Matplot lib is one of the most popular Python Package used for data visualization. We can make cross platform 2D Plots from the data in array. We can export visualization to all the common formats such as PNG,JPG,PDF, GIF etc.. Current Matplotlib supports 3D Plotting as well. It was created by John Hunter in the year 2002

In [ ]:

```
1
```

pip install matplotlib ==> For installing Matplot lib Library

In [ ]:

```
1
```

Types of Plots

```
1  **Function**                                  **Description**
2
3  Bar                                           Makes a Bar Plot
4  Barh                                          Makes a horizontal Bar Plot
5  BoxPlot                                       Makes a Box and a Whisker Plot
6  Hist                                          Plots a histogram
7  hist2d                                        Makes a 2d Histogram Plot
8  Pie                                           Plots a Pie Chart
9  Plot                                          Plots a line on the axes
10 Polar                                         Makes a Polar Plot
11 Scatter                                       Makes a Scatter Plot of x vs y
12 StackPlot                                     Draw a Stacked area plot
13 Stem                                          Create a Stem Plot
14 Step                                          Create a Step Plot
```

In [ ]:

```
1
```

Axis Function

```
1  Axes                                  Adds axes to the figure
2  Text                                  Add text to the axes
3  Title                                 Set the Title of the current axes
4  Xlabel                                Set the x axis label
5  Xlim                                  Set the limit of current x axis
6  Xscale                                Set the scaling of the x-axis
7  Xticks                                Get or Set the limit of the current tick location and labels
8  Ylabel                                Set the y axis label of the current y - axis
9  Ylim                                  Set the limit of current Y axis
10 Yscale                                Set the scaling of the Y-axis
11 Yticks                                Get or set the y-limit of the current tick location and labels
```

In [ ]:

```
1
```

Working with matplot lib

In [5]:

```
1 import numpy as np
2 import pandas as pd
```

In [6]:

```
1 import matplotlib.pyplot as plt
```

In [7]:

```
1 x1
```

Out[7]:

```
[1, 16, 30, 42, 55, 68, 77, 88]
```

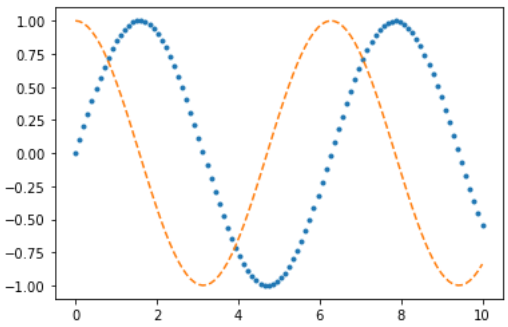
A Figure object is the outermost container for a Matplotlib Plot. The figure object contains multiple axes objects. The figure is the final graphic that may contain one or more than one axes. The axes represent an individual plot

In [8]:

```
x1 = np.linspace(0,10,100)
fig = plt.figure()
plt.plot(x1,np.sin(x1),'.')
plt.plot(x1,np.cos(x1),'--')
```

Out[8]:

```
[<matplotlib.lines.Line2D at 0x7fa1c0648310>]
```



In [9]:

```
x1
```

Out[9]:

```
array([ 0.        ,  0.1010101 ,  0.2020202 ,  0.3030303 ,  0.4040404 ,
        0.50505051,  0.60606061,  0.70707071,  0.80808081,  0.90909091,
        1.01010101,  1.11111111,  1.21212121,  1.31313131,  1.41414141,
        1.51515152,  1.61616162,  1.71717172,  1.81818182,  1.91919192,
        2.02020202,  2.12121212,  2.22222222,  2.32323232,  2.42424242,
        2.52525253,  2.62626263,  2.72727273,  2.82828283,  2.92929293,
        3.03030303,  3.13131313,  3.23232323,  3.33333333,  3.43434343,
        3.53535354,  3.63636364,  3.73737374,  3.83838384,  3.93939394,
        4.04040404,  4.14141414,  4.24242424,  4.34343434,  4.44444444,
        4.54545455,  4.64646465,  4.74747475,  4.84848485,  4.94949495,
        5.05050505,  5.15151515,  5.25252525,  5.35353535,  5.45454545,
        5.55555556,  5.65656566,  5.75757576,  5.85858586,  5.95959596,
        6.06060606,  6.16161616,  6.26262626,  6.36363636,  6.46464646,
        6.56565657,  6.66666667,  6.76767677,  6.86868687,  6.96969697,
        7.07070707,  7.17171717,  7.27272727,  7.37373737,  7.47474747,
        7.57575758,  7.67676768,  7.77777778,  7.87878788,  7.97979798,
        8.08080808,  8.18181818,  8.28282828,  8.38383838,  8.48484848,
        8.58585859,  8.68686869,  8.78787879,  8.88888889,  8.98989899,
        9.09090909,  9.19191919,  9.29292929,  9.39393939,  9.49494949,
        9.5959596 ,  9.6969697 ,  9.7979798 ,  9.8989899 , 10.        ])
```

In [ ]:

```

```

In [ ]:

```

```

In [10]:

```
import math
x = np.arange(0,math.pi*2,0.05)
```

In [11]:

```
x
```

Out[11]:

```
array([0.  , 0.05, 0.1 , 0.15, 0.2 , 0.25, 0.3 , 0.35, 0.4 , 0.45, 0.5 ,
       0.55, 0.6 , 0.65, 0.7 , 0.75, 0.8 , 0.85, 0.9 , 0.95, 1.  , 1.05,
       1.1 , 1.15, 1.2 , 1.25, 1.3 , 1.35, 1.4 , 1.45, 1.5 , 1.55, 1.6 ,
       1.65, 1.7 , 1.75, 1.8 , 1.85, 1.9 , 1.95, 2.  , 2.05, 2.1 , 2.15,
       2.2 , 2.25, 2.3 , 2.35, 2.4 , 2.45, 2.5 , 2.55, 2.6 , 2.65, 2.7 ,
       2.75, 2.8 , 2.85, 2.9 , 2.95, 3.  , 3.05, 3.1 , 3.15, 3.2 , 3.25,
       3.3 , 3.35, 3.4 , 3.45, 3.5 , 3.55, 3.6 , 3.65, 3.7 , 3.75, 3.8 ,
       3.85, 3.9 , 3.95, 4.  , 4.05, 4.1 , 4.15, 4.2 , 4.25, 4.3 , 4.35,
       4.4 , 4.45, 4.5 , 4.55, 4.6 , 4.65, 4.7 , 4.75, 4.8 , 4.85, 4.9 ,
       4.95, 5.  , 5.05, 5.1 , 5.15, 5.2 , 5.25, 5.3 , 5.35, 5.4 , 5.45,
       5.5 , 5.55, 5.6 , 5.65, 5.7 , 5.75, 5.8 , 5.85, 5.9 , 5.95, 6.  ,
       6.05, 6.1 , 6.15, 6.2 , 6.25])
```

In [12]:

```
y = np.sin(x)
```

In [13]:

```
1 y
```

Out[13]:

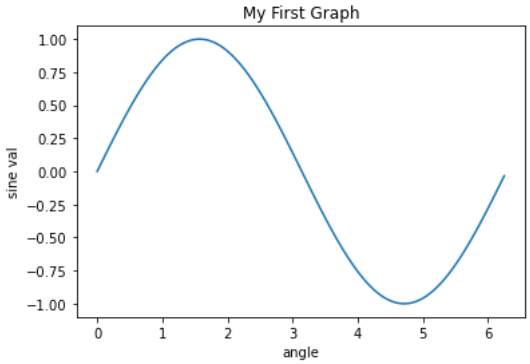
```
array([ 0.        ,  0.04997917,  0.09983342,  0.14943813,  0.19866933,
        0.24740396,  0.29552021,  0.34289781,  0.38941834,  0.43496553,
        0.47942554,  0.52268723,  0.56464247,  0.60518641,  0.64421769,
        0.68163876,  0.71735609,  0.75128041,  0.78332691,  0.8134155 ,
        0.84147098,  0.86742323,  0.89120736,  0.91276394,  0.93203909,
        0.94898462,  0.96355819,  0.97572336,  0.98544973,  0.99271299,
        0.99749499,  0.99978376,  0.9995736 ,  0.99686503,  0.99166481,
        0.98398595,  0.97384763,  0.9612752 ,  0.94630009,  0.92895972,
        0.90929743,  0.88736237,  0.86320937,  0.83689879,  0.8084964 ,
        0.7780732 ,  0.74570521,  0.71147335,  0.67546318,  0.6377647 ,
        0.59847214,  0.55768372,  0.51550137,  0.47203054,  0.42737988,
        0.38166099,  0.33498815,  0.28747801,  0.23924933,  0.19042265,
        0.14112001,  0.09146464,  0.04158066, -0.00840725, -0.05837414,
       -0.10819513, -0.15774569, -0.20690197, -0.2555411 , -0.30354151,
       -0.35078323, -0.39714817, -0.44252044, -0.48678665, -0.52983614,
       -0.57156132, -0.61185789, -0.65062514, -0.68776616, -0.72318812,
       -0.7568025 , -0.78852525, -0.81827711, -0.8459837 , -0.87157577,
       -0.89498936, -0.91616594, -0.93505258, -0.95160207, -0.96577306,
       -0.97753012, -0.98684386, -0.993691  , -0.99805444, -0.99992326,
       -0.99929279, -0.99616461, -0.99054654, -0.98245261, -0.97190307,
       -0.95892427, -0.94354867, -0.92581468, -0.90576664, -0.88345466,
       -0.85893449, -0.83226744, -0.80352016, -0.77276449, -0.74007731,
       -0.70554033, -0.66923986, -0.63126664, -0.59171558, -0.55068554,
       -0.50827908, -0.46460218, -0.41976402, -0.37387666, -0.32705481,
       -0.2794155 , -0.23107779, -0.1821625 , -0.13279191, -0.0830894 ,
       -0.03317922])
```

In [14]:

```
1 plt.plot(x,y)
2 plt.xlabel("angle")
3 plt.ylabel("sine val")
4 plt.title("My First Graph")
```

Out[14]:

```
Text(0.5, 1.0, 'My First Graph')
```



In [ ]:

```
1
```

In [ ]:

```
1
```

In [15]:

```
1 x = np.linspace(-3,3,30)
2 x
```

Out[15]:

```
array([-3.        , -2.79310345, -2.5862069 , -2.37931034, -2.17241379,
       -1.96551724, -1.75862069, -1.55172414, -1.34482759, -1.13793103,
       -0.93103448, -0.72413793, -0.51724138, -0.31034483, -0.10344828,
        0.10344828,  0.31034483,  0.51724138,  0.72413793,  0.93103448,
        1.13793103,  1.34482759,  1.55172414,  1.75862069,  1.96551724,
        2.17241379,  2.37931034,  2.5862069 ,  2.79310345,  3.        ])
```

In [16]:

```
1 y = x**2
```

In [17]:

```
y
```

Out[17]:

```
array([9.        , 7.80142687, 6.68846611, 5.66111772, 4.71938169,
       3.86325803, 3.09274673, 2.4078478 , 1.80856124, 1.29488704,
       0.86682521, 0.52437574, 0.26753864, 0.09631391, 0.01070155,
       0.01070155, 0.09631391, 0.26753864, 0.52437574, 0.86682521,
       1.29488704, 1.80856124, 2.4078478 , 3.09274673, 3.86325803,
       4.71938169, 5.66111772, 6.68846611, 7.80142687, 9.        ])
```

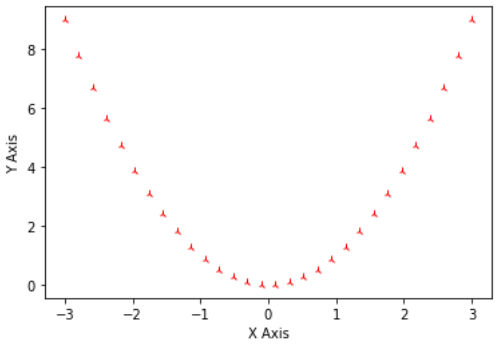
```
color code: b,g,r,c,m,y,k,w

designs :

-
--
-.
,
o
^
<
>
s
+
x
D
d
1
2
3
4
h
H
p
|
_
```

In [18]:

```
plt.plot(x,y,'r2')
plt.xlabel("X Axis")
plt.ylabel("Y Axis")
plt.show()
```



In [19]:

```
x1
```

Out[19]:

```
array([ 0.        ,  0.1010101 ,  0.2020202 ,  0.3030303 ,  0.4040404 ,
        0.50505051,  0.60606061,  0.70707071,  0.80808081,  0.90909091,
        1.01010101,  1.11111111,  1.21212121,  1.31313131,  1.41414141,
        1.51515152,  1.61616162,  1.71717172,  1.81818182,  1.91919192,
        2.02020202,  2.12121212,  2.22222222,  2.32323232,  2.42424242,
        2.52525253,  2.62626263,  2.72727273,  2.82828283,  2.92929293,
        3.03030303,  3.13131313,  3.23232323,  3.33333333,  3.43434343,
        3.53535354,  3.63636364,  3.73737374,  3.83838384,  3.93939394,
        4.04040404,  4.14141414,  4.24242424,  4.34343434,  4.44444444,
        4.54545455,  4.64646465,  4.74747475,  4.84848485,  4.94949495,
        5.05050505,  5.15151515,  5.25252525,  5.35353535,  5.45454545,
        5.55555556,  5.65656566,  5.75757576,  5.85858586,  5.95959596,
        6.06060606,  6.16161616,  6.26262626,  6.36363636,  6.46464646,
        6.56565657,  6.66666667,  6.76767677,  6.86868687,  6.96969697,
        7.07070707,  7.17171717,  7.27272727,  7.37373737,  7.47474747,
        7.57575758,  7.67676768,  7.77777778,  7.87878788,  7.97979798,
        8.08080808,  8.18181818,  8.28282828,  8.38383838,  8.48484848,
        8.58585859,  8.68686869,  8.78787879,  8.88888889,  8.98989899,
        9.09090909,  9.19191919,  9.29292929,  9.39393939,  9.49494949,
        9.5959596 ,  9.6969697 ,  9.7979798 ,  9.8989899 , 10.        ])
```

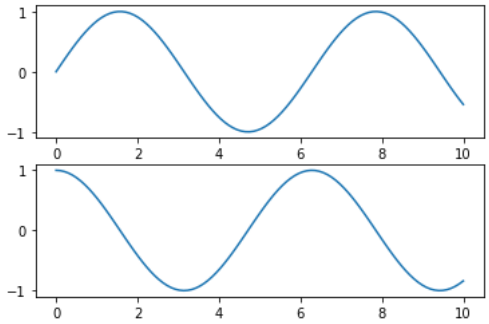
In [20]:

```
plt.figure()
plt.subplot(2,1,1)
plt.plot(x1,np.sin(x1))

plt.subplot(2,1,2)
plt.plot(x1,np.cos(x1))
```

Out[20]:

```
[<matplotlib.lines.Line2D at 0x7fa1c0495820>]
```



In [ ]:

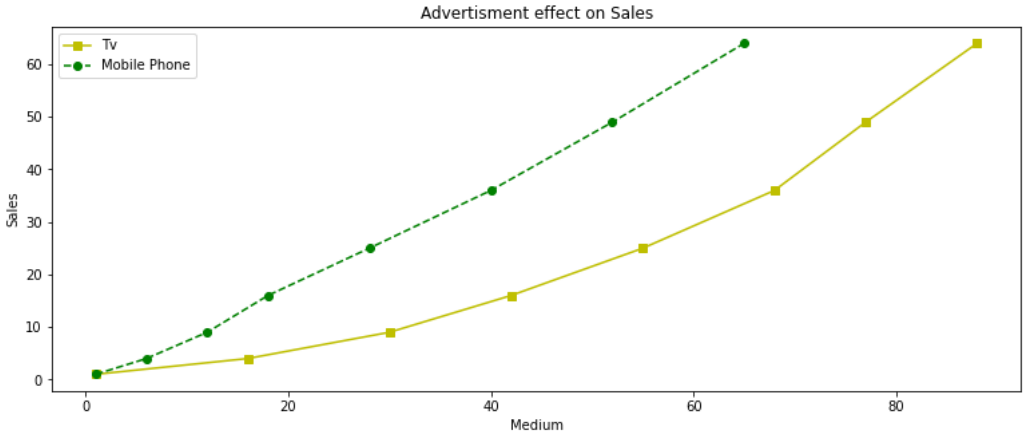
```

```

In [21]:

```
y = [1,4,9,16,25,36,49,64]
x1 = [1,16,30,42,55,68,77,88]
x2 = [1,6,12,18,28,40,52,65]

f = plt.figure()
f.set_figwidth(13)
f.set_figheight(5)
plt.plot(x1,y,'ys-')
plt.plot(x2,y,'go--')
plt.legend(['Tv','Mobile Phone'])
plt.title("Advertisment effect on Sales")
plt.xlabel('Medium')
plt.ylabel('Sales')
plt.show()
f.savefig('Tv.png')
```



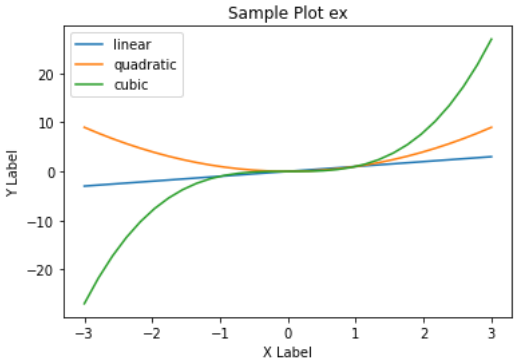
In [ ]:

```

```

In [22]:

```
x_data = np.linspace(0,2,100)
plt.plot(x,x,label='linear')
plt.plot(x,x**2,label='quadratic')
plt.plot(x,x**3,label='cubic')
plt.xlabel('X Label')
plt.ylabel('Y Label')
plt.title('Sample Plot ex')
plt.legend()
plt.show()
```

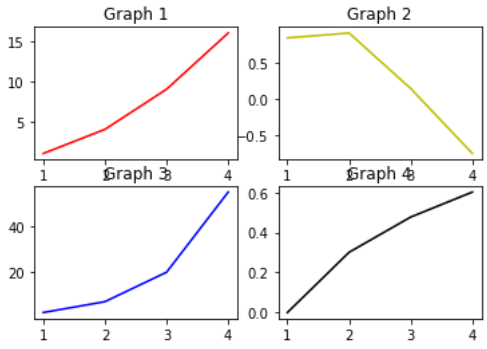


In [24]:

```
x = np.arange(1,5)
```

In [37]:

```
fig = plt.figure()
ax1 = fig.add_subplot(2,2,1)
ax1.plot(x,x*x,'r')
ax1.set_title('Graph 1')
ax2 = fig.add_subplot(2,2,2)
ax2.plot(x,np.sin(x),'y')
ax2.set_title('Graph 2')
ax3 = fig.add_subplot(2,2,3)
ax3.plot(x,np.exp(x),'b')
ax3.set_title('Graph 3')
ax4 = fig.add_subplot(2,2,4)
ax4.plot(x,np.log10(x),'k')
ax4.set_title('Graph 4')
plt.show()
```



In [ ]:

```
```

In [38]:

```
data = np.random.randn(1000)
```

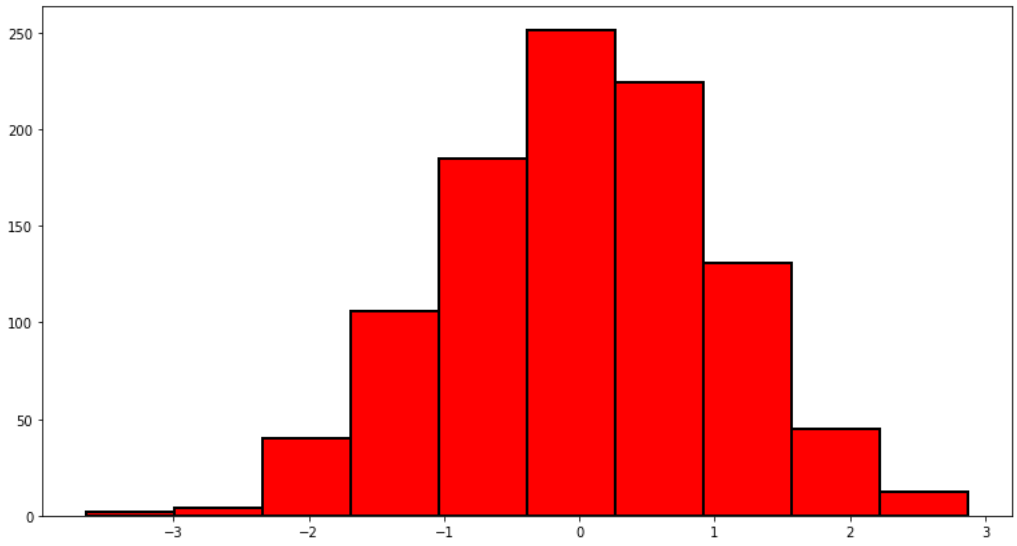
In [39]:

```
data
```

```
       1.67208056e+00, -3.26150639e-01, -4.44936689e-05, -4.55246167e-01,
       1.14643915e-01,  1.49293845e-01, -9.73588701e-02,  2.09076913e+00,
       1.61003031e+00, -2.10409562e-01, -1.83898068e+00, -6.34086811e-01,
       9.70834706e-01,  2.94271014e-01, -4.58200391e-01,  4.09216879e-01,
       1.58042204e-01,  6.25805719e-01, -3.64081863e+00, -8.81453862e-01,
       3.76019702e-01, -2.21509088e-01,  1.54847007e+00, -8.70684691e-01,
       3.67098315e-02, -4.44557365e-01,  1.01952159e+00, -1.23221091e+00,
      -1.15713364e+00,  1.27407693e-01, -9.11653648e-01, -1.01971131e+00,
       7.85486380e-01,  2.41983042e-01, -7.59118032e-01, -1.15313596e+00,
      -1.76948249e+00, -4.17440539e-02, -4.79326102e-01,  6.87477019e-01,
       4.30449464e-01, -4.55072751e-01, -2.83898852e-01,  9.75948028e-01,
      -2.95740169e-01, -1.52768459e-01, -5.41869566e-01,  3.47099484e-01,
       1.00544443e+00,  5.26650123e-01, -1.20023341e+00, -3.68488879e-01,
      -8.20281339e-01,  1.29028442e+00,  6.01636207e-01,  3.96323495e-01,
      -2.55377508e-01, -5.24954924e-01, -1.18963125e+00, -5.20702724e-01,
       2.71372260e-01,  3.97639127e-01, -1.74663806e+00, -1.60081276e-01,
      -7.70146711e-01,  2.83172691e-02, -9.12854825e-01, -1.00608801e+00,
       1.28735307e+00, -3.29410700e-01, -1.01703528e+00,  5.97721383e-01,
       4.44837082e-01, -8.82246436e-02,  1.77375843e+00,  2.01783918e+00,
       5.23422314e-01,  1.87139502e+00,  3.93856586e-01,  1.22477305e+00,
```

In [46]:

```
f = plt.figure()
f.set_figwidth(13)
f.set_figheight(7)
plt.hist(data,color='red',edgecolor='black',linewidth=2)
plt.show()
```



In [ ]:

```

```