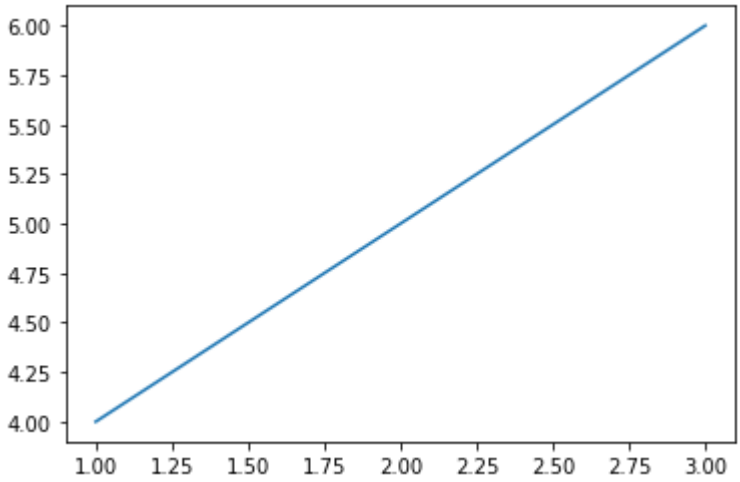
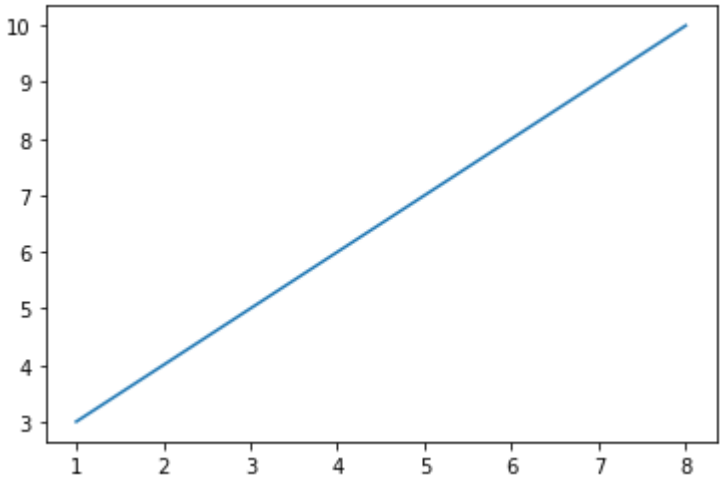
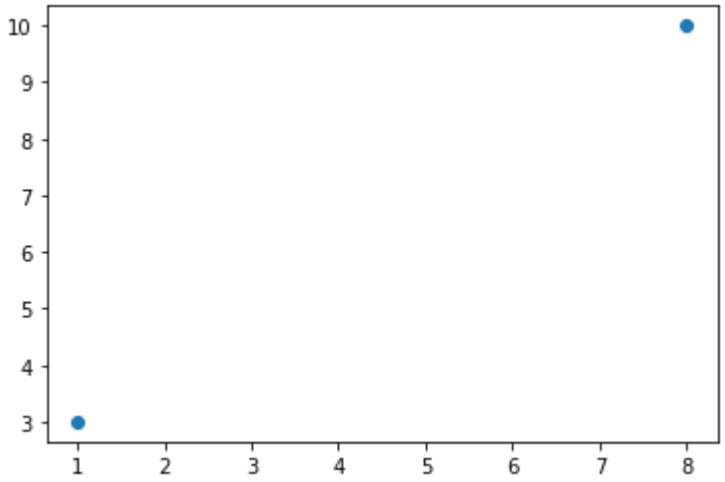
In [68]:
```
import matplotlib.pyplot as plt
import numpy as np
plt.plot([1,2,3],[4,5,6])
plt.show()
```



In [3]:
```
xpoints = np.array([1, 8])
ypoints = np.array([3, 10])

plt.plot(xpoints, ypoints)
plt.show()
```

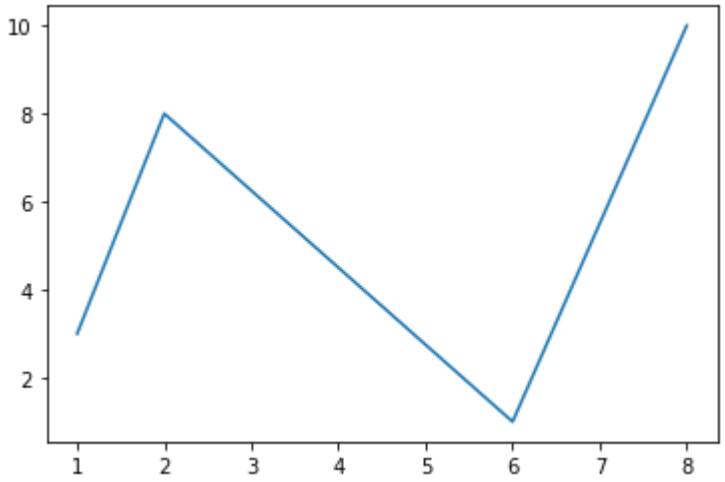


In [4]:
```
xpoints = np.array([1, 8])
ypoints = np.array([3, 10])

plt.plot(xpoints, ypoints,'o')
plt.show()
```



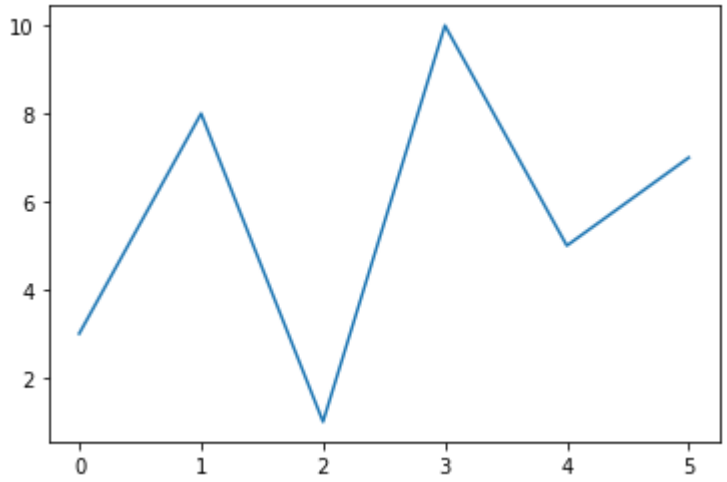
In [5]:

```
xpoints = np.array([1, 2, 6, 8])
ypoints = np.array([3, 8, 1, 10])

plt.plot(xpoints, ypoints)
plt.show()
```



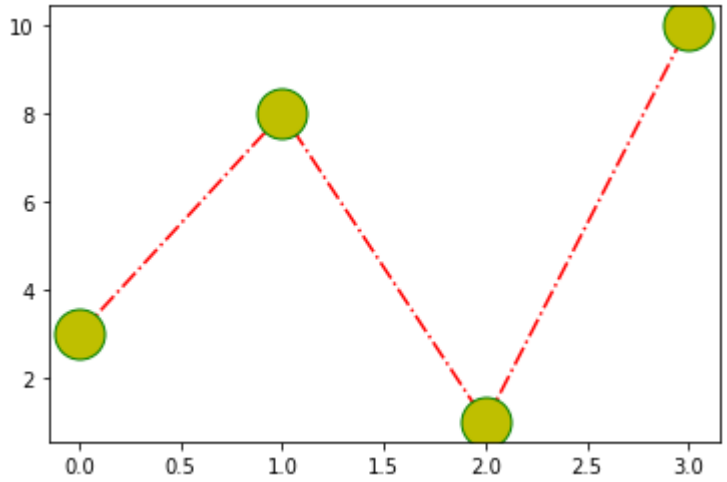
In [6]:

```
ypoints = np.array([3, 8, 1, 10, 5, 7])

plt.plot(ypoints)
plt.show()
```

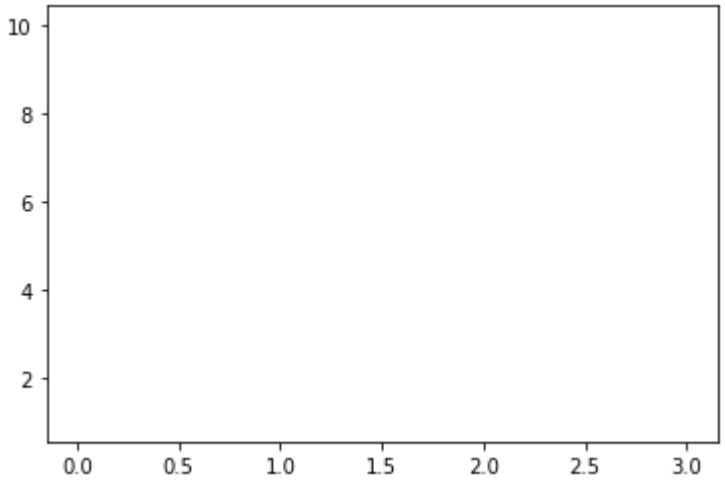


In [ ]:

In [41]:
```
ypoints = np.array([3, 8, 1, 10])

plt.plot(ypoints,'o-.r',ms=25,mec='g',mfc='y')
plt.show()
```

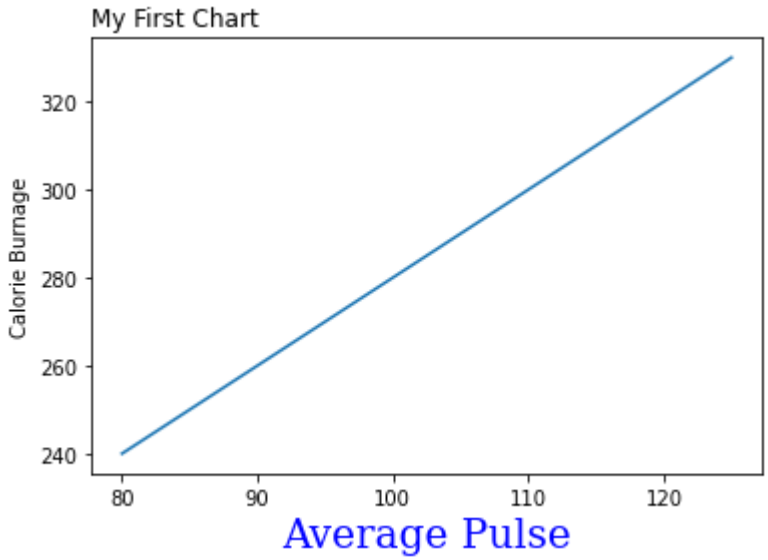


In [47]:
```
ypoints = np.array([3, 8, 1, 10])

plt.plot(ypoints, linestyle = 'None')
plt.show()
```



In [54]:

```
x = np.array([80, 85, 90, 95, 100, 105, 110, 115, 120, 125])
y = np.array([240, 250, 260, 270, 280, 290, 300, 310, 320, 330])

plt.plot(x, y)
font1 = {'family':'serif','color':'blue','size':20}
font2 = {'family':'serif','color':'darkred','size':15}


plt.xlabel("Average Pulse",fontdict=font1)
plt.ylabel("Calorie Burnage",)
plt.title("My First Chart",loc='left')

plt.show()
```



In [73]:

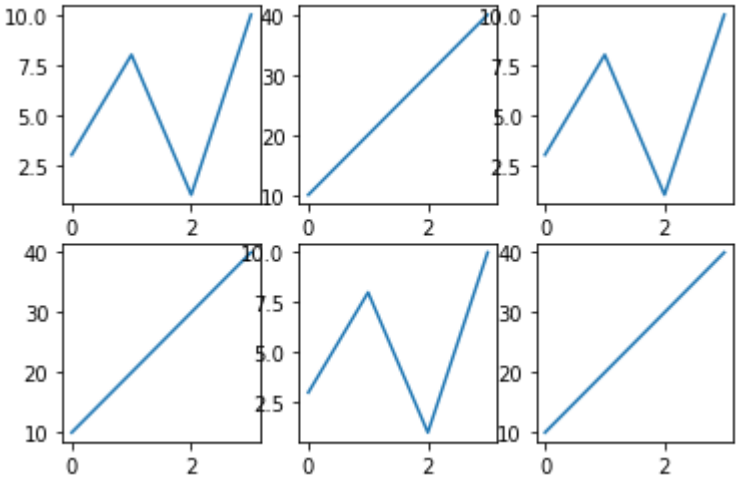
```
import numpy as np
import math
t=np.arange(0,2.5,0.1)
y1=np.sin(math.pi*t)
y2=np.sin(math.pi*t+math.pi/2)
y3=np.sin(math.pi*t+math.pi/2)
f = plt.figure()
f.set_figwidth(20)
f.set_figheight(15)
```

```
plt.plot(t,y1,marker='o',t,y2,t,y3,marker='b*',ms=12)
plt.show()
```

```
  File "C:\Users\student\AppData\Local\Temp/ipykernel_7292/2911271571.py", line 10
    plt.plot(t,y1,marker='o',t,y2,t,y3,marker='b*',ms=12)
                                                       ^
SyntaxError: positional argument follows keyword argument
```

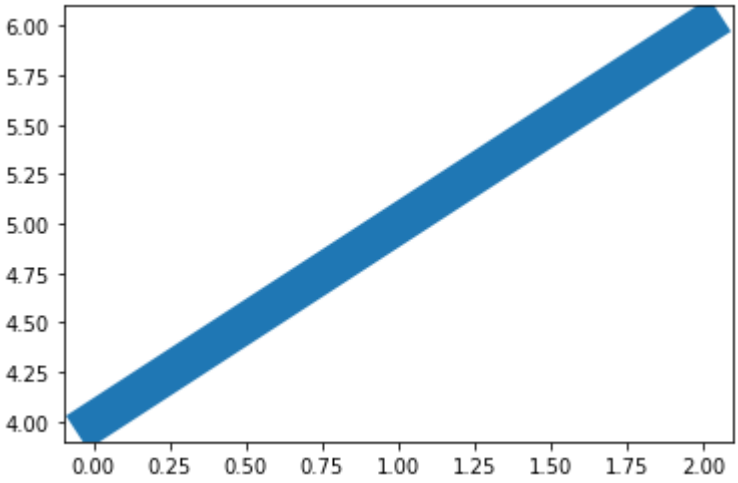
In [69]:

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

x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])

plt.subplot(2, 3, 1)
plt.plot(x,y)

x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])

plt.subplot(2, 3, 2)
plt.plot(x,y)

x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])

plt.subplot(2, 3, 3)
plt.plot(x,y)

x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])

plt.subplot(2, 3, 4)
plt.plot(x,y)

x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])

plt.subplot(2, 3, 5)
plt.plot(x,y)

x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])

plt.subplot(2, 3, 6)
plt.plot(x,y)

plt.show()
```



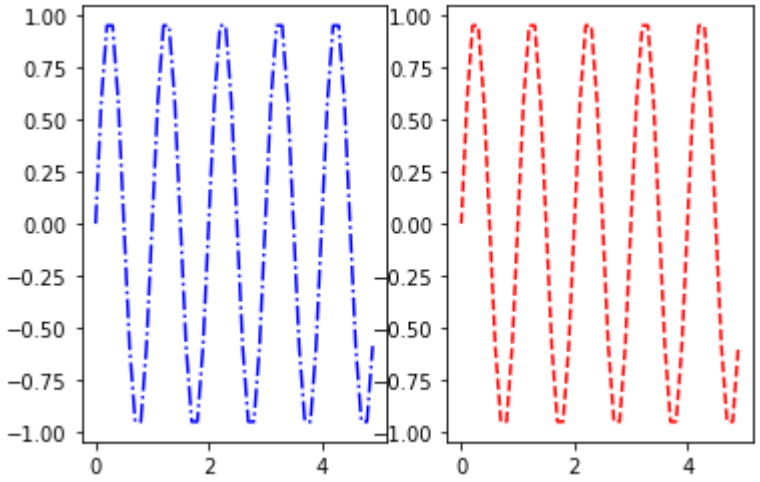
In [2]:

```
import matplotlib.pyplot as plt
plt.plot([4,5,6],linewidth=20)
plt.show()
```



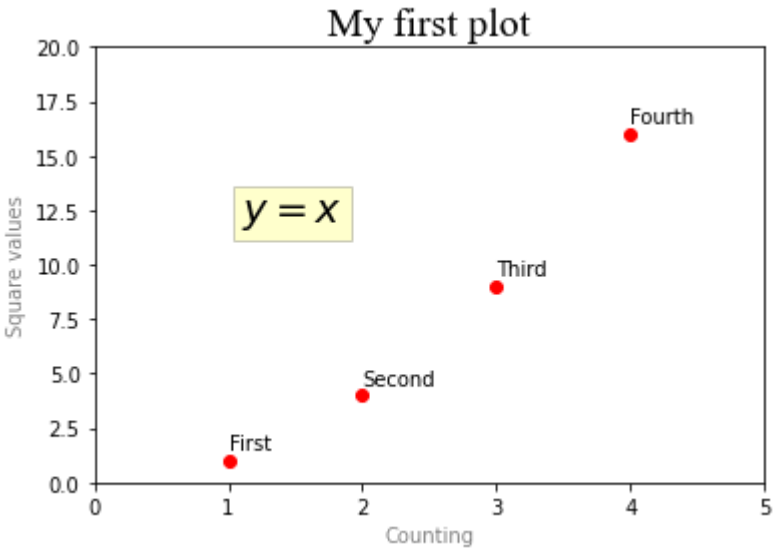
In [3]:

```
#Working with Multiple Figures and Axes
import matplotlib.pyplot as plt
import numpy as np
t = np.arange(0,5,0.1)
y1 = np.sin(2*np.pi*t)
y2 = np.sin(2*np.pi*t)
plt.subplot(121)
plt.plot(t,y1,'b-.')
plt.subplot(122)
plt.plot(t,y2,'r--')
plt.show()
```



In [4]:

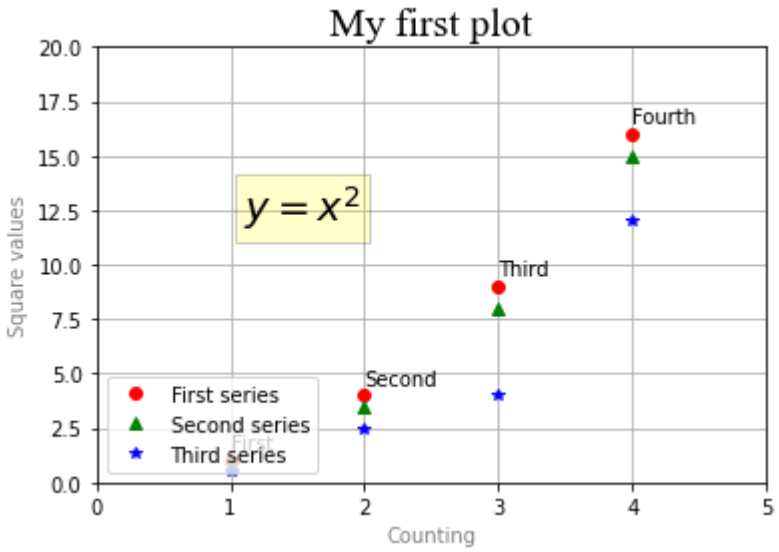
```
#Adding Elements to the Chart
#Adding text
import matplotlib.pyplot as plt
import numpy as np
plt.axis([0,5,0,20])
plt.title('My first plot',fontsize=20,fontname='Times New Roman')
plt.xlabel('Counting',color='gray')
plt.ylabel('Square values',color='gray')
plt.text(1,1.5,'First')
plt.text(2,4.5,'Second')
plt.text(3,9.5,'Third')
plt.text(4,16.5,'Fourth')
plt.text(1.1,12,'$y = x$',fontsize=20,bbox={'facecolor':'yellow','alpha':0.2})
plt.plot([1,2,3,4],[1,4,9,16],'ro')
plt.show()
```



In [5]:

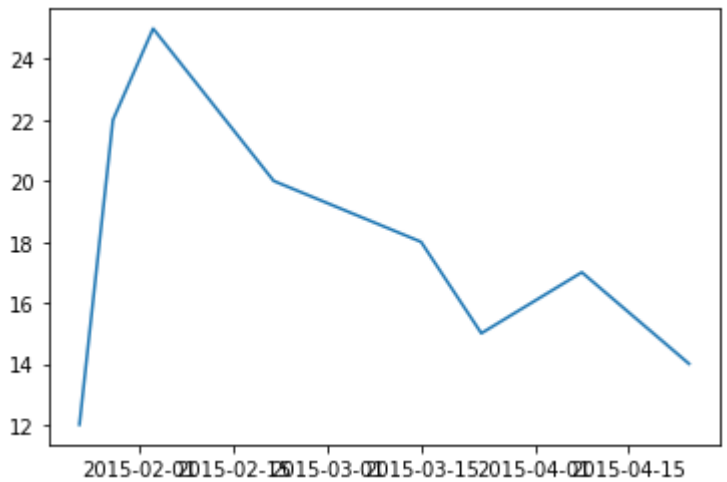
```
import matplotlib.pyplot as plt
plt.axis([0,5,0,20])
plt.title('My first plot',fontsize=20,fontname='Times New Roman')
plt.xlabel('Counting',color='gray')
plt.ylabel('Square values',color='gray')
plt.text(1,1.5,'First')
plt.text(2,4.5,'Second')
plt.text(3,9.5,'Third')
```

```
plt.text(4,16.5,'Fourth')
plt.text(1.1,12,'$y = x^2$',fontsize=20,bbox={'facecolor':'yellow','alpha':0.2})
plt.grid(True)
plt.plot([1,2,3,4],[1,4,9,16],'ro')
plt.plot([1,2,3,4],[0.8,3.5,8,15],'g^')
plt.plot([1,2,3,4],[0.5,2.5,4,12],'b*')
plt.legend(['First series','Second series','Third series'],loc=3)
plt.savefig('my_plot.png')
plt.show()
```



In [6]:

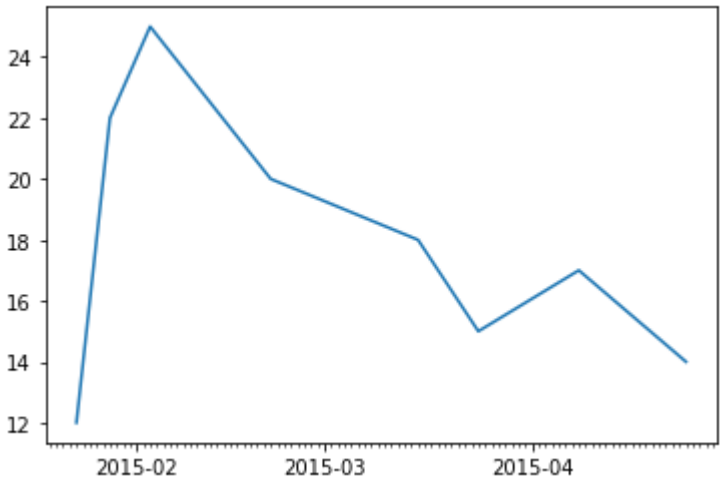
```
#Handling Date Values
import datetime
import numpy as np
import matplotlib.pyplot as plt
events = [datetime.date(2015,1,23),datetime.
date(2015,1,28),datetime.date(2015,2,3),datetime.
date(2015,2,21),datetime.date(2015,3,15),datetime.
date(2015,3,24),datetime.date(2015,4,8),datetime.date(2015,4,24)]
readings = [12,22,25,20,18,15,17,14]
plt.plot(events,readings)
plt.show()
```



In [8]:

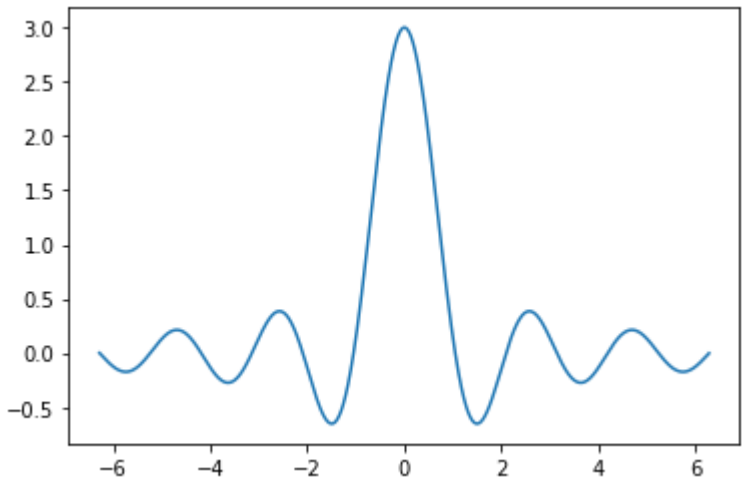
```
#Handlinf Date Values
```

```
import datetime
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.dates as mdates
months = mdates.MonthLocator()
days = mdates.DayLocator()
timeFmt = mdates.DateFormatter('%Y-%m')
events = [datetime.date(2015,1,23),datetime.
date(2015,1,28),datetime.date(2015,2,3),datetime.
date(2015,2,21),datetime.date(2015,3,15),datetime.
date(2015,3,24),datetime.date(2015,4,8),datetime.date(2015,4,24)]
readings = [12,22,25,20,18,15,17,14]
fig, ax = plt.subplots()
plt.plot(events,readings)
ax.xaxis.set_major_locator(months)
ax.xaxis.set_major_formatter(timeFmt)
ax.xaxis.set_minor_locator(days)
plt.show()
```

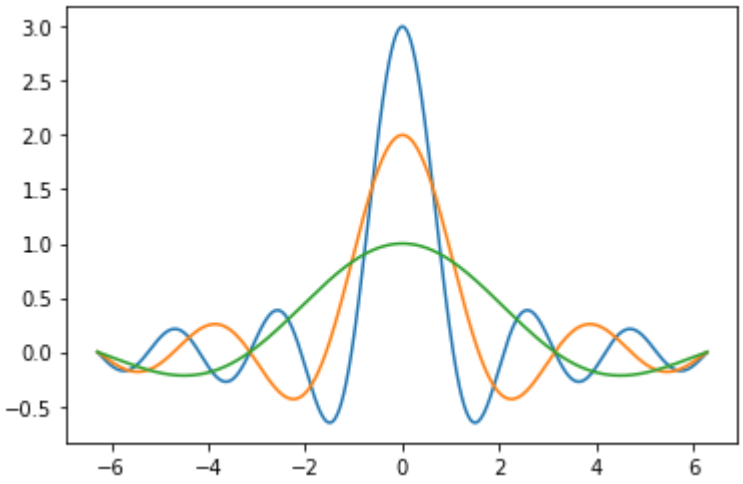


In [9]:

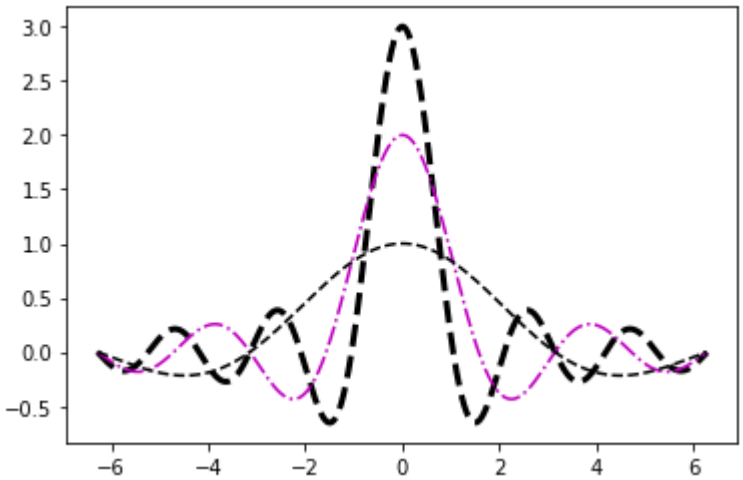
```
#Line Chart
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-2*np.pi,2*np.pi,0.01)
y = np.sin(3*x)/x
plt.plot(x,y)
plt.show()
```



In [10]:
```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-2*np.pi,2*np.pi,0.01)
y = np.sin(3*x)/x
y2 = np.sin(2*x)/x
y3 = np.sin(1*x)/x
plt.plot(x,y)
plt.plot(x,y2)
plt.plot(x,y3)
plt.show()
```

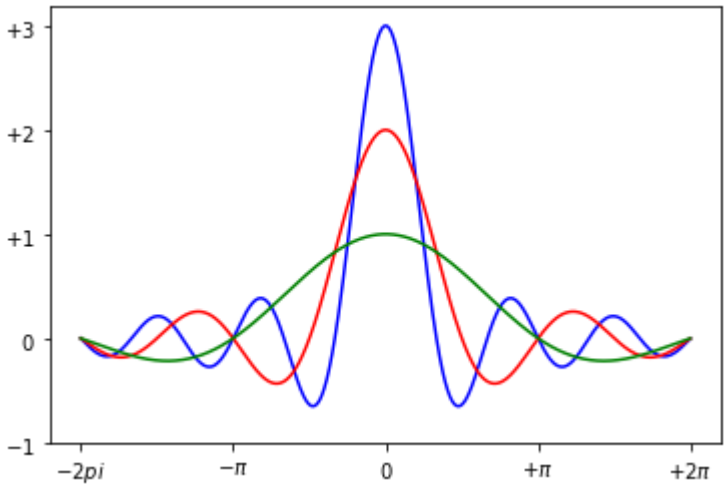


In [11]:
```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-2*np.pi,2*np.pi,0.01)
y = np.sin(3*x)/x
y2 = np.sin(2*x)/x
y3 = np.sin(1*x)/x
plt.plot(x,y,'k--',linewidth=3)
plt.plot(x,y2,'m-.')
plt.plot(x,y3,color='#010203',linestyle='--')
plt.show()
```



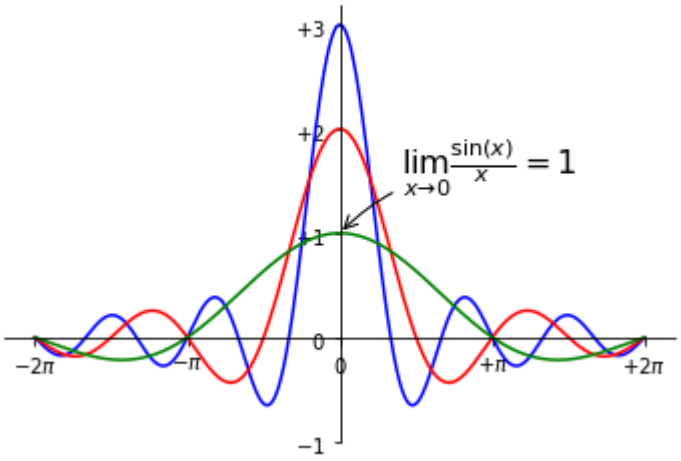
In [12]:
```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-2*np.pi,2*np.pi,0.01)
```

```
y = np.sin(3*x)/x
y2 = np.sin(2*x)/x
y3 = np.sin(x)/x
plt.plot(x,y,color='b')
plt.plot(x,y2,color='r')
plt.plot(x,y3,color='g')
plt.xticks([-2*np.pi, -np.pi, 0, np.pi, 2*np.pi], ['$-2pi$','$-\pi$','$0$','$+\pi$','$+
plt.yticks([-1,0,1,2,3], ['$-1$','$0$','$+1$','$+2$','$+3$'])
plt.show()
```



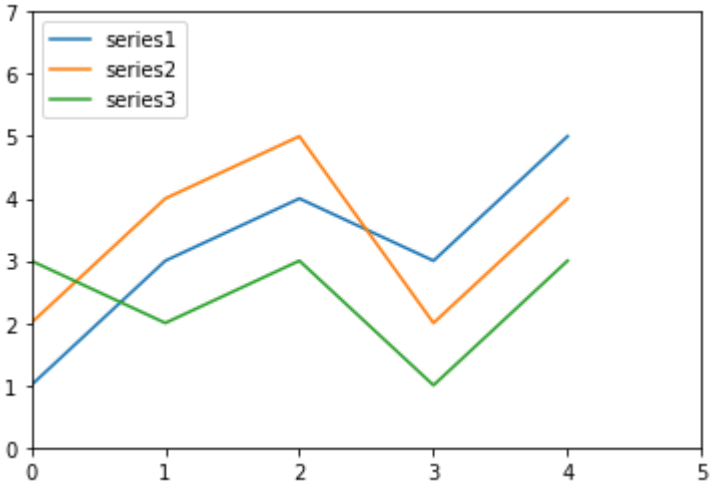
In [14]:

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-2*np.pi,2*np.pi,0.01)
y = np.sin(3*x)/x
y2 = np.sin(2*x)/x
y3 = np.sin(x)/x
plt.plot(x,y,color='b')
plt.plot(x,y2,color='r')

plt.plot(x,y3,color='g')
plt.xticks([-2*np.pi, -np.pi, 0, np.pi, 2*np.pi], [r'$-2\pi$',r'$-\pi$',r'$0$',r'$+\pi$
plt.yticks([-1,0,+1,+2,+3], [r'$-1$',r'$0$',r'$+1$',r'$+2$',r'$+3$'])
plt.annotate(r'$\lim_{x\to 0}\frac{\sin(x)}{x}= 1$', xy=[0,1],xycoords='data',xytext=[3
ax = plt.gca()
ax.spines['right'].set_color('none')
ax.spines['top'].set_color('none')
ax.xaxis.set_ticks_position('bottom')
ax.spines['bottom'].set_position(('data',0))
ax.yaxis.set_ticks_position('left')
ax.spines['left'].set_position(('data',0))
plt.show()
```



In [15]:

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
data = {'series1':[1,3,4,3,5], 'series2':[2,4,5,2,4], 'series3':[3,2,3,1,3]}
df = pd.DataFrame(data)
x = np.arange(5)
plt.axis([0,5,0,7])
plt.plot(x,df)
plt.legend(data, loc=2)
plt.show()
```

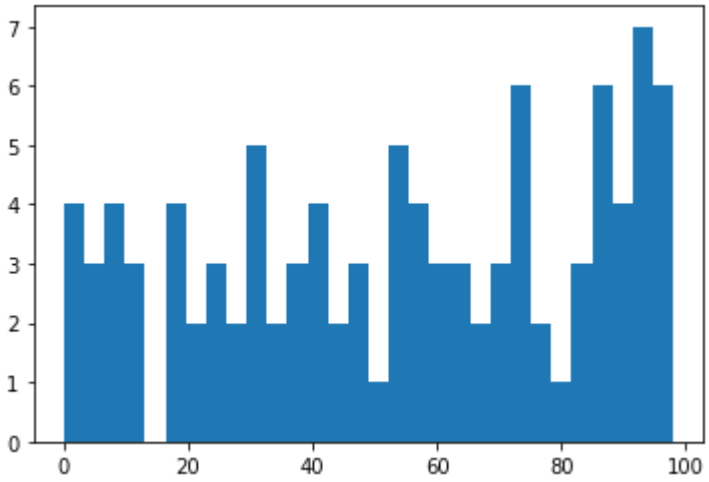


In [16]:

```
#Histograms
import matplotlib.pyplot as plt
import numpy as np
pop = np.random.randint(0,100,100)
print(pop)
n,bins,patches = plt.hist(pop,bins=30)
print(n,bins,patches)
plt.show()
```

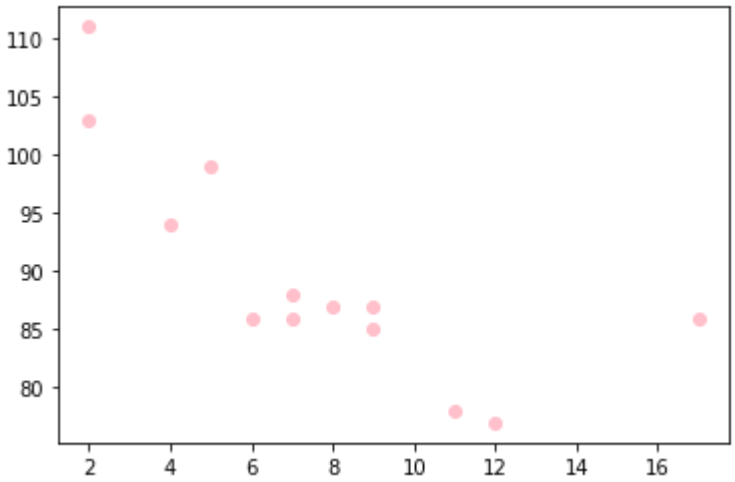
```
[25 40 58 93 70 95 48 89 37 75 17 92 64 35 89 58 19 12  1 32 91  7 59 70
 92 30 63 74 75 90 81 72 19 29  3  8 43 42 94 59 55 86 50 76 73 71 84  8
 30 87 66 10 37 57 96 44 77  4 29 92 56 47 67 87 97 22  0 93  4 54 84  4
 19 53 25 96 98 83  9 26 53 10 53 95 85 60 21 87 75  2 63 34 42 41 31 93
 32 47 37 88]
[4. 3. 4. 3. 0. 4. 2. 3. 2. 5. 2. 3. 4. 2. 3. 1. 5. 4. 3. 3. 2. 3. 6. 2.
```

```
 1. 3. 6. 4. 7. 6.] [ 0.          3.26666667  6.53333333  9.8        13.06666667 16.3333
3333
 19.6        22.86666667 26.13333333 29.4        32.66666667 35.93333333
 39.2        42.46666667 45.73333333 49.         52.26666667 55.53333333
 58.8        62.06666667 65.33333333 68.6        71.86666667 75.13333333
 78.4        81.66666667 84.93333333 88.2        91.46666667 94.73333333
 98.        ] <BarContainer object of 30 artists>
```



In [104…

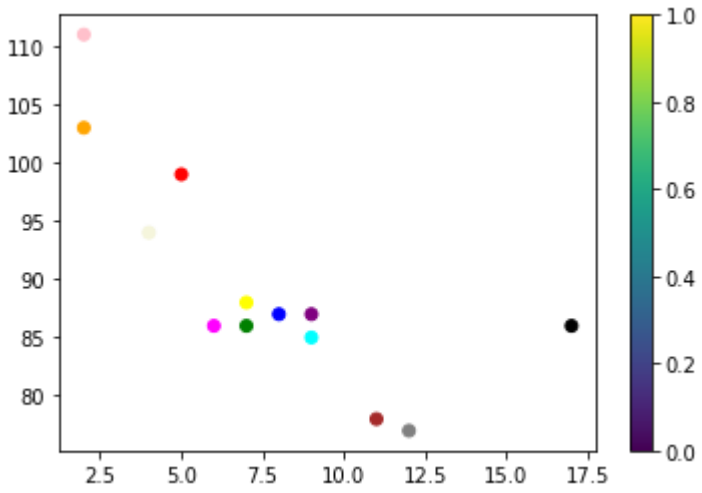
```
#scatter chart
import matplotlib.pyplot as plt
import numpy as np

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])

plt.scatter(x, y,c='pink')
plt.show()
```



In [98]:

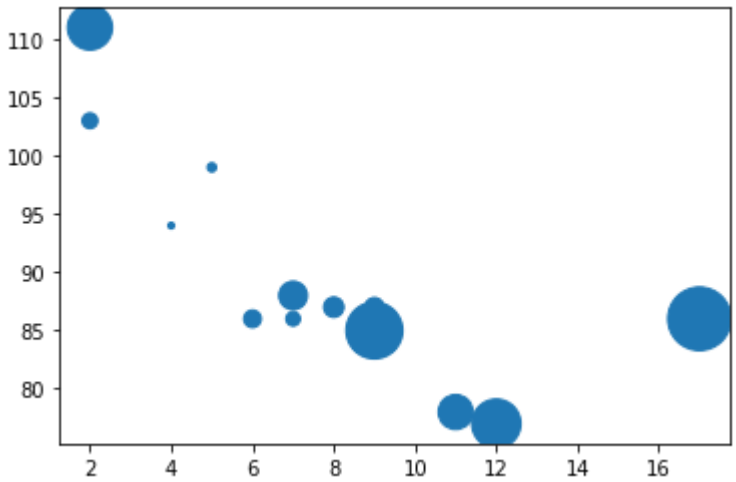
```
#scatter chart
import matplotlib.pyplot as plt
import numpy as np

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
colors = np.array(["red","green","blue","yellow","pink","black","orange","purple","beig

plt.scatter(x, y,c=colors,cmap='Greens')
```

```
plt.colorbar()
plt.show()
```



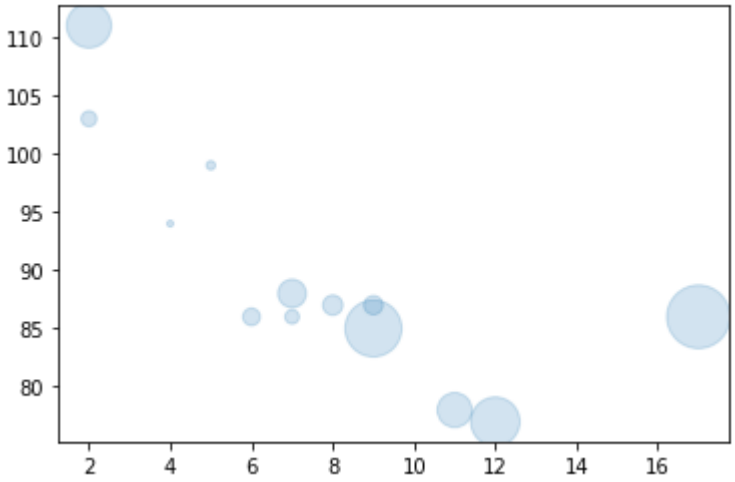
In [95]:

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

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
sizes = np.array([20,50,100,200,500,1000,60,90,10,300,600,800,75])

plt.scatter(x, y, s=sizes)

plt.show()
```



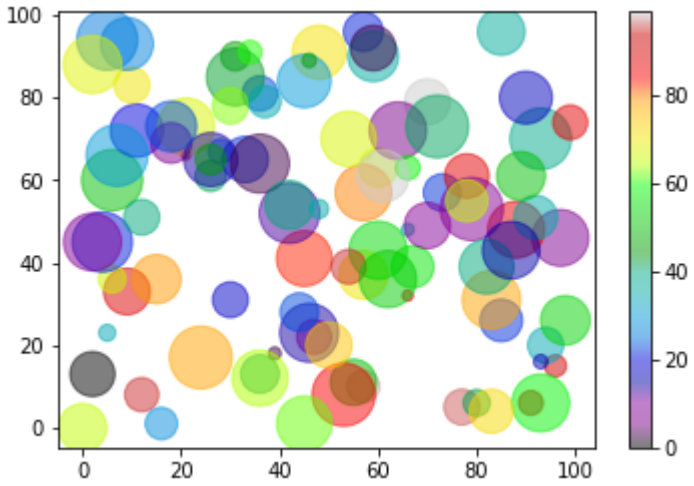
In [106…

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

x = np.array([5,7,8,7,2,17,2,9,4,11,12,9,6])
y = np.array([99,86,87,88,111,86,103,87,94,78,77,85,86])
sizes = np.array([20,50,100,200,500,1000,60,90,10,300,600,800,75])

plt.scatter(x, y, s=sizes, alpha=0.2)

plt.show()
```



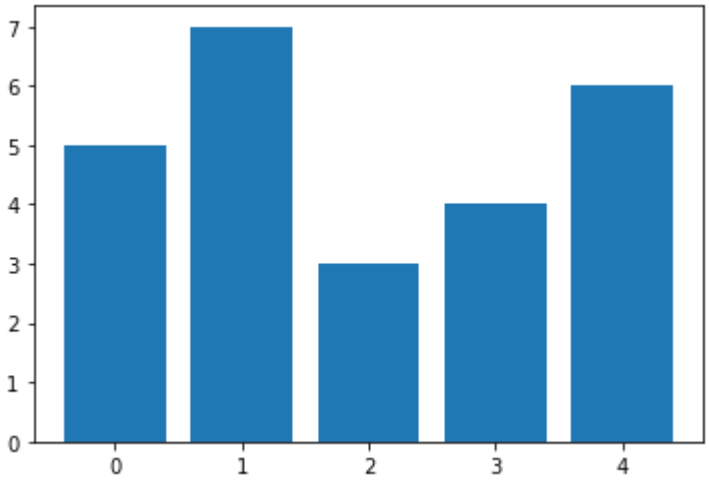
In [111…

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

x = np.random.randint(100, size=(100))
y = np.random.randint(100, size=(100))
colors = np.random.randint(100, size=(100))
sizes = 10 * np.random.randint(100, size=(100))

plt.scatter(x, y, c=colors, s=sizes, alpha=0.5, cmap='nipy_spectral')
#nipy_spectral
plt.colorbar()

plt.show()
```



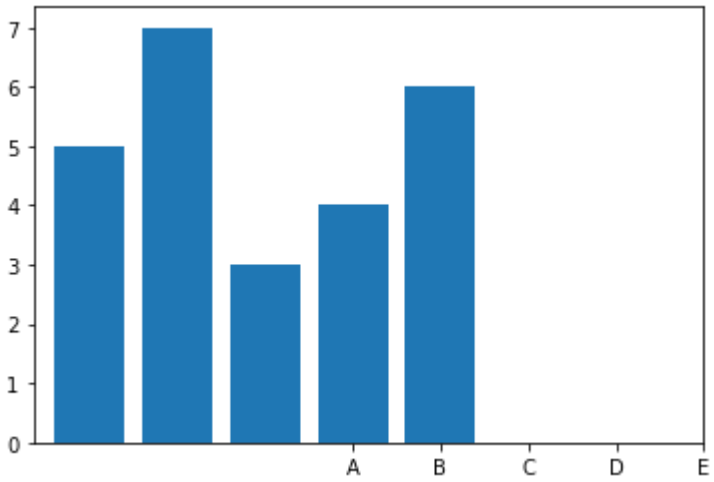
In [100…

```
import matplotlib.pyplot as plt
index = [0,1,2,3,4]
values = [5,7,3,4,6]
plt.bar(index,values)
plt.show()
```



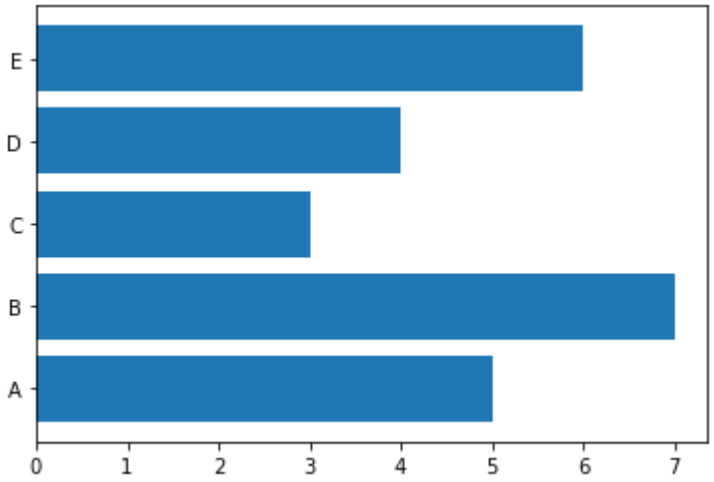
In [116…

```
import matplotlib.pyplot as plt
import numpy as np
index = np.arange(5)
values = [5,7,3,4,6]
plt.bar(index,values)
plt.xticks(index+3,['A','B','C','D','E'])
plt.show()
```



In [117…

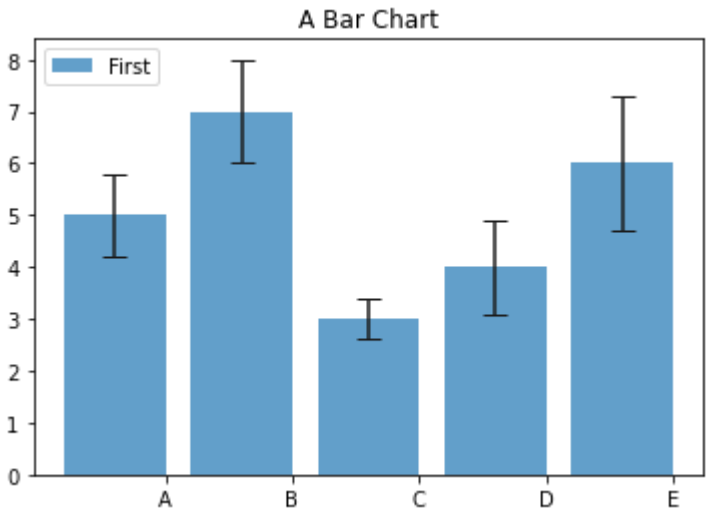
```
import matplotlib.pyplot as plt
import numpy as np
index = np.arange(5)
values = [5,7,3,4,6]
plt.barh(index,values)
plt.yticks(index,['A','B','C','D','E'])
plt.show()
```



In [ ]:

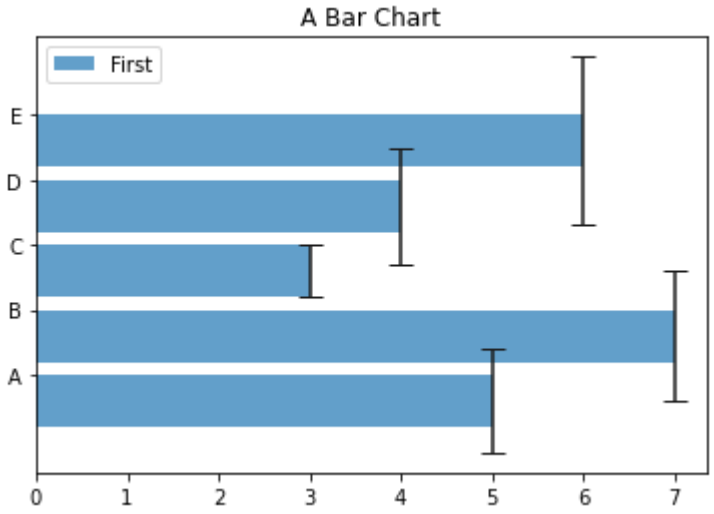
In [13]:
```
import numpy as np
index = np.arange(5)
values1 = [5,7,3,4,6]
std1 = [0.8,1,0.4,0.9,1.3]
plt.title('A Bar Chart')

plt.bar(index,values1,yerr=std1,error_kw={'ecolor':'0.1',
 'capsize':6},alpha=0.7,label='First')
plt.xticks(index+0.4,['A','B','C','D','E'])
plt.legend(loc=2)
```

Out[13]: <matplotlib.legend.Legend at 0x127c9e4f850>

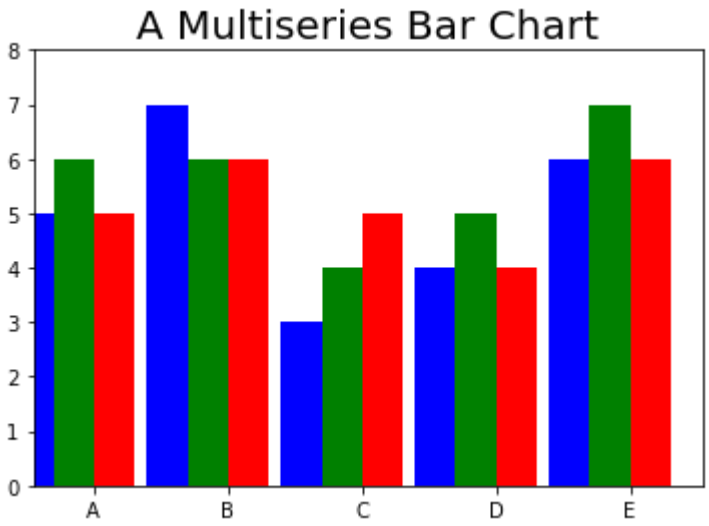


In [14]:
```
import numpy as np
index = np.arange(5)
values1 = [5,7,3,4,6]
std1 = [0.8,1,0.4,0.9,1.3]
plt.title('A Horizontal Bar Chart')
plt.barh(index,values1,yerr=std1,error_kw={'ecolor':'0.1',
 'capsize':6},alpha=0.7,label='First')
plt.yticks(index+0.4,['A','B','C','D','E'])
plt.legend(loc=2)
```

Out[14]:
```
<matplotlib.legend.Legend at 0x127c9d87fd0>
```



In [18]:
```
#Multiserial Bar Charts
import matplotlib.pyplot as plt
import numpy as np
index = np.arange(5)
values1 = [5,7,3,4,6]
values2 = [6,6,4,5,7]
values3 = [5,6,5,4,6]
bw = 0.3
plt.axis([0,5,0,8])
plt.title('A Multiseries Bar Chart',fontsize=20)
plt.bar(index,values1,bw,color='b')
plt.bar(index+bw,values2,bw,color='g')
plt.bar(index+2*bw,values3,bw,color='r')
plt.xticks(index+1.5*bw,['A','B','C','D','E'])
plt.show()
```

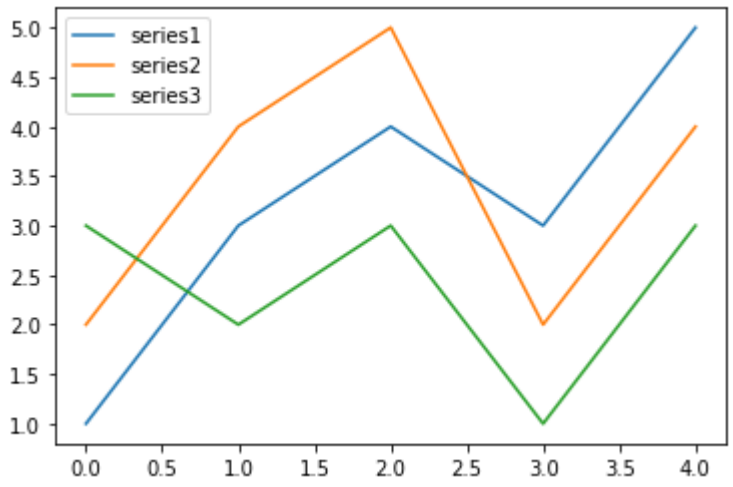


In [123…
```
#Multiseries Bar Charts with pandas Dataframe
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
data = {'series1':[1,3,4,3,5], 'series2':[2,4,5,2,4], 'series3':[3,2,3,1,3]}
```

```
df = pd.DataFrame(data)
df.plot(kind='line')
```

Out[123…

```
<AxesSubplot:>
```

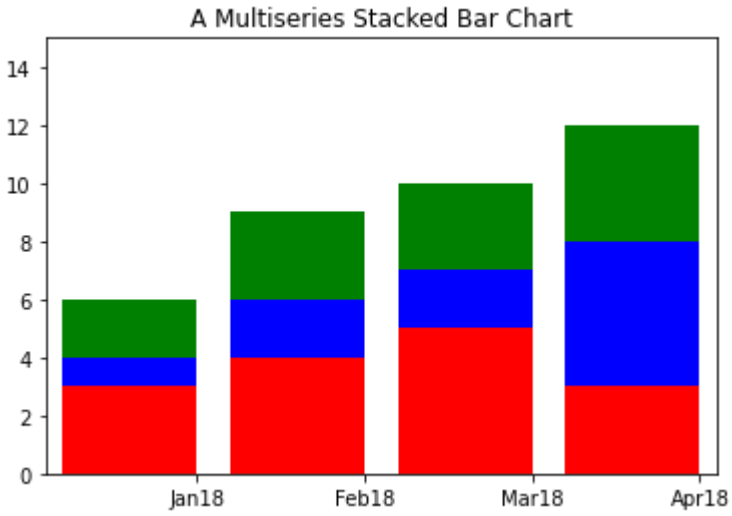


In [125…

```
#Multiseries Stacked Bar Charts
import matplotlib.pyplot as plt
import numpy as np
series1 = np.array([3,4,5,3])
series2 = np.array([1,2,2,5])
series3 = np.array([2,3,3,4])
index = np.arange(4)
plt.axis([-0.5,3.5,0,15])
plt.title('A Multiseries Stacked Bar Chart')
plt.bar(index,series1,color='r')
plt.bar(index,series2,color='b',left=series1)
plt.bar(index,series3,color='g',bottom=(series2+series1))
plt.xticks(index+0.4,['Jan18','Feb18','Mar18','Apr18'])
```

Out[125…

```
([<matplotlib.axis.XTick at 0x127cdba24c0>,
  <matplotlib.axis.XTick at 0x127cdba2490>,
  <matplotlib.axis.XTick at 0x127cdb880d0>,
  <matplotlib.axis.XTick at 0x127cdc02310>],
 [Text(0.4, 0, 'Jan18'),
  Text(1.4, 0, 'Feb18'),
  Text(2.4, 0, 'Mar18'),
  Text(3.4, 0, 'Apr18')])
```

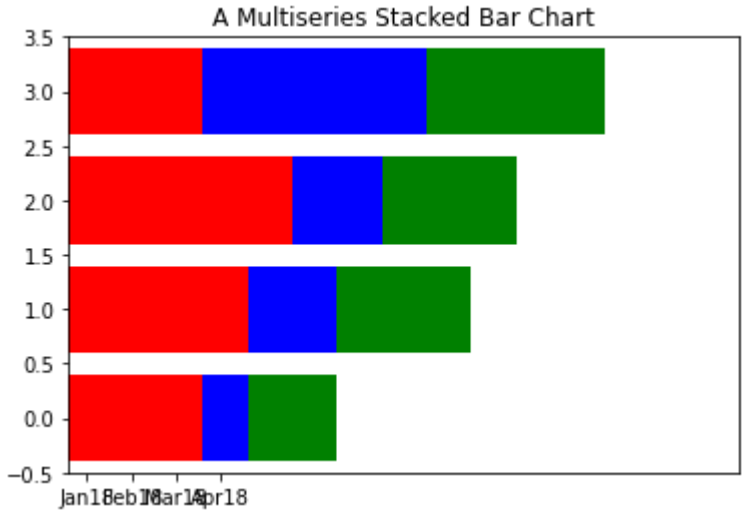


In [126…

```
#Multiseries Stacked Bar Charts
import matplotlib.pyplot as plt
import numpy as np
series1 = np.array([3,4,5,3])
series2 = np.array([1,2,2,5])
series3 = np.array([2,3,3,4])
index = np.arange(4)
plt.axis([0,15,-0.5,3.5])
plt.title('A Multiseries Stacked Bar Chart')
plt.barh(index,series1,color='r')
plt.barh(index,series2,color='b',left=series1)
plt.barh(index,series3,color='g',left=(series2+series1))
plt.xticks(index+0.4,['Jan18','Feb18','Mar18','Apr18'])
```

Out[126…

```
([<matplotlib.axis.XTick at 0x127cdbfdb20>,
  <matplotlib.axis.XTick at 0x127cdbfdac0>,
  <matplotlib.axis.XTick at 0x127cd9ed2b0>,
  <matplotlib.axis.XTick at 0x127cda03c40>],
 [Text(0.4, 0, 'Jan18'),
  Text(1.4, 0, 'Feb18'),
  Text(2.4, 0, 'Mar18'),
  Text(3.4, 0, 'Apr18')])
```



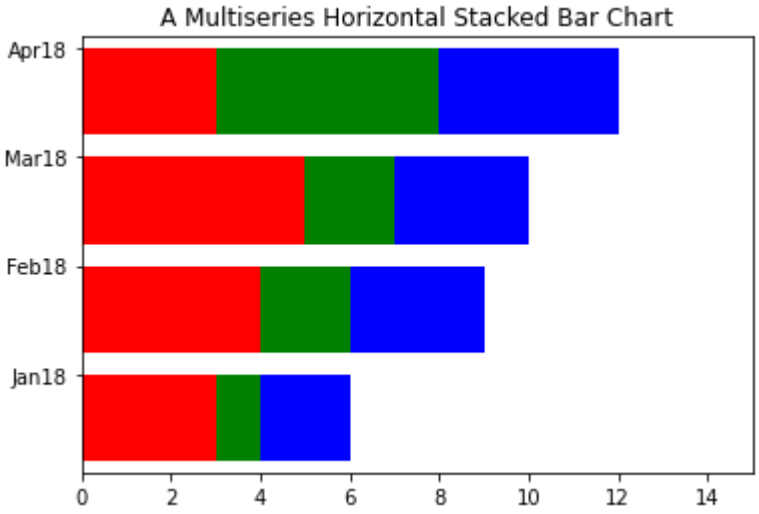
In [28]:

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

```
index = np.arange(4)
series1 = np.array([3,4,5,3])
series2 = np.array([1,2,2,5])
series3 = np.array([2,3,3,4])
plt.axis([0,15,-0.5,3.5])
plt.title('A Multiseries Horizontal Stacked Bar Chart')
plt.barh(index,series1,color='r')
plt.barh(index,series2,color='g',left=series1)
plt.barh(index,series3,color='b',left=(series1+series2))
plt.yticks(index+0.4,['Jan18','Feb18','Mar18','Apr18'])
```

Out[28]:

```
([<matplotlib.axis.YTick at 0x127cc0289a0>,
  <matplotlib.axis.YTick at 0x127cc028220>,
  <matplotlib.axis.YTick at 0x127cc016be0>,
  <matplotlib.axis.YTick at 0x127cc063a30>],
 [Text(0, 0.4, 'Jan18'),
  Text(0, 1.4, 'Feb18'),
  Text(0, 2.4, 'Mar18'),
  Text(0, 3.4, 'Apr18')])
```

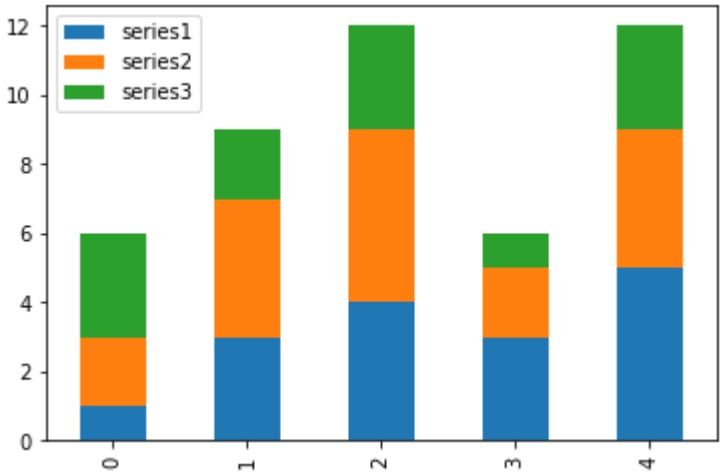


In [31]:

```
#Stacked Bar Charts with a pandas Dataframe
import matplotlib.pyplot as plt
import pandas as pd
data = {'series1':[1,3,4,3,5],
 'series2':[2,4,5,2,4],
 'series3':[3,2,3,1,3]}
df = pd.DataFrame(data)
df.plot(kind='bar', stacked=True)
```

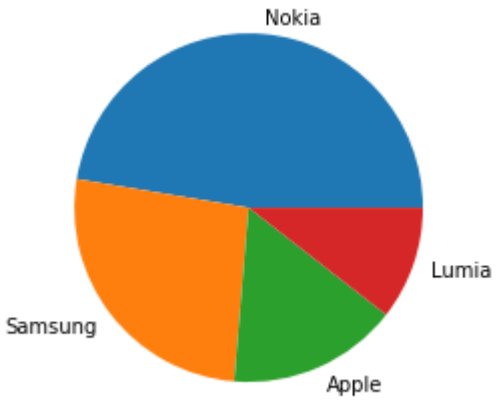
Out[31]:

```
<AxesSubplot:>
```



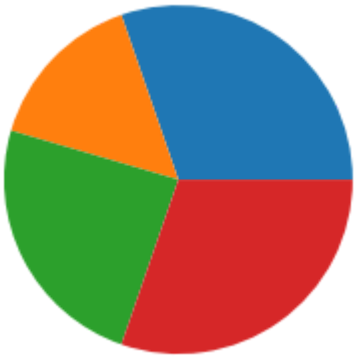
In [77]:

```
#Pie Chart
import matplotlib.pyplot as plt
x = np.array([45,25,15,10])
mylabels = ['Nokia','Samsung','Apple','Lumia']
plt.pie(x,labels=mylabels)
plt.show()
```



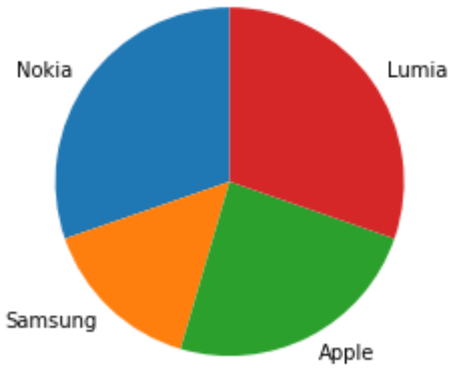
In [84]:

```
#Pie Chart
import matplotlib.pyplot as plt
x = np.array([10,5,8,10])
plt.pie(x)
plt.show()
```



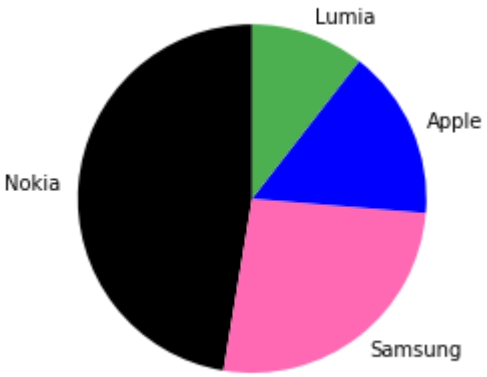
In [85]:

```
#Pie Chart
import matplotlib.pyplot as plt
x = np.array([10,5,8,10])
labels = ['Nokia','Samsung','Apple','Lumia']
plt.pie(x,labels=labels,startangle=90)
plt.show()
```



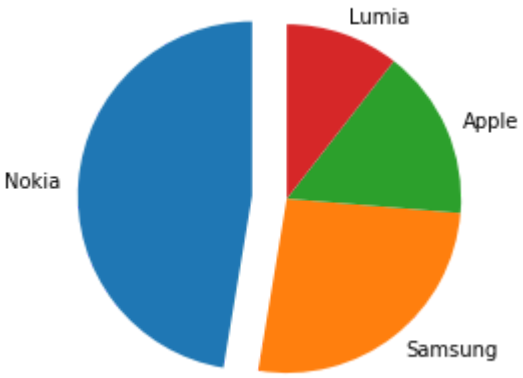
In [73]:

```
#Pie Chart
import matplotlib.pyplot as plt
x = np.array([45,25,15,10])
labels = ['Nokia','Samsung','Apple','Lumia']
mycolors = ["black", "hotpink", "b", "#4CAF50"]
plt.pie(x,labels=labels,colors=mycolors,startangle=90)
plt.show()
```



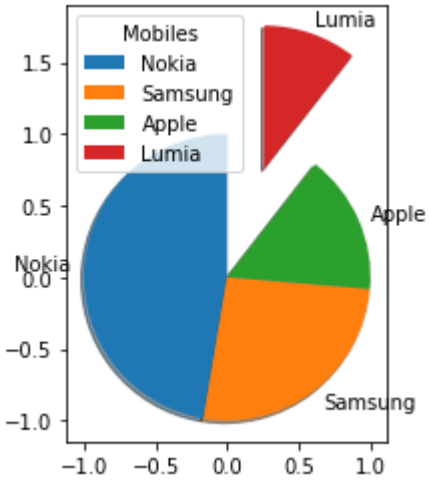
In [127…

```
#Pie Chart
import matplotlib.pyplot as plt
x = np.array([45,25,15,10])
labels = ['Nokia','Samsung','Apple','Lumia']
myexplode=[0.2,0,0,0]
plt.pie(x,labels=labels,startangle=90,explode=myexplode)
plt.show()
```



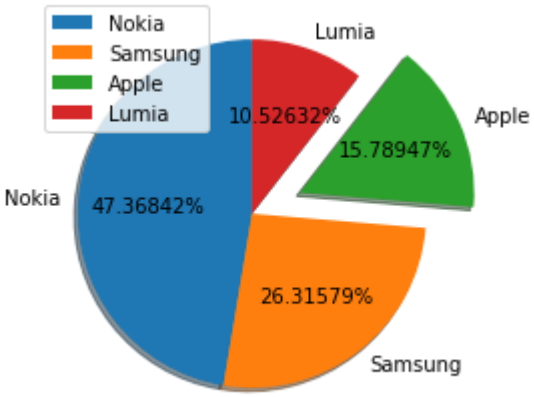
In [136…

```
#Pie Chart
import matplotlib.pyplot as plt
x = np.array([45,25,15,10])
labels = ['Nokia','Samsung','Apple','Lumia']
myexplode=[0,0,0,0.8]
plt.pie(x,labels=labels,startangle=90,explode=myexplode,shadow=True,frame=True)
plt.legend(title='Mobiles')
plt.show()
```



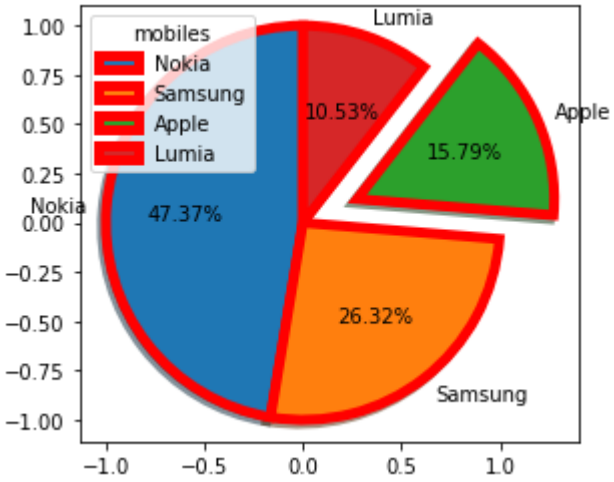
In [139…

```
#Pie Chart
import matplotlib.pyplot as plt
x = np.array([45,25,15,10])
labels = ['Nokia','Samsung','Apple','Lumia']
myexplode=[0,0,0.3,0]
plt.pie(x,labels=labels,startangle=90,explode=myexplode,shadow=True,autopct='%0.5f%%')
plt.legend()
plt.show()
```



In [140…

```
#Pie Chart
import matplotlib.pyplot as plt
x = np.array([45,25,15,10])
labels = ['Nokia','Samsung','Apple','Lumia']
myexplode=[0,0,0.3,0]
plt.pie(x,labels=labels,startangle=90,explode=myexplode,frame=True,shadow=True,autopct=

                                    'edgecolor' : "red" })
plt.legend(title='mobiles')
plt.show()
```

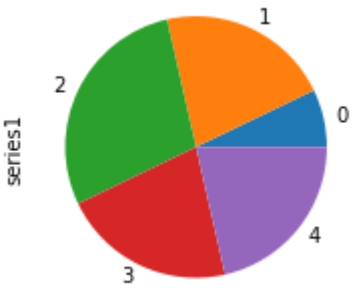


In [146…

```
#Pie Charts with a pandas Dataframe
import matplotlib.pyplot as plt
import pandas as pd
data = {'series1':[1,3,4,3,3],
 'series2':[2,4,5,2,4],
 'series3':[3,2,3,1,3]}
df = pd.DataFrame(data)
df['series1'].plot(kind='pie',figsize=(3,3))
```

Out[146…

```
<AxesSubplot:ylabel='series1'>
```

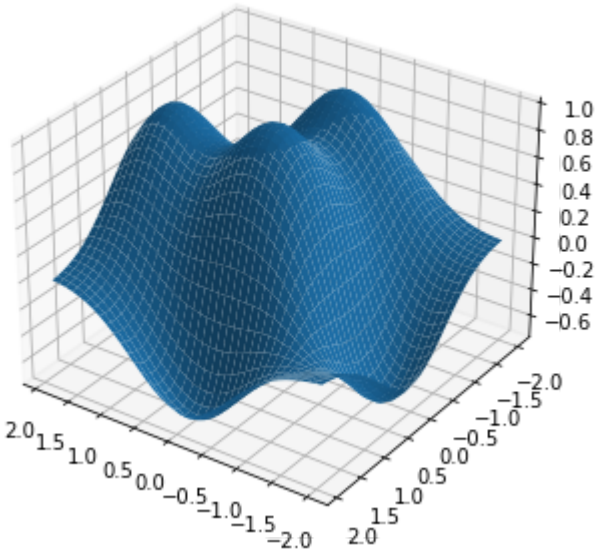


In [1]:

```
from mpl_toolkits.mplot3d import Axes3D
import matplotlib.pyplot as plt
import numpy as np
fig = plt.figure()
ax = Axes3D(fig)
X = np.arange(-2,2,0.1)
Y = np.arange(-2,2,0.1)
X,Y = np.meshgrid(X,Y)
def f(x,y):
    return (1 - y**5 + x**5)*np.exp(-x**2-y**2)
#ax.plot_surface(X,Y,f(X,Y), rstride=1, cstride=1)
ax.plot_surface(X,Y,f(X,Y), rstride=1, cstride=1)
ax.view_init(elev=30,azim=125)
plt.show()
```

```
C:\Users\student\AppData\Local\Temp/ipykernel_3920/3809197905.py:5: MatplotlibDeprecatio
nWarning: Axes3D(fig) adding itself to the figure is deprecated since 3.4. Pass the keyw
ord argument auto_add_to_figure=False and use fig.add_axes(ax) to suppress this warning.
The default value of auto_add_to_figure will change to False in mpl3.5 and True values w
```

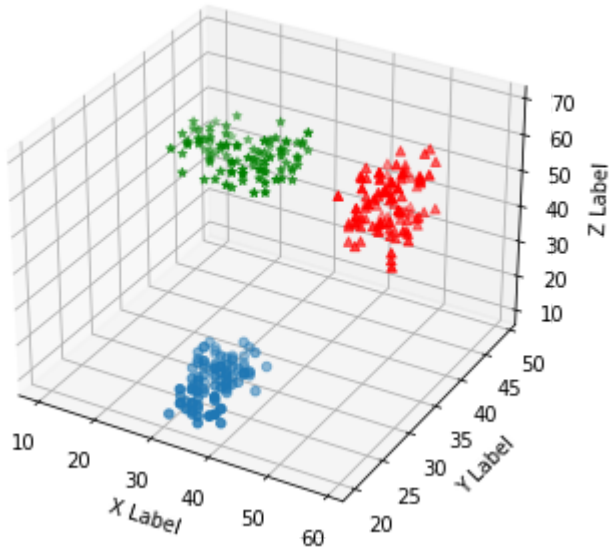
```
ill no longer work in 3.6.  This is consistent with other Axes classes.
  ax = Axes3D(fig)
```



In [2]:

```
import matplotlib.pyplot as plt
import numpy as np
from mpl_toolkits.mplot3d import Axes3D
xs = np.random.randint(30,40,100)
ys = np.random.randint(20,30,100)
zs = np.random.randint(10,20,100)
xs2 = np.random.randint(50,60,100)
ys2 = np.random.randint(30,40,100)
zs2 = np.random.randint(50,70,100)
xs3 = np.random.randint(10,30,100)
ys3 = np.random.randint(40,50,100)
zs3 = np.random.randint(40,50,100)
fig = plt.figure()
ax = Axes3D(fig)
ax.scatter(xs,ys,zs)
ax.scatter(xs2,ys2,zs2,c='r',marker='^')
ax.scatter(xs3,ys3,zs3,c='g',marker='*')
ax.set_xlabel('X Label')
ax.set_ylabel('Y Label')
ax.set_zlabel('Z Label')
plt.show()
```

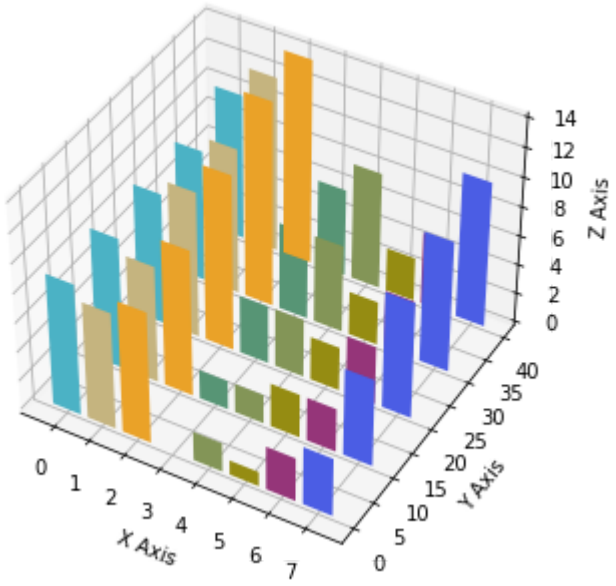
```
C:\Users\student\AppData\Local\Temp/ipykernel_3920/939336687.py:14: MatplotlibDeprecatio
nWarning: Axes3D(fig) adding itself to the figure is deprecated since 3.4. Pass the keyw
ord argument auto_add_to_figure=False and use fig.add_axes(ax) to suppress this warning.
The default value of auto_add_to_figure will change to False in mpl3.5 and True values w
ill no longer work in 3.6.  This is consistent with other Axes classes.
  ax = Axes3D(fig)
```



In [3]:

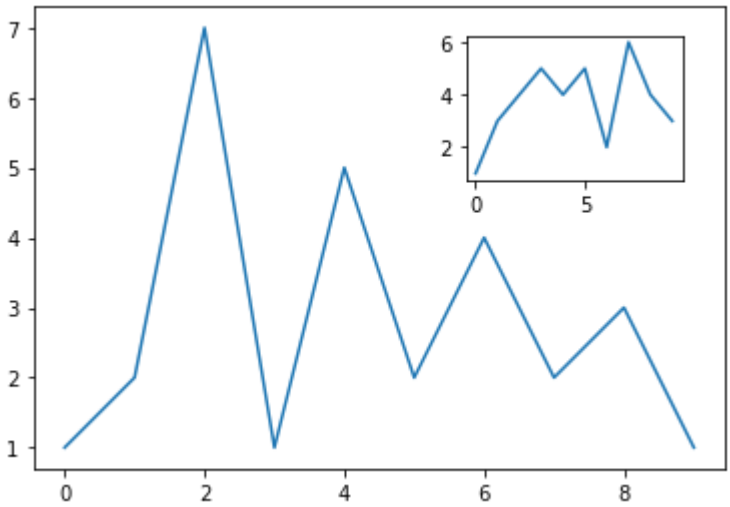
```
import matplotlib.pyplot as plt
import numpy as np
from mpl_toolkits.mplot3d import Axes3D
x = np.arange(8)
y = np.random.randint(0,10,8)
y2 = y + np.random.randint(0,3,8)
y3 = y2 + np.random.randint(0,3,8)
y4 = y3 + np.random.randint(0,3,8)
y5 = y4 + np.random.randint(0,3,8)
clr = ['#4bb2c5', '#c5b47f', '#EAA228', '#579575', '#839557','#958c12', '#953579', '#4b
fig = plt.figure()
ax = Axes3D(fig)
ax.bar(x,y,0,zdir='y',color=clr)
ax.bar(x,y2,10,zdir='y',color=clr)
ax.bar(x,y3,20,zdir='y',color=clr)
ax.bar(x,y4,30,zdir='y',color=clr)
ax.bar(x,y5,40,zdir='y',color=clr)
ax.set_xlabel('X Axis')
ax.set_ylabel('Y Axis')
ax.set_zlabel('Z Axis')
ax.view_init(elev=40)
plt.show()
```

```
C:\Users\student\AppData\Local\Temp/ipykernel_3920/1450371942.py:12: MatplotlibDeprecati
onWarning: Axes3D(fig) adding itself to the figure is deprecated since 3.4. Pass the key
word argument auto_add_to_figure=False and use fig.add_axes(ax) to suppress this warnin
g. The default value of auto_add_to_figure will change to False in mpl3.5 and True value
s will no longer work in 3.6.  This is consistent with other Axes classes.
  ax = Axes3D(fig)
```



In [4]:

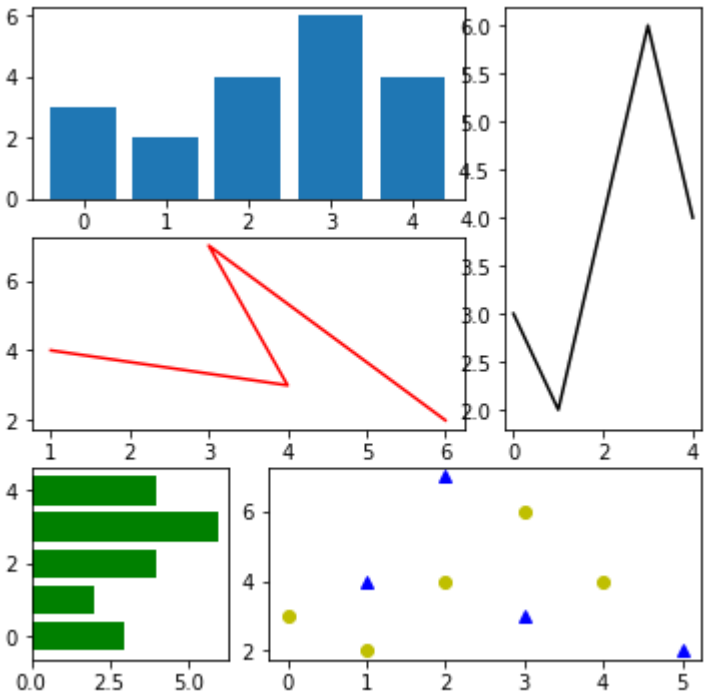
```
import matplotlib.pyplot as plt
import numpy as np
fig = plt.figure()
ax = fig.add_axes([0.1,0.1,0.8,0.8])
inner_ax = fig.add_axes([0.6,0.6,0.25,0.25])
x1 = np.arange(10)
y1 = np.array([1,2,7,1,5,2,4,2,3,1])
x2 = np.arange(10)
y2 = np.array([1,3,4,5,4,5,2,6,4,3])
ax.plot(x1,y1)
inner_ax.plot(x2,y2)
plt.show()
```



In [5]:

```
import matplotlib.pyplot as plt
import numpy as np
gs = plt.GridSpec(3,3)
fig = plt.figure(figsize=(6,6))
x = np.array([1,4,3,6])
y = np.array([4,3,7,2])
x1 = np.array([1,3,2,5])
y1 = np.array([4,3,7,2])
```

```
x2 = np.arange(5)
y2 = np.array([3,2,4,6,4])
s1 = fig.add_subplot(gs[1,:2])
s1.plot(x,y,'r')
s2 = fig.add_subplot(gs[0,:2])
s2.bar(x2,y2)
s3 = fig.add_subplot(gs[2,0])
s3.barh(x2,y2,color='g')
s4 = fig.add_subplot(gs[:2,2])
s4.plot(x2,y2,'k')
s5 = fig.add_subplot(gs[2,1:])
s5.plot(x1,y1,'b^',x2,y2,'yo')
plt.show()
```



In [ ]: