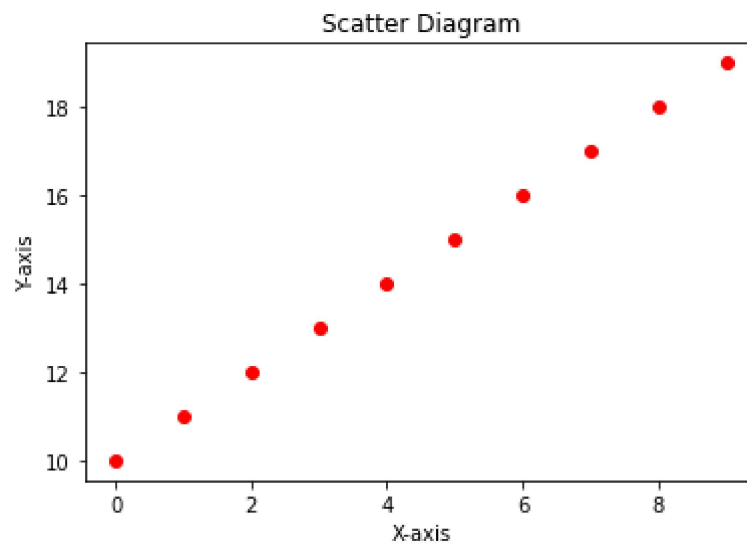


```
In [9]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

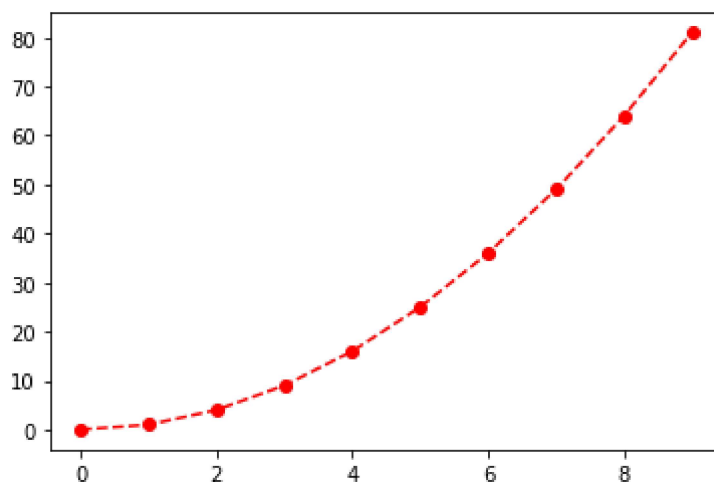
```
In [7]: x = np.arange(0,10)
y = np.arange(10,20)
```

```
In [12]: plt.scatter(x,y,c='r')
plt.title("Scatter Diagram")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.savefig("scatter.png")
```



```
In [17]: y = x**2
plt.plot(x,y,'go--',c='r')
```

Out[17]: [



## creating subplots

In [23]:

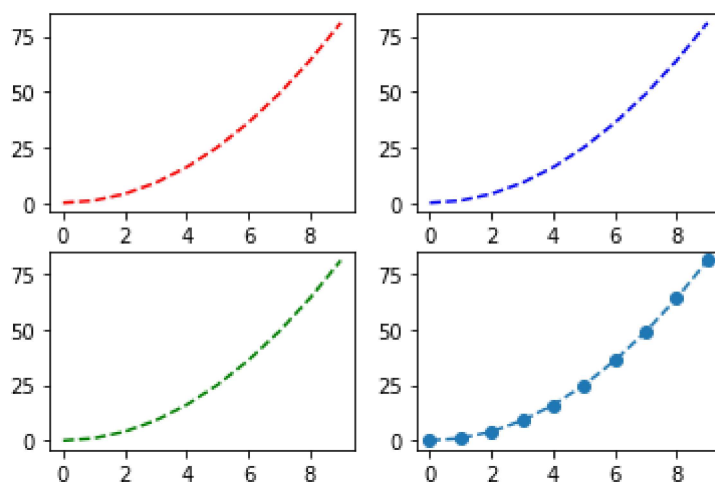
```
plt.subplot(2,2,1)
plt.plot(x,y, 'r--')

plt.subplot(2,2,2)
plt.plot(x,y, 'b--')

plt.subplot(2,2,3)
plt.plot(x,y, 'g--')

plt.subplot(2,2,4)
plt.plot(x,y, 'o--')
```

Out[23]: [



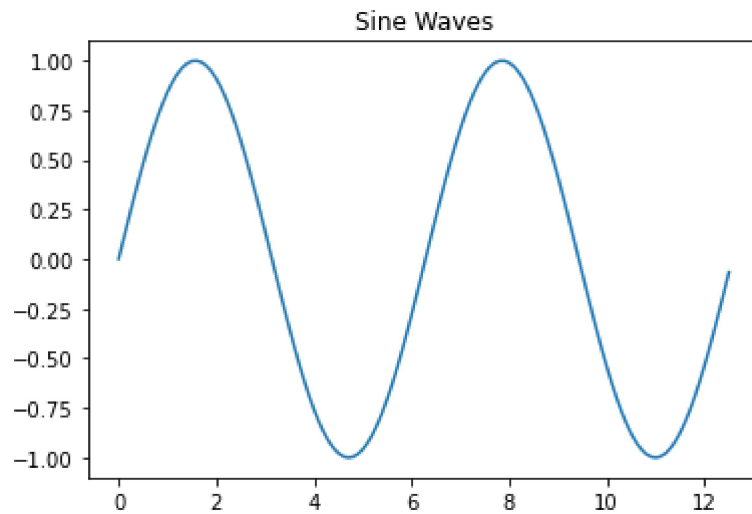
In [24]: np.pi

Out[24]: 3.141592653589793

***compute the x and y coordinates for points on a sine waves***

```
In [25]: x = np.arange(0,4 * np.pi, 0.1)
y = np.sin(x)
plt.title("Sine Waves")
plt.plot(x,y)
```

Out[25]: [<matplotlib.lines.Line2D at 0x12af7ae7220>]

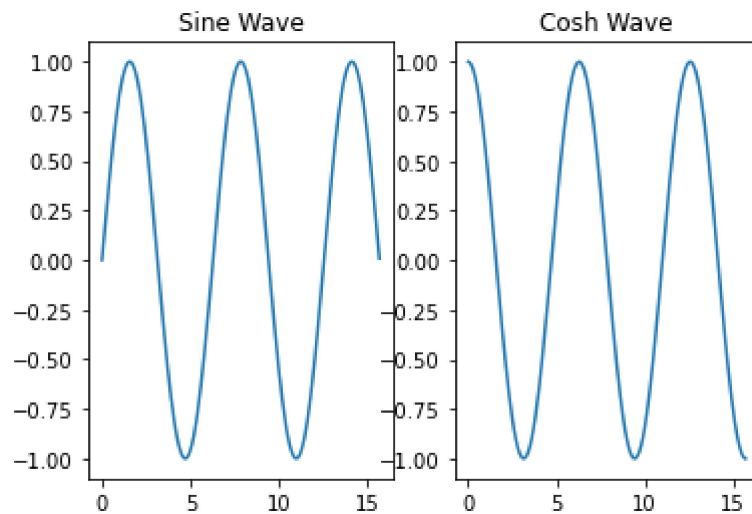


```
In [28]: x = np.arange(0,5*np.pi,0.1)
y_sin = np.sin(x)
y_cos = np.cos(x)

plt.subplot(1,2,1)
plt.plot(x,y_sin)
plt.title("Sine Wave")

plt.subplot(1,2,2)
plt.plot(x,y_cos)
plt.title("Cosh Wave")
```

Out[28]: Text(0.5, 1.0, 'Cosh Wave')



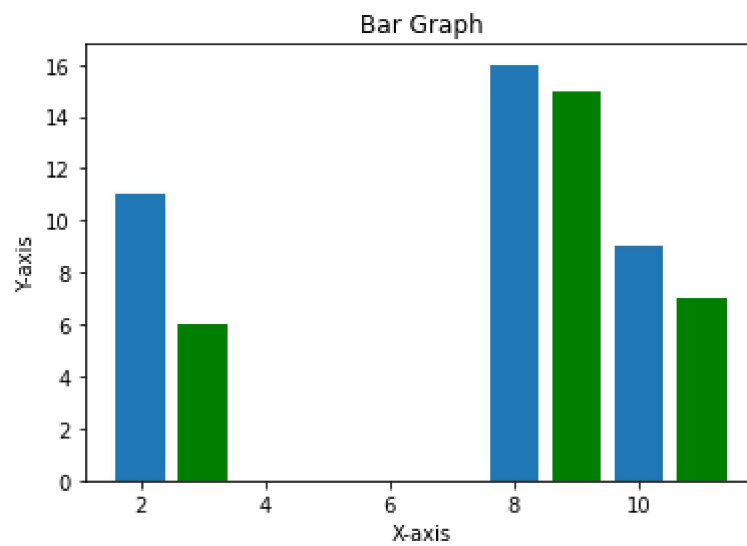
### ***Bar Graph***

```
In [29]: x = [2,8,10]
y = [11,16,9]

x2 = [3,9,11]
y2 = [6,15,7]

plt.bar(x,y)
plt.bar(x2,y2, color = 'g')
plt.title("Bar Graph")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
```

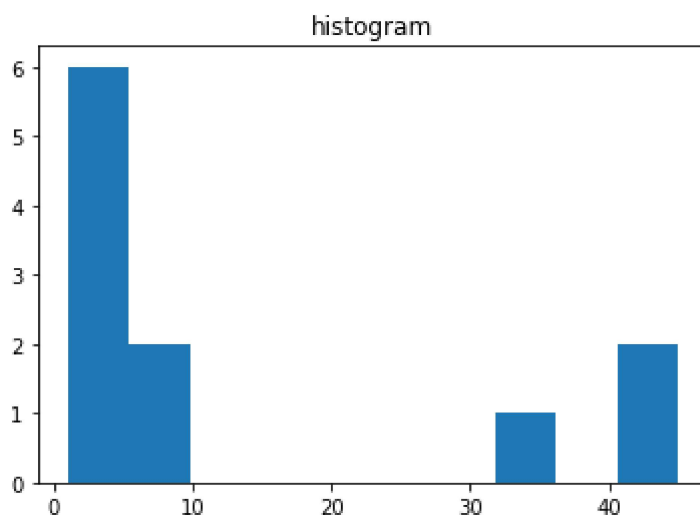
Out[29]: Text(0, 0.5, 'Y-axis')



### ***Histogram***

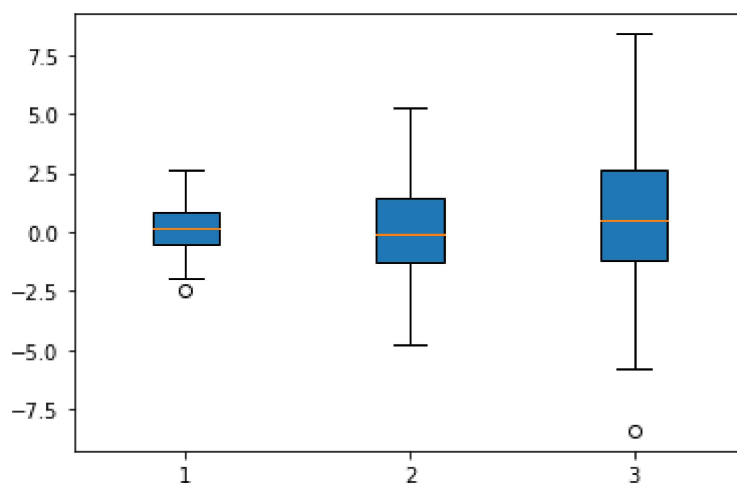
```
In [33]: ### It's specify the density or count of that plot  
a = np.array([1,2,3,44,5,6,7,4,33,45,5])  
plt.hist(a)  
plt.title("histogram")
```

Out[33]: Text(0.5, 1.0, 'histogram')



### Box Plot

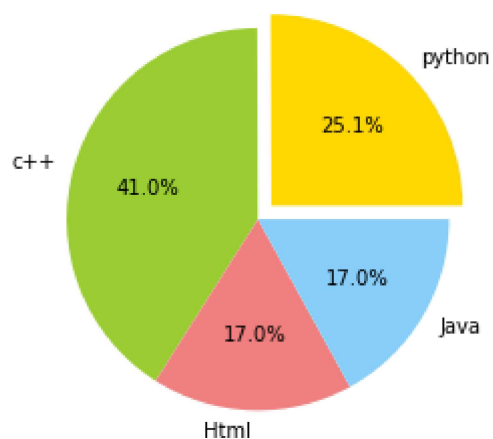
```
In [40]: data = [np.random.normal(0,std,100) for std in range(1,4)]  
plt.boxplot(data,vert = True,patch_artist = True);
```



### Pie Chart

```
In [52]: labels = 'python','c++','Html','Java'
colors = ['gold','yellowgreen','lightcoral','lightskyblue']
sizes = [345,564,234,234]
explode = (0.1,0,0,0)

plt.pie(sizes,explode = explode,labels = labels,colors = colors,
        autopct = '%1.1f%%',shadow = False)
plt.axis('equal')
plt.show()
```



In [ ]:

In [ ]:

In [ ]:

In [ ]: