5. Pie Chart

```
import matplotlib.pyplot as plt

# Sample data
Country = ['USA', 'INDIA', 'CHINA', 'UK']
Population = [30, 145, 140, 25]

# Create pie chart
plt.pie(Population, labels= Country)
plt.title("Pie Chart")
plt.show()
```
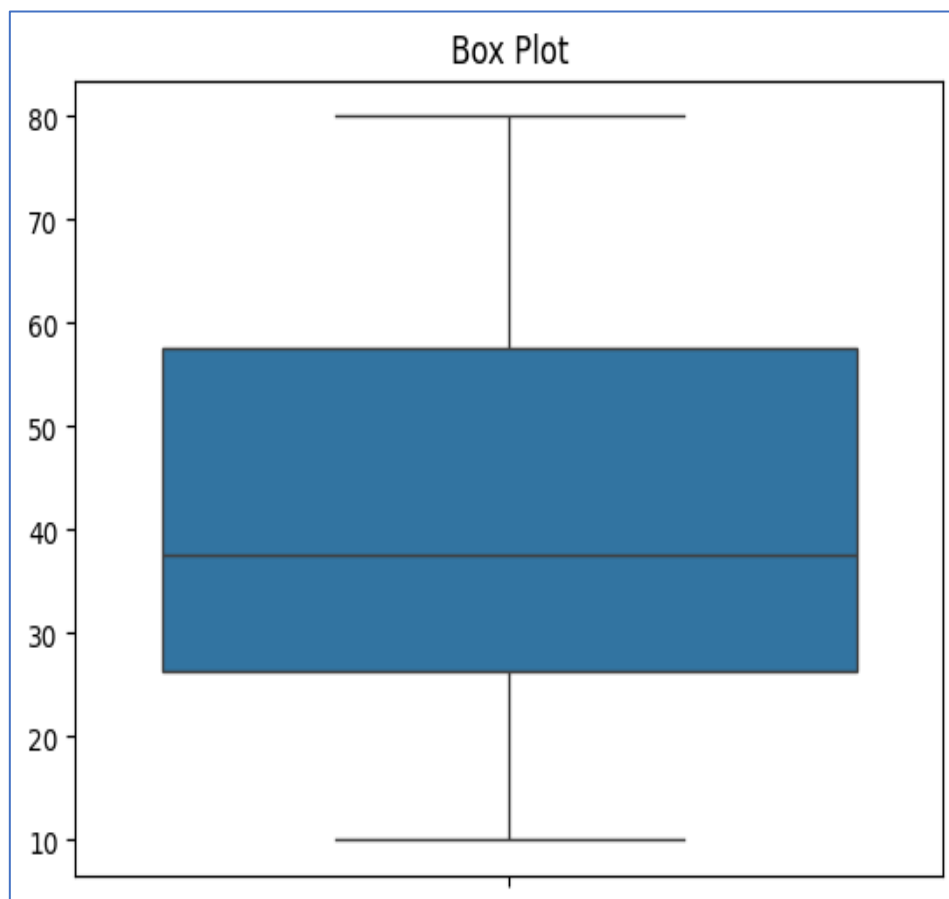
Pie Chart

B

30.0%

A

15.0%

10.0%

D

45.0%

C

6. Box Plot (with Seaborn)

```python
import seaborn as sns
import matplotlib.pyplot as plt

# Sample data
data = [10, 20, 25, 30, 35, 40, 50, 60, 70, 80]

# Create box plot
sns.boxplot(data=data)
plt.title("Box Plot")
plt.show()
```
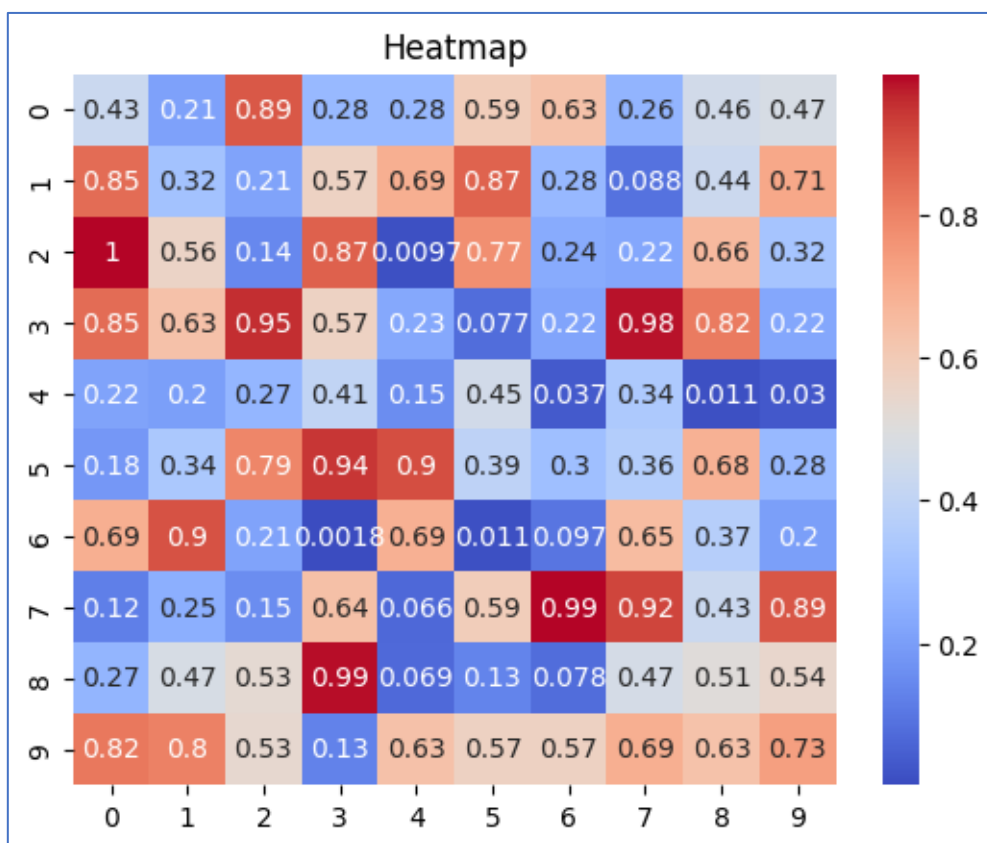
7. Heatmap (with Seaborn)

```python
import seaborn as sns
import numpy as np
import matplotlib.pyplot as plt

# Sample data
data = np.random.rand(10, 10)

# Create heatmap
sns.heatmap(data, annot=True, cmap='coolwarm')
plt.title("Heatmap")
plt.show()
```
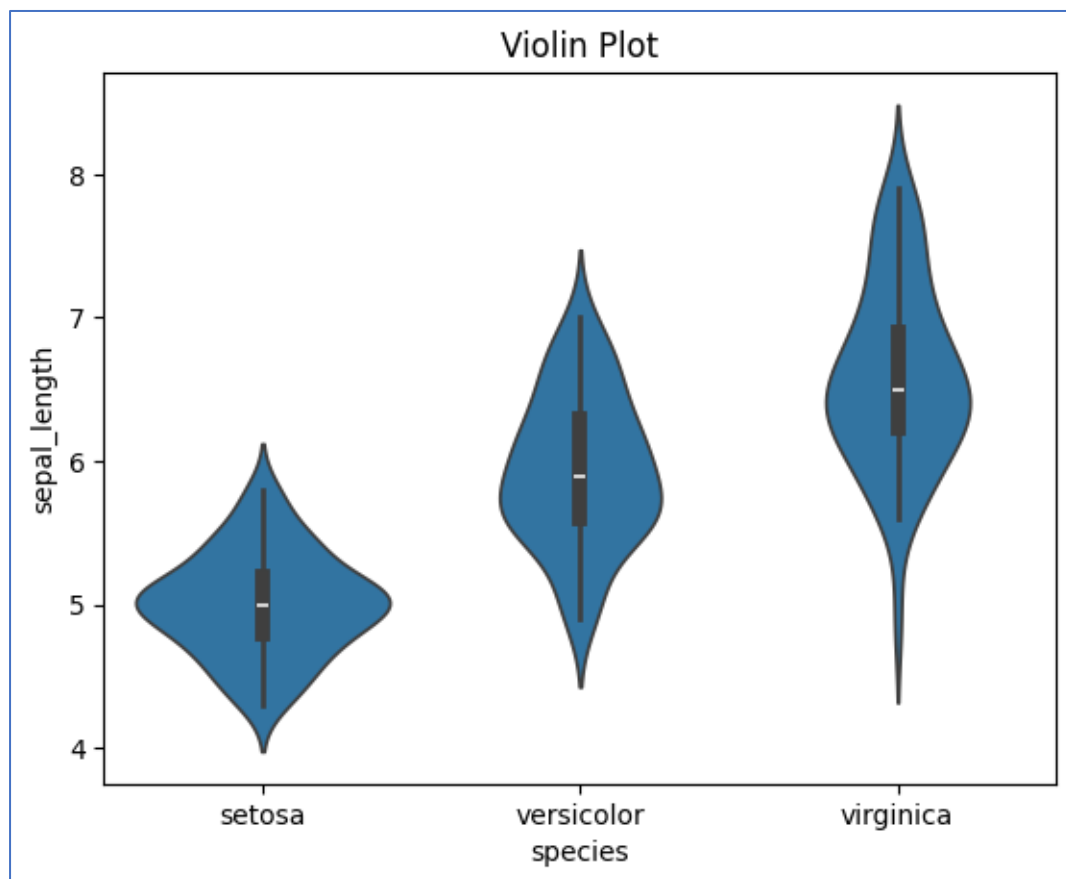
8. Violin Plot (with Seaborn)

```
import seaborn as sns
import matplotlib.pyplot as plt

# Sample data
data = sns.load_dataset("iris")

# Create violin plot
sns.violinplot(x="species", y="sepal_length", data=data)
plt.title("Violin Plot")
plt.show()
```
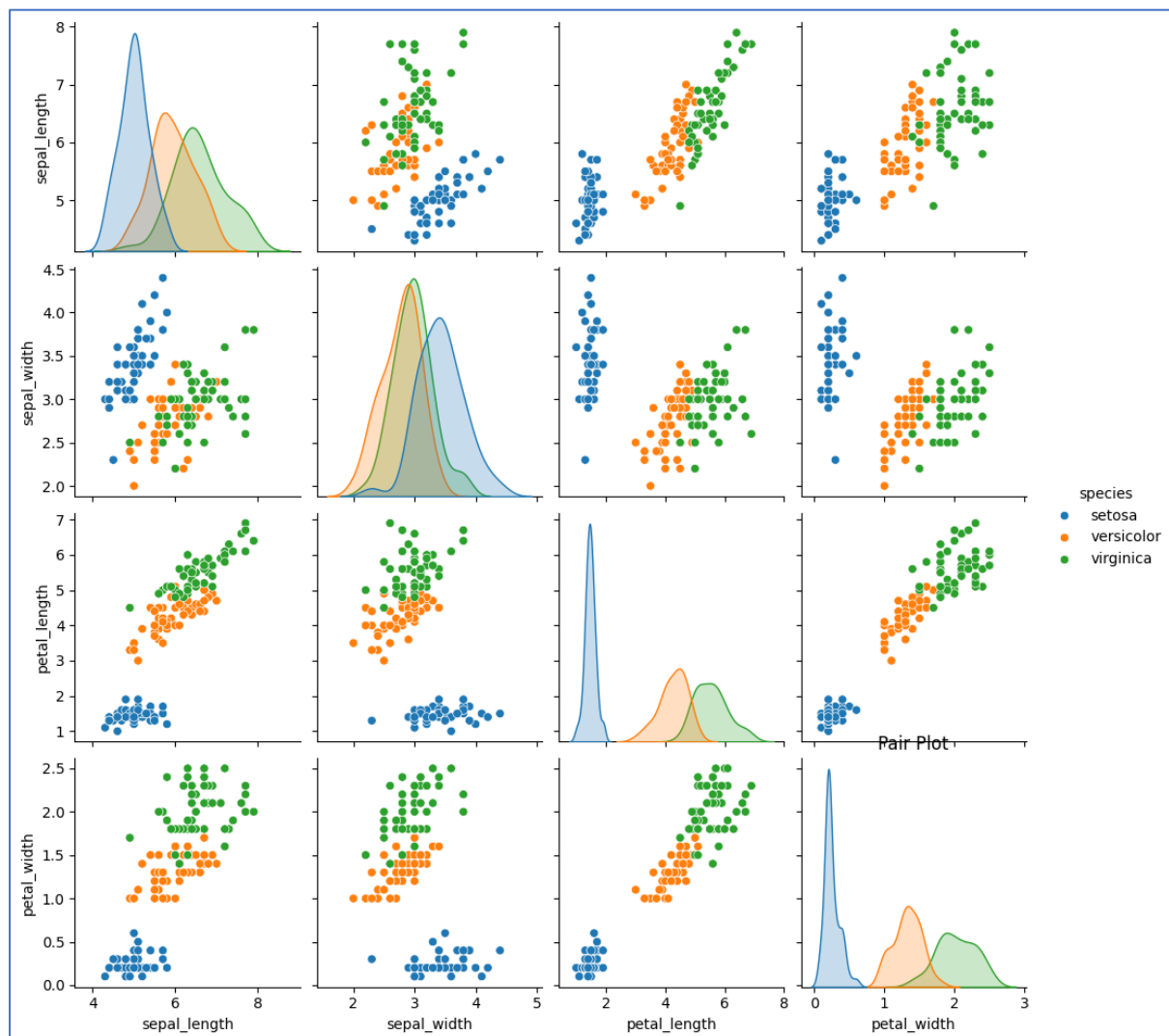
9. Pair Plot (with Seaborn)

```
import seaborn as sns
import matplotlib.pyplot as plt

# Sample data
data = sns.load_dataset("iris")

# Create pair plot
sns.pairplot(data, hue="species")
plt.title("Pair Plot")
plt.show()
```
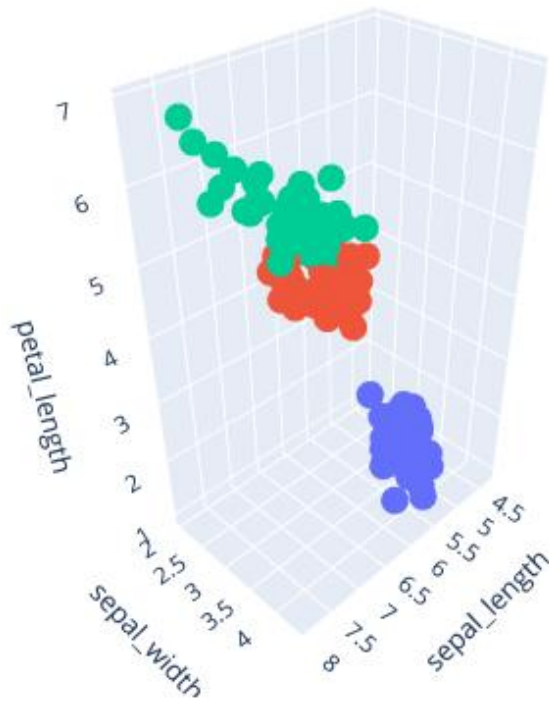
10. 3D Scatter Plot (with Plotly)

```
import plotly.express as px

# Sample data
df = px.data.iris()

# Create 3D scatter plot
fig = px.scatter_3d(df, x='sepal_length', y='sepal_width', z='petal_length',
color='species')
fig.update_layout(title="3D Scatter Plot")
fig.show()
```
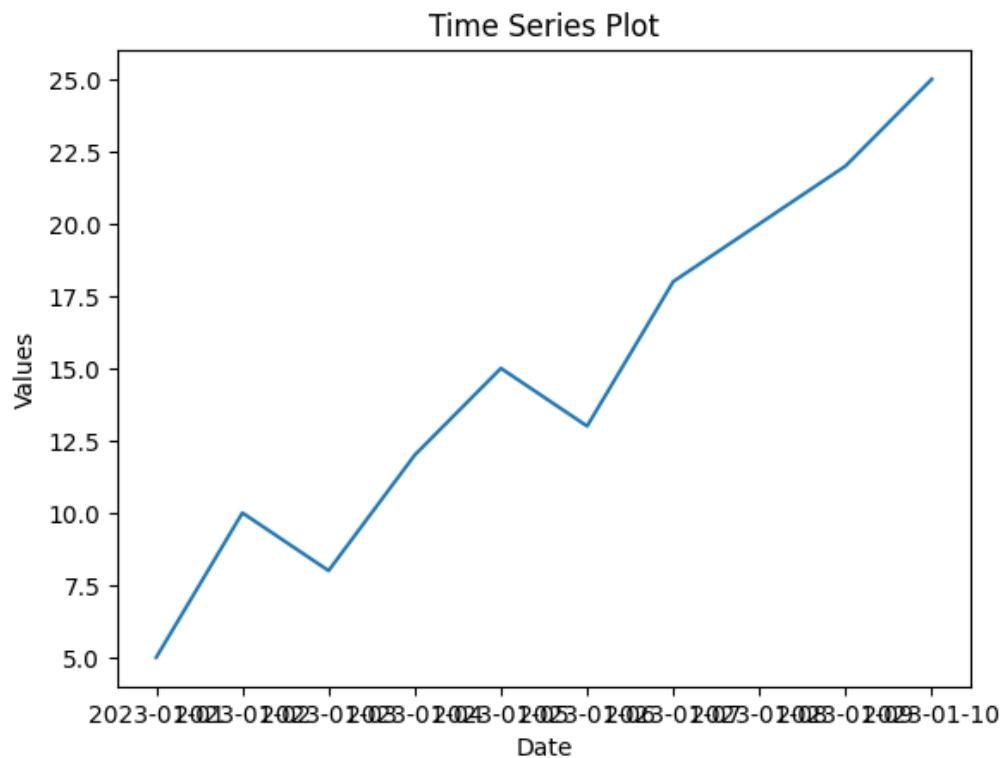
11. Time Series Plot

```python
import matplotlib.pyplot as plt
import pandas as pd

# Sample data
dates = pd.date_range('20230101', periods=10)

values = [5, 10, 8, 12, 15, 13, 18, 20, 22, 25]
# Create time series plot
plt.plot(dates, values)
plt.title("Time Series Plot")
plt.xlabel("Date")
plt.ylabel("Values")
plt.show()
```
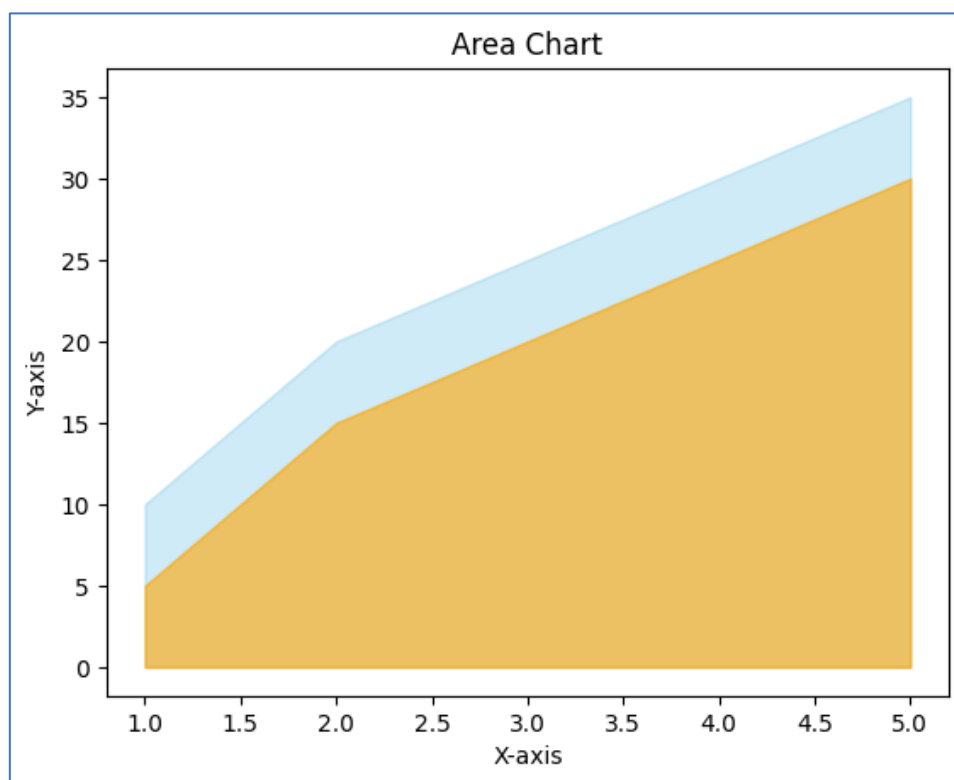
12. Area Chart

```python
import matplotlib.pyplot as plt

# Sample data
x = [1, 2, 3, 4, 5]
y1 = [10, 20, 25, 30, 35]
y2 = [5, 15, 20, 25, 30]

# Create area chart
plt.fill_between(x, y1, color="skyblue", alpha=0.4)
plt.fill_between(x, y2, color="orange", alpha=0.6)
plt.title("Area Chart")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.show()
```
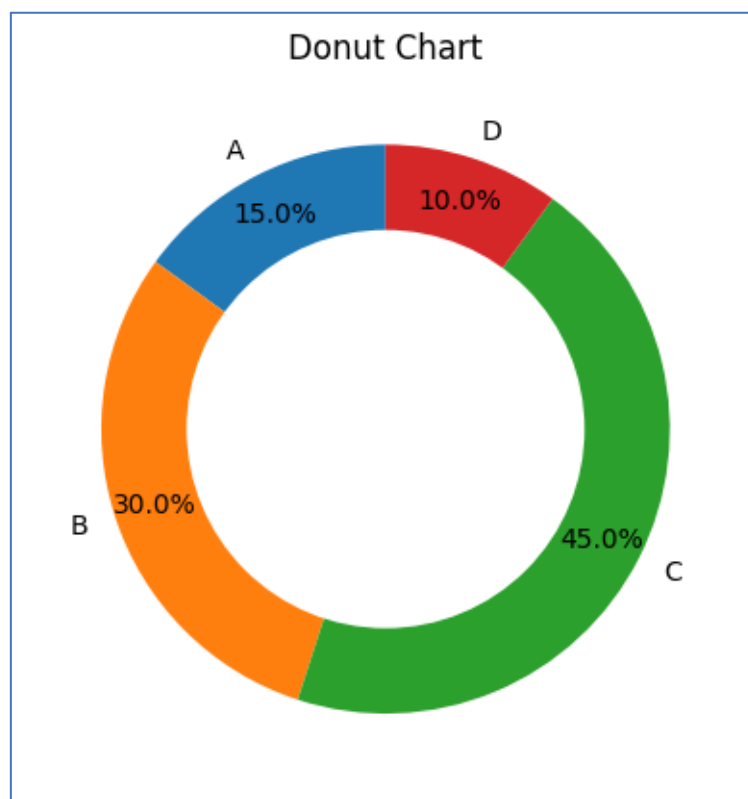
13. Donut Chart

```
import matplotlib.pyplot as plt

# Sample data
labels = ['A', 'B', 'C', 'D']
sizes = [15, 30, 45, 10]

# Create donut chart
plt.pie(sizes, labels=labels, autopct='%1.1f%%', startangle=90, pctdistance=0.85)
center_circle = plt.Circle((0, 0), 0.70, fc='white')
fig = plt.gcf()
fig.gca().add_artist(center_circle)
plt.title("Donut Chart")
plt.show()
```

14. Radar Chart

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

# Sample data
labels = np.array(['A', 'B', 'C', 'D', 'E'])
values = np.array([4, 3, 2, 5, 4])

# Create radar chart
angles = np.linspace(0, 2 * np.pi, len(labels), endpoint=False).tolist()
values = np.concatenate((values, [values[0]]))
angles += angles[:1]
fig, ax = plt.subplots(figsize=(6, 6), subplot_kw=dict(polar=True))
ax.fill(angles, values, color='blue', alpha=0.25)
ax.plot(angles, values, color='blue', linewidth=2)
ax.set_yticklabels([])
ax.set_xticks(angles[:-1])
ax.set_xticklabels(labels)
plt.title("Radar Chart")
plt.show()
```
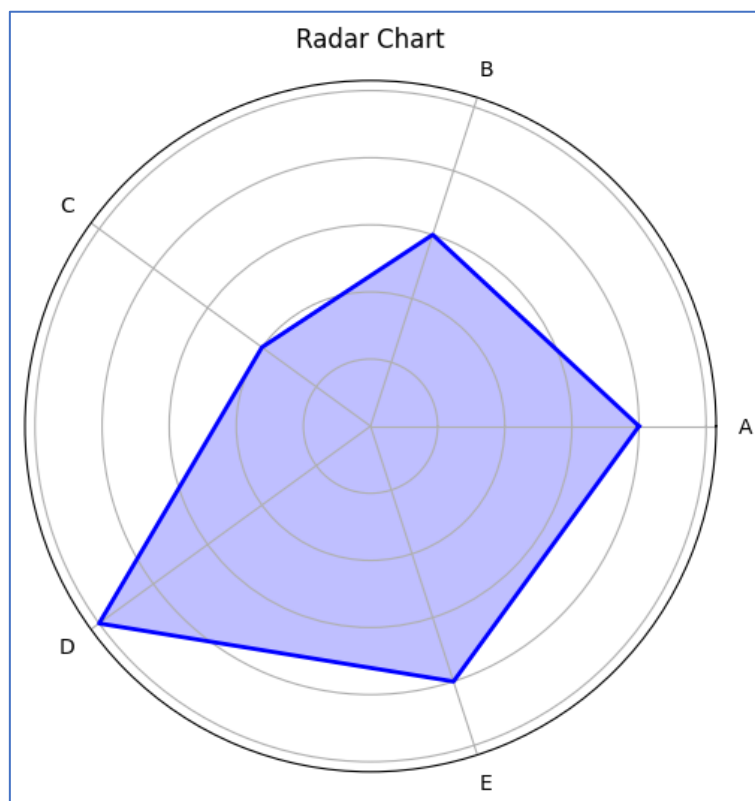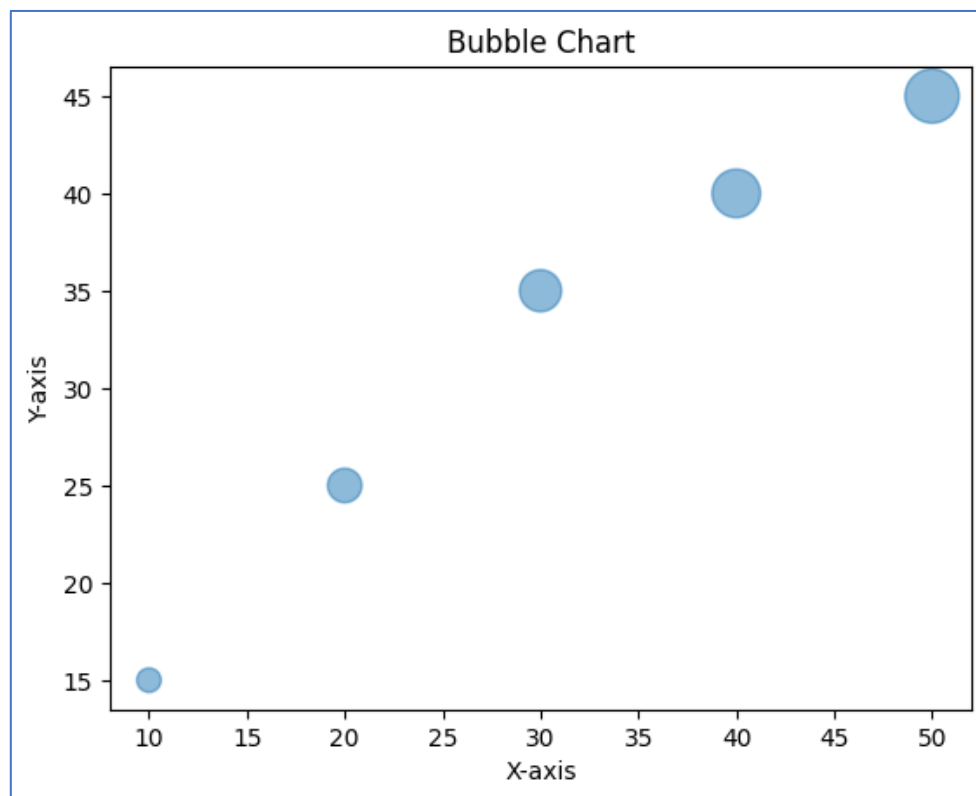
15. Bubble Chart

```python
import matplotlib.pyplot as plt

# Sample data
x = [10, 20, 30, 40, 50]
y = [15, 25, 35, 45, 55]
sizes = [100, 200, 300, 400, 500]
# Create bubble chart
plt.scatter(x, y, s=sizes, alpha=0.5)
plt.title("Bubble Chart")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.show()
```


These examples cover a wide range of common data visualizations. You can adjust the data and parameters to fit your specific needs. Each of these charts can be further customized with additional features, such as adding labels, changing colors, or adjusting the layout.