PYPLOT TUTORIAL

- Pyplot is a Matplotlib module that provides a MATLAB-like interface.
- Each pyplot function makes some changes to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc.

In [ ]:
```
import matplotlib.pyplot as plt
import numpy as np
```

In [ ]:
```
plt.figure(figsize=(4,3))
plt.plot([1, 2, 3, 4], [1, 4, 9, 16])
plt.axis([0, 6, 0, 20])
plt.show()
```

linear plotting

controlling line properties

In [ ]:
```
year = [1972, 1982, 1992, 2002, 2012]
india = [100.6, 158.61, 305.54, 394.96, 724.79]
bangladesh = [10.5, 25.21, 58.65, 119.27, 274.87]

plt.plot(year, india, color ='orange',label ='India',marker ='d', markersize =
plt.plot(year, bangladesh, color ='g', label ='Bangladesh',marker ='.', marker

# naming of x-axis and y-axis
plt.xlabel('Years')
plt.ylabel('Power consumption in kWh')

# naming the title of the plot
plt.title('Electricity consumption per capita of India and Bangladesh')

plt.legend()
plt.show()
```

scatter plot

In [ ]:
```
x = np.random.rand(50)
y = np.random.rand(50)
colors = np.random.rand(50)

plt.scatter(x, y, c=colors, cmap='viridis', s=100, alpha=0.8)
cbar = plt.colorbar()
cbar.set_label('Color Value')

plt.title('Scatter Plot with Color Mapping')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.show()
```

categorical variables

creating subplots

In [ ]:
```
names = ['a', 'b', 'c']
values = [20, 50, 100]
plt.figure(figsize=(9, 3))
plt.subplot(121)
plt.bar(names, values)
plt.subplot(122)
plt.scatter(names, values)


plt.suptitle('Categorical Plotting')
plt.show()
```

working with text

In [ ]:
```
mu, sigma = 100, 15
x = mu + sigma * np.random.randn(10000)

plt.figure(figsize=(4,4))

plt.hist(x, bins=50, color='skyblue', edgecolor='black')

plt.xlabel('IQ')
plt.ylabel('Probability')
plt.title('Histogram of IQ')
plt.text(40, 560, r'$\mu=100,\ \sigma=15$')
plt.show()
```

annotate text

In [ ]:
```
plt.figure(figsize=(4,3))
plt.plot([1, 2, 3, 4], [1, 4, 9, 16],marker='o')
plt.annotate('point', xy=(2, 3.5), xytext=(3, 3.4),arrowprops=dict(facecolor='
plt.show()
```

In [ ]:
```

```