In [18]:

```
import matplotlib.pyplot as plt

%matplotlib inline
```

In [19]:

```
import numpy as np
```

In [25]:

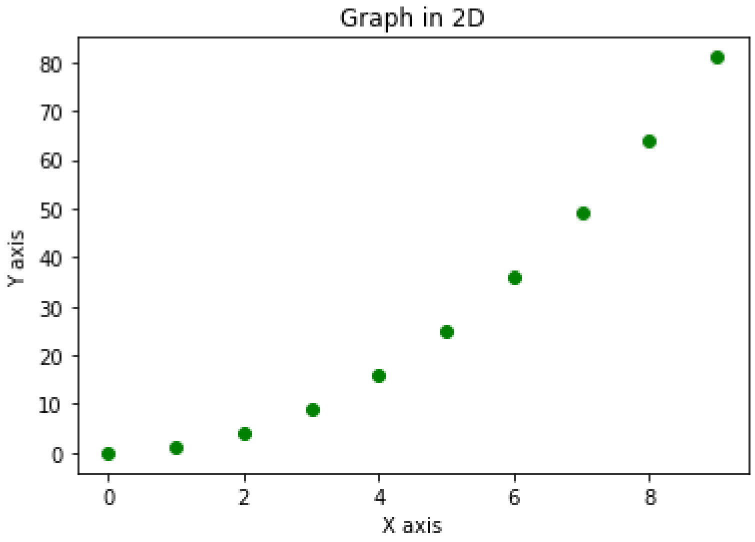
```
## Simple Examples

x=np.arange(0,10)
y=np.arange(11,21)
```

In [28]:

```
a=np.arange(40,50)
b=np.arange(50,60)
```

In [34]:

```
##plotting using matplotlib

##plt scatter

plt.scatter(x,y,c='g')
plt.xlabel('X axis')
plt.ylabel('Y axis')
plt.title('Graph in 2D')
plt.savefig('Test.png')
```



In [36]:

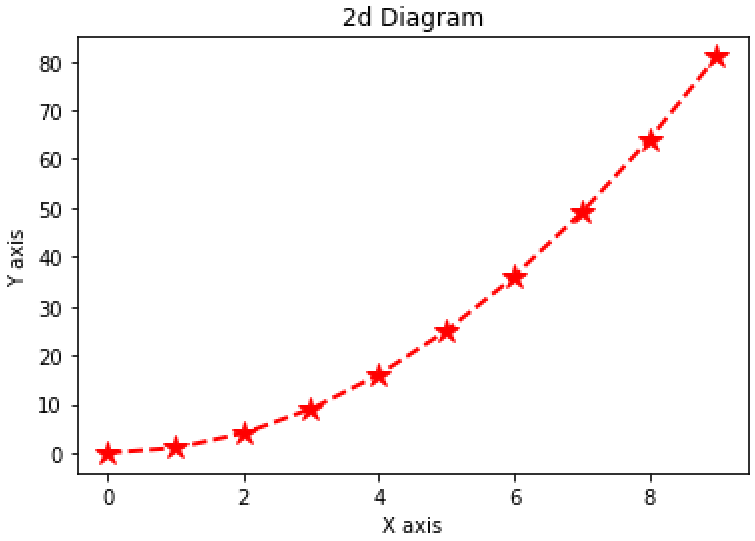
```
y=x*x
```

In [39]:

```
## plt plot

plt.plot(x,y,'r*',linestyle='dashed',linewidth=2, markersize=12)
plt.xlabel('X axis')
plt.ylabel('Y axis')
plt.title('2d Diagram')
```

Out[39]:

```
Text(0.5, 1.0, '2d Diagram')
```

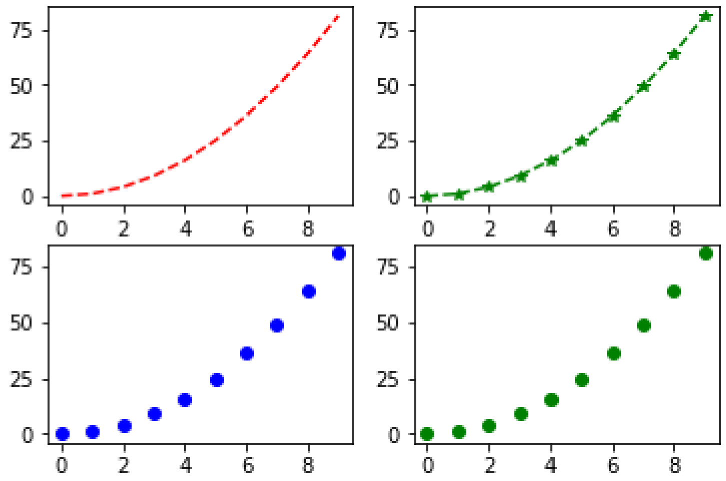


In [38]:

```
## Creating Subplots

plt.subplot(2,2,1)
plt.plot(x,y,'r--')
plt.subplot(2,2,2)
plt.plot(x,y,'g*--')
plt.subplot(2,2,3)
plt.plot(x,y,'bo')
plt.subplot(2,2,4)
plt.plot(x,y,'go')
```

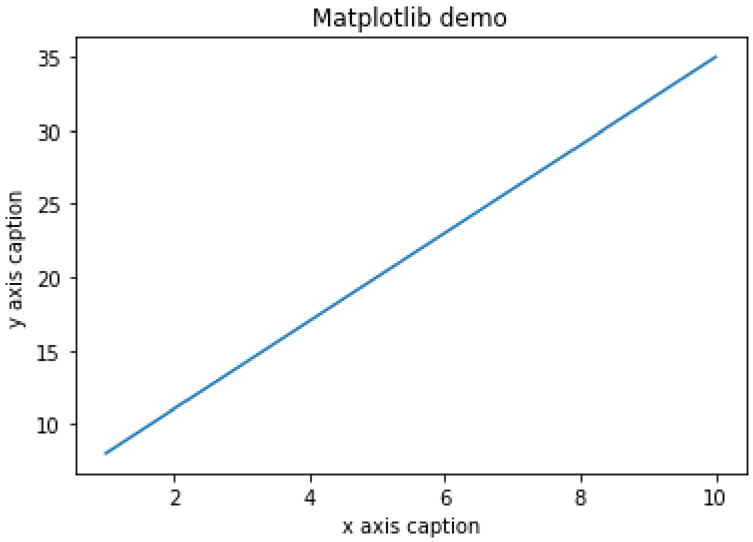
Out[38]:

```
[<matplotlib.lines.Line2D at 0x19f6e1d7630>]
```



In [40]:

```
x = np.arange(1,11)
y = 3 * x + 5
plt.title("Matplotlib demo")
plt.xlabel("x axis caption")
plt.ylabel("y axis caption")
plt.plot(x,y)
plt.show()
```



In [41]:

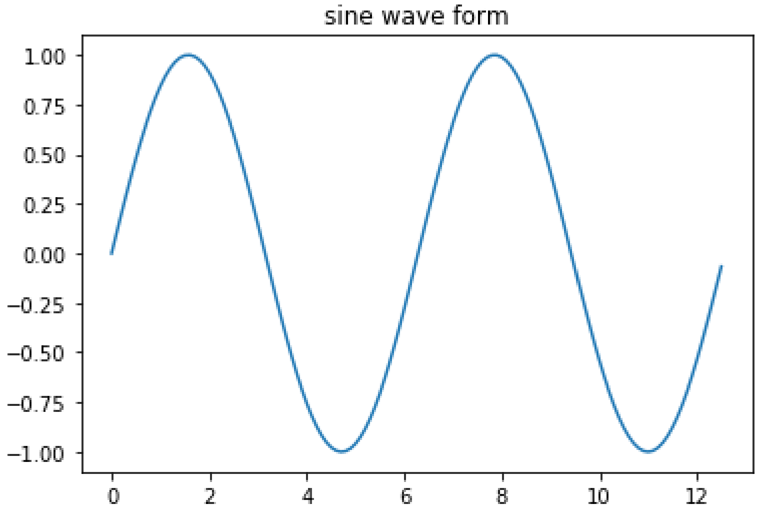
```
np.pi
```

Out[41]:

```
3.141592653589793
```

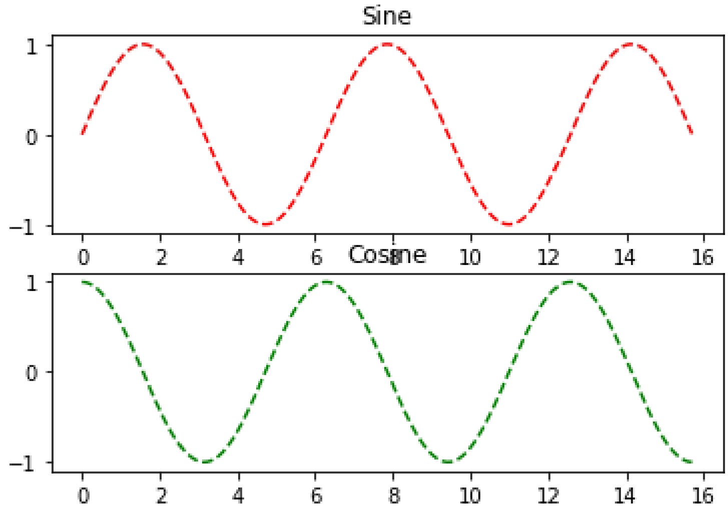
In [43]:

```
# Compute the x and y coordinates for points on a sine curve
x = np.arange(0, 4 * np.pi, 0.1)
y = np.sin(x)
plt.title("sine wave form")

# Plot the points using matplotlib
plt.plot(x, y)
plt.show()
```



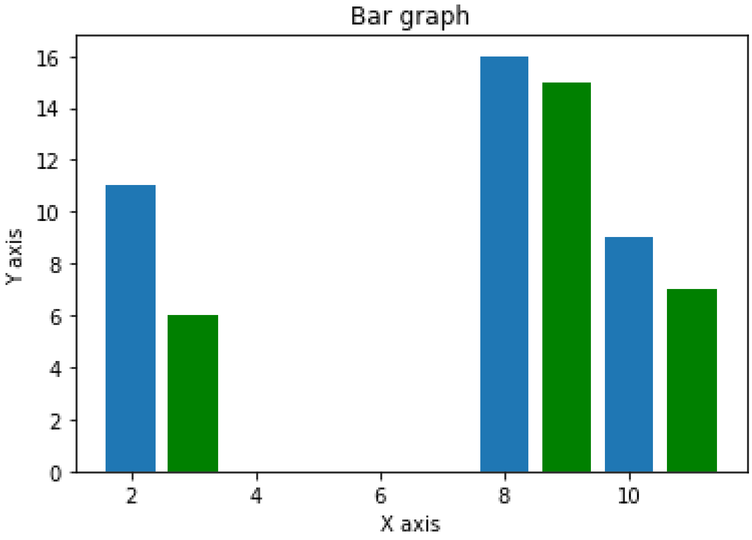
In [45]:

```
#Subplot()
# Compute the x and y coordinates for points on sine and cosine curves
x = np.arange(0, 5 * np.pi, 0.1)
y_sin = np.sin(x)
y_cos = np.cos(x)

# Set up a subplot grid that has height 2 and width 1,
# and set the first such subplot as active.
plt.subplot(2, 1, 1)

# Make the first plot
plt.plot(x, y_sin,'r--')
plt.title('Sine')

# Set the second subplot as active, and make the second plot.
plt.subplot(2, 1, 2)
plt.plot(x, y_cos,'g--')
plt.title('Cosine')

# Show the figure.
plt.show()
```



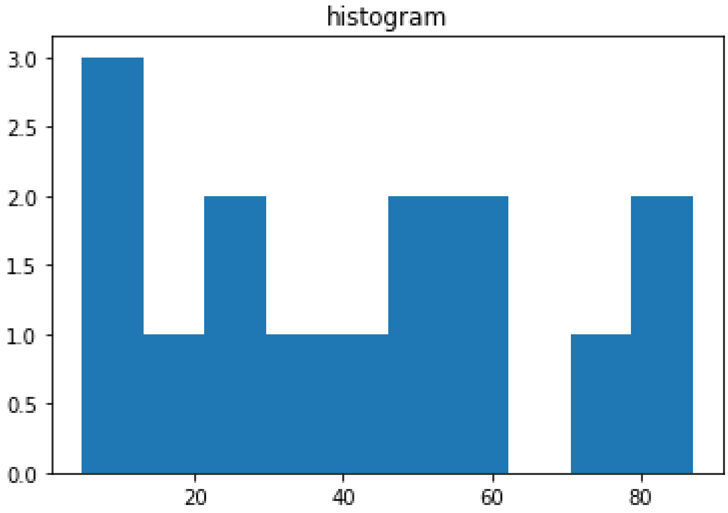
In [46]:

```
## Bar plot

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.ylabel('Y axis')
plt.xlabel('X axis')

plt.show()
```



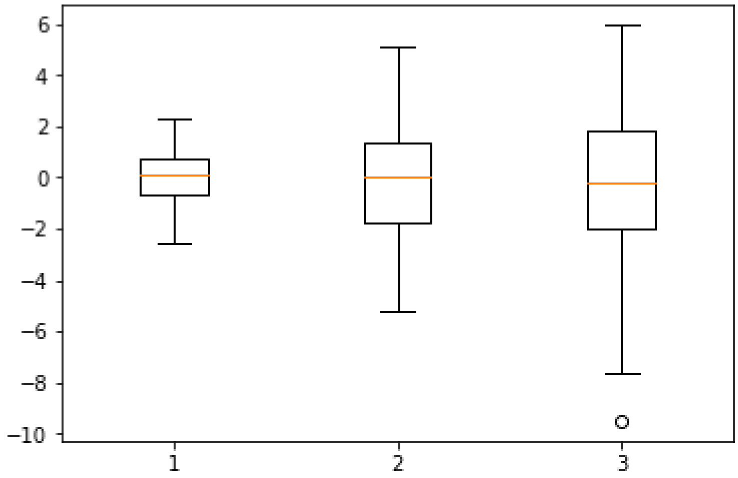
In [47]:

```
a = np.array([22,87,5,43,56,73,55,54,11,20,51,5,79,31,27])
plt.hist(a)
plt.title("histogram")
plt.show()
```



In [48]:

```
data = [np.random.normal(0, std, 100) for std in range(1, 4)]

# rectangular box plot
plt.boxplot(data,vert=True,patch_artist=False);
```



In [49]:

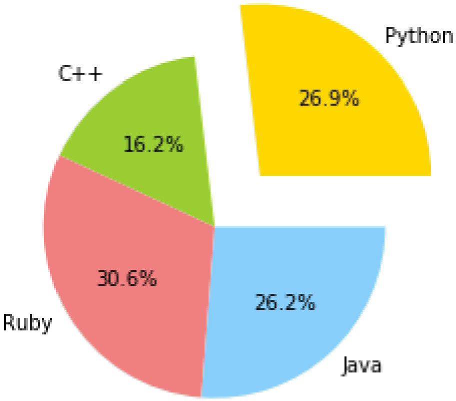
```
data
```

Out[49]:

```
[array([ 0.58504671, -2.42974343, -0.87807212,  0.71858855,  0.19451475,
         0.17303892, -0.84444114,  0.32612587,  0.86878209,  0.37730556,
         2.17469804,  1.63135053,  0.8395441 ,  0.62709484,  1.76115476,
         2.30605803,  0.072138  , -0.22507634,  0.59837975,  0.6184037 ,
        -0.29260248,  0.52244502, -1.41300711,  0.57073106, -0.21507045,
        -1.11923587,  0.43217588, -1.51230292,  1.23205926, -0.14142468,
        -0.87706184, -0.49979235,  1.19602235, -0.31594077,  0.56550357,
         0.72200673,  1.22648904,  0.63411471, -1.09226353,  0.1010199 ,
        -0.77885802, -0.35511938, -0.3756945 ,  0.54383162,  1.6731872 ,
        -0.70185931, -1.3670604 ,  0.19388628, -1.28266501, -0.04823673,
         0.10782148,  1.95815277,  0.68885592,  1.23793056, -2.51692068,
        -1.73765613,  0.76213279,  0.13265716,  1.21003138,  1.32695668,
        -1.63716946, -2.10970485, -0.46485711,  1.61054418,  1.24012637,
         0.16259744,  1.96779766,  2.09128109,  1.00001079, -0.22335079,
        -0.51816058,  0.66126304,  0.23451894, -0.43515676,  1.53187283,
        -0.55618087, -0.81812007,  0.53326243,  1.97731335,  0.19072302,
        -0.10221549,  0.09176608, -0.21815642, -1.14878973, -1.60459598,
         1.04666692, -0.8981098 , -0.20830587, -1.02803627, -0.64417093,
        -0.72172321, -0.45508901, -0.87384667,  0.37577657, -0.67601576,
         0.33178059, -2.26430786, -0.24216645,  1.51467658, -0.68156033]),
 array([ 2.39757262,  2.74399968, -0.75989482, -4.26114517, -1.80110566,
         0.57480583,  1.33483028, -2.49762657, -1.44751213,  2.81140914,
         1.8292919 , -1.15478458,  4.10877717,  0.89963471, -0.29582823,
        -4.38903595, -0.38549566, -3.60851726, -0.22060642, -2.03192216,
         3.89103451,  4.12598922,  1.75938297, -1.89828706,  1.39755224,
         0.58455399,  0.11512576,  3.56831907, -2.7059527 ,  1.61987152,
         2.24961343, -0.62407822,  2.70878929, -0.18364358,  0.3782594 ,
        -1.80163772, -2.13304259, -0.11028747, -0.53854786,  0.48576546,
        -2.39393639, -3.3118441 , -2.39628927, -3.1162055 ,  1.04191607,
        -3.64948397, -3.58266108,  1.4339995 ,  0.90005896, -1.78775946,
        -1.51791255, -0.58054874,  0.18305238,  0.36569407, -0.40180358,
        -1.94481004,  5.08736186,  1.16086626, -2.31853969,  3.11128484,
         0.46886757, -1.80233453,  0.19754135,  1.92842538, -4.74937684,
        -2.16864212,  1.99652951, -0.01193476,  1.23598666,  0.44478576,
         1.00272555, -0.02143306, -1.70422643,  4.60558454,  3.1843202 ,
        -1.75649444,  1.17151828,  0.00895449, -1.60152541,  0.72525242,
         4.18490847, -0.15177633,  0.61553151,  1.27330578, -1.08311012,
         0.42874645, -3.39590889,  0.68195592, -1.22643768, -0.60238626,
        -0.9784442 ,  1.41962007, -5.21125433, -4.56646962,  0.20156596,
         2.13782937, -0.53930096, -2.23339868,  1.85651077,  1.58838732]),
 array([-0.46813528, -5.39188897, -0.35860612,  3.16538809, -4.49522574,
        -3.84458331, -3.37315175, -1.19829063,  0.65034886,  0.08958735,
        -3.30533751, -4.02608506, -1.33965818,  0.08521081,  5.95907907,
        -1.49659721, -2.63732646,  0.63657732,  3.10515928, -0.63876556,
         3.49367813,  0.91798042, -2.43299596, -0.17403589,  3.77718343,
         4.17885917, -2.79131688,  0.88634798,  0.23514888, -3.28044413,
        -0.67618458, -0.73950108,  0.58727282, -2.66695856,  0.97577819,
         2.71467279, -2.84248337,  2.64707155, -2.08106232, -5.35279904,
         3.73711532, -1.97505552,  1.09767953, -1.19807206,  5.43153888,
        -1.27109587,  1.93009506, -1.8568635 , -0.17060989, -9.50706159,
         3.50143286, -1.27320584,  1.81000529,  1.00959187, -4.28321702,
        -2.01591079,  3.42278899, -7.61326112, -0.67751191,  2.50867343,
        -2.46062236,  2.45095904, -1.71633283,  1.35203982, -5.16007168,
         2.69362946,  0.35902891,  5.00972343,  1.7843087 ,  1.89508631,
        -0.43550184, -4.00886466,  2.37549762,  2.19215615,  1.04445794,
         0.41093029,  1.74624972, -1.68477072, -0.34322167, -1.48703608,
        -2.72498388,  0.46554118,  0.43959455,  1.78344169, -0.31535279,
         3.30285073, -1.98289587, -3.03499834,  4.52372711, -0.64513321,
        -1.04974701, -5.12679527,  2.35839539,  1.48345365,  1.95883738,
        -5.22536477,  1.30151887, -0.11828971, -3.81039229,  3.93224803])]
```

In [50]:

```
# Data to plot
labels = 'Python', 'C++', 'Ruby', 'Java'
sizes = [215, 130, 245, 210]
colors = ['gold', 'yellowgreen', 'lightcoral', 'lightskyblue']
explode = (0.4, 0, 0, 0)  # explode 1st slice

# Plot
plt.pie(sizes, explode=explode, labels=labels, colors=colors,
autopct='%1.1f%%', shadow=False)

plt.axis('equal')
plt.show()
```



In [ ]: