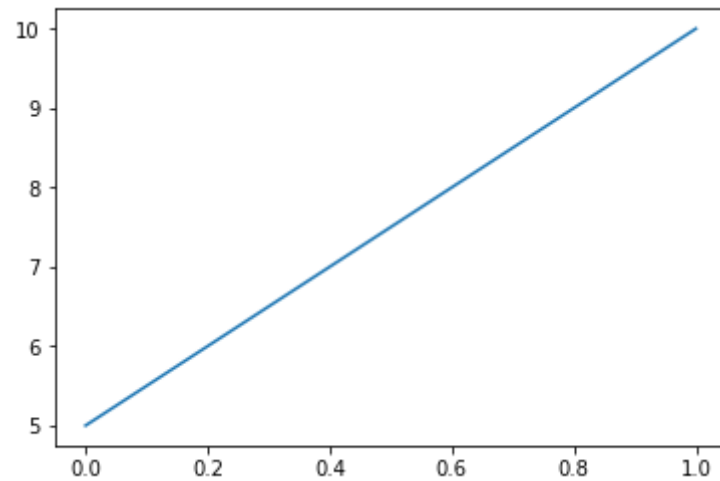
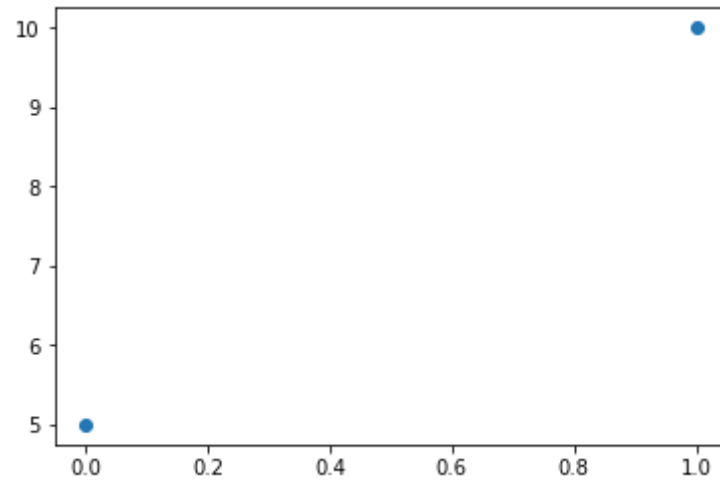


```
In [1]: 1 import matplotlib.pyplot as plt  
2 import numpy as np  
3 import pandas as pd
```

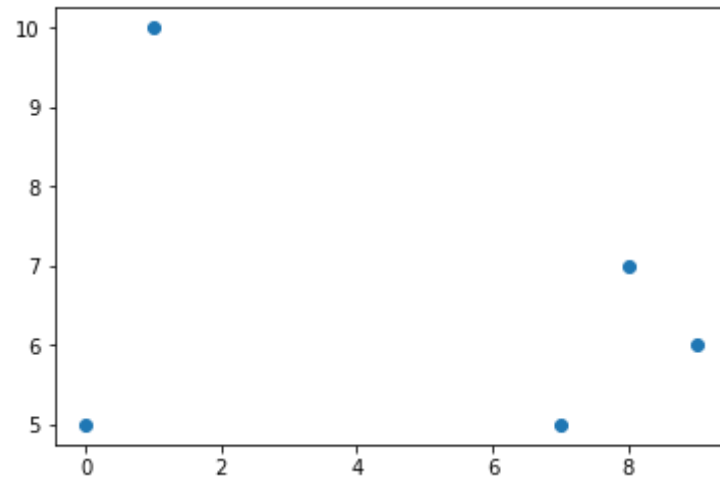
```
In [2]: 1 ## Plot a Line in pyplot  
2 xpoints = np.array([0,1])  
3 ypoints = np.array([5,10])  
4  
5 plt.plot(xpoints,ypoints)  
6 plt.show()
```



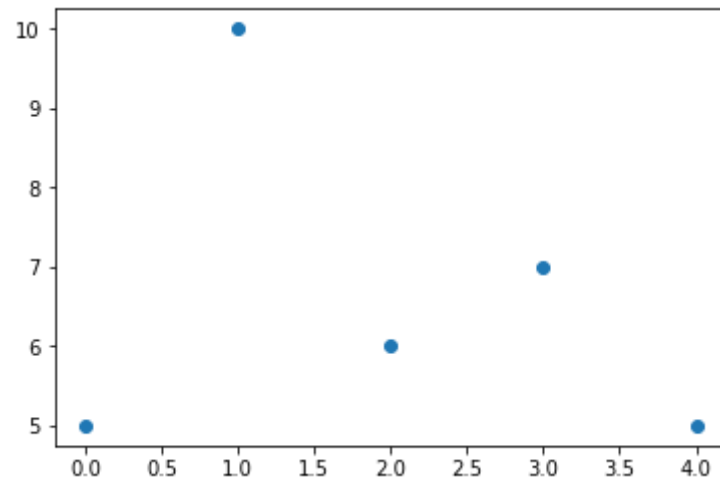
```
In [3]: 1  ## Plot points in pyplot
2  xpoints = np.array([0,1])
3  ypoints = np.array([5,10])
4
5  plt.plot(xpoints,ypoints,'o')
6  plt.show()
```



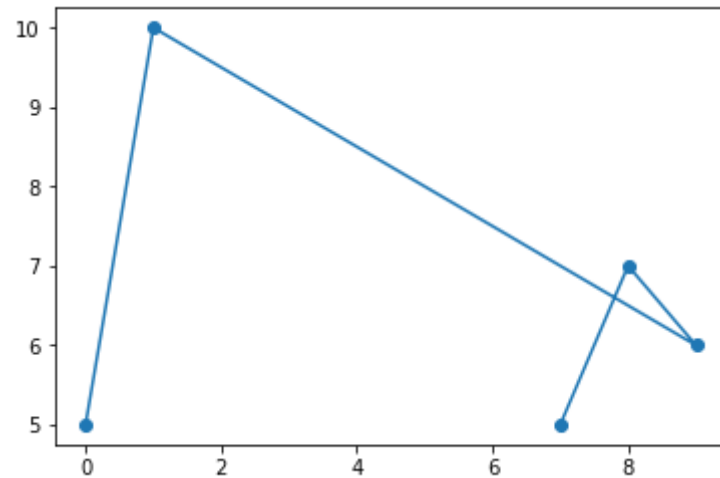
```
In [4]: 1  ## Plot points in pyplot
2  xpoints = np.array([0,1,9,8,7])
3  ypoints = np.array([5,10,6,7,5])
4
5  plt.plot(xpoints,ypoints,'o')
6  plt.show()
```



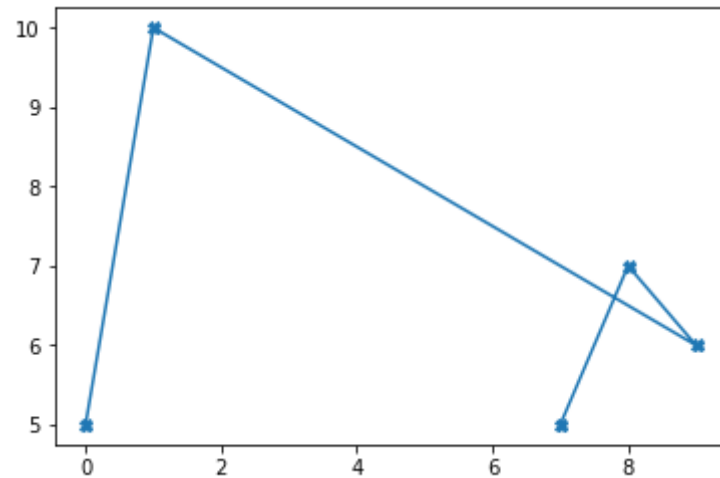
```
In [5]: 1  ## Plot points in pyplot  
2  ypoints = np.array([5,10,6,7,5])  
3  
4  plt.plot(ypoints, 'o')  
5  plt.show()
```



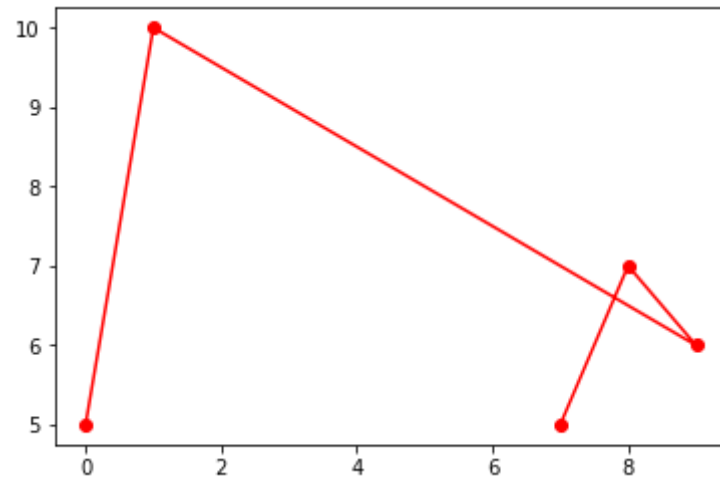
```
In [6]: 1  ## Plot lines with markers in pyplot
2  xpoints = np.array([0,1,9,8,7])
3  ypoints = np.array([5,10,6,7,5])
4
5  plt.plot(xpoints,ypoints,marker = 'o')
6  plt.show()
```



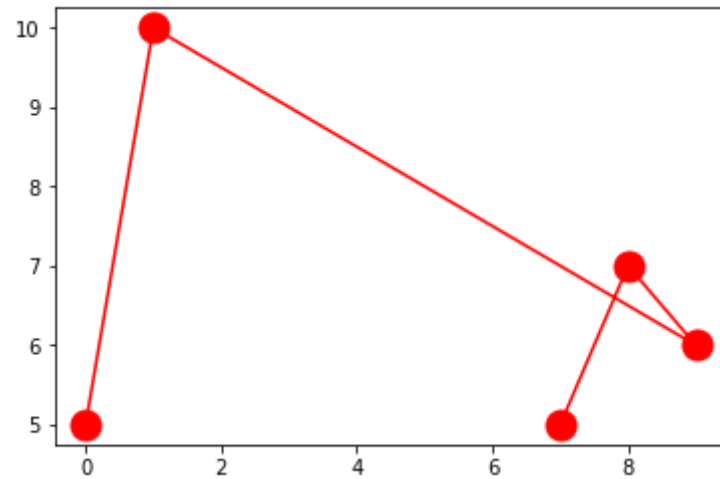
```
In [7]: 1  ## Plot lines with custom markers in pyplot
2  xpoints = np.array([0,1,9,8,7])
3  ypoints = np.array([5,10,6,7,5])
4
5  plt.plot(xpoints,ypoints,marker = 'x')
6  plt.show()
```



```
In [8]: 1  ## Plot lines with custom colors in pyplot
2
3  xpoints = np.array([0,1,9,8,7])
4  ypoints = np.array([5,10,6,7,5])
5
6  plt.plot(xpoints,ypoints,color = 'r',marker = 'o')
7  plt.show()
```

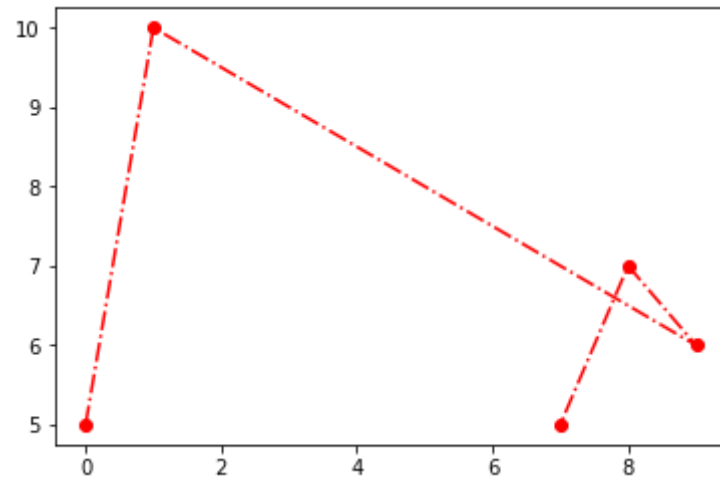


```
In [9]: 1  ## Plot lines with custom colors in pyplot
2
3  xpoints = np.array([0,1,9,8,7])
4  ypoints = np.array([5,10,6,7,5])
5
6  plt.plot(xpoints,ypoints,color = 'r',marker = 'o',ms = 15)
7  plt.show()
```

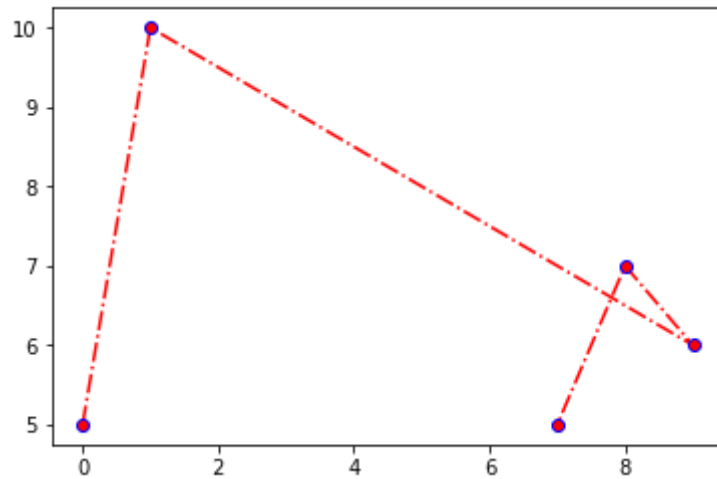




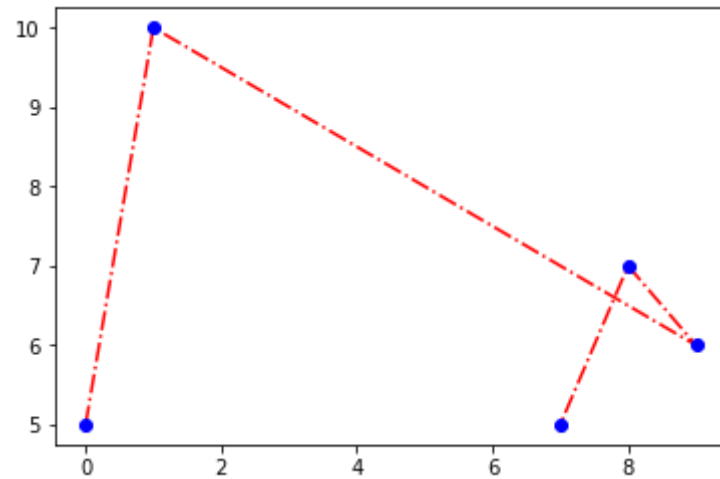
```
In [10]: 1  ## Plot lines with custom colors in pyplot
2
3  xpoints = np.array([0,1,9,8,7])
4  ypoints = np.array([5,10,6,7,5])
5
6  plt.plot(xpoints,ypoints,color = 'r',marker = 'o',linestyle = '-.')
7  plt.show()
```



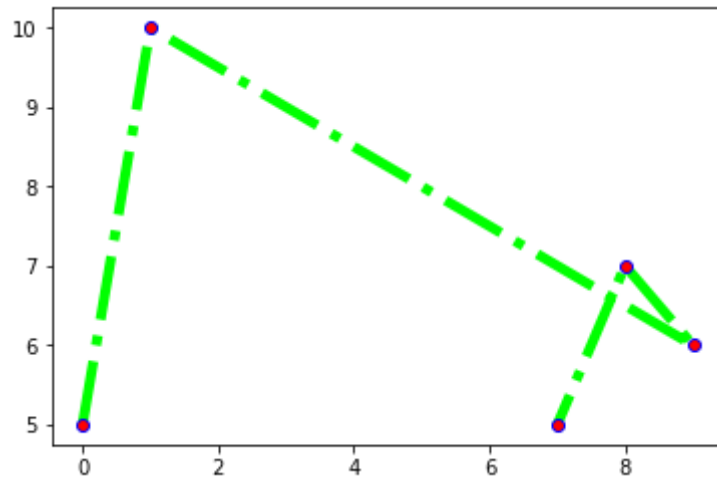
```
In [11]: 1  ## Plot lines with custom colors for markers in pyplot
2
3  xpoints = np.array([0,1,9,8,7])
4  ypoints = np.array([5,10,6,7,5])
5
6  plt.plot(xpoints,ypoints,color = 'r',marker = 'o',linestyle = '-.',mec = 'b')
7  plt.show()
```



```
In [12]: 1  ## Plot lines with custom colors for markers in pyplot
2
3  xpoints = np.array([0,1,9,8,7])
4  ypoints = np.array([5,10,6,7,5])
5
6  plt.plot(xpoints,ypoints,color = 'r',marker = 'o',linestyle = '-.',mec = 'b',mfc = 'b')
7  plt.show()
```

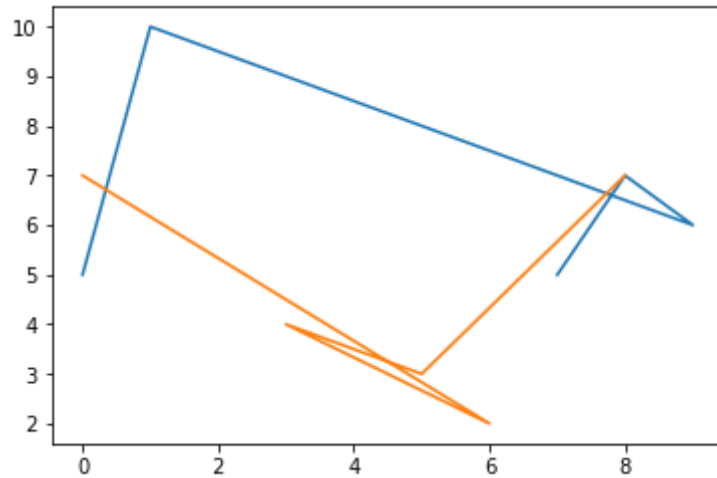


```
In [13]: 1  ## Plot lines with custom colors for markers in pyplot
2
3  xpoints = np.array([0,1,9,8,7])
4  ypoints = np.array([5,10,6,7,5])
5
6  plt.plot(xpoints,ypoints,color = '#00FF00',marker = 'o',linestyle = '-.',mec = 'b',mfc = 'r',linewidth = 5)
7  plt.show()
```



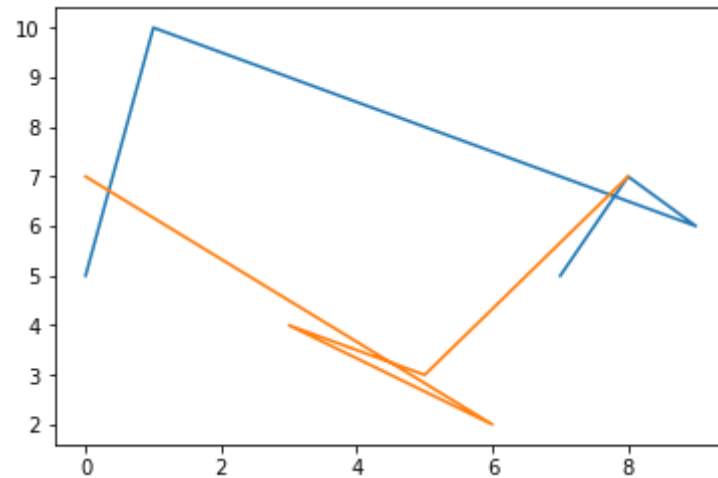
In [14]:

```
1  ## Plot multiple lines in pyplot
2
3  x1 = np.array([0,1,9,8,7])
4  y1 = np.array([5,10,6,7,5])
5  x2 = np.array([0,6,3,5,8])
6  y2 = np.array([7,2,4,3,7])
7
8  plt.plot(x1,y1,x2,y2)
9  plt.show()
```



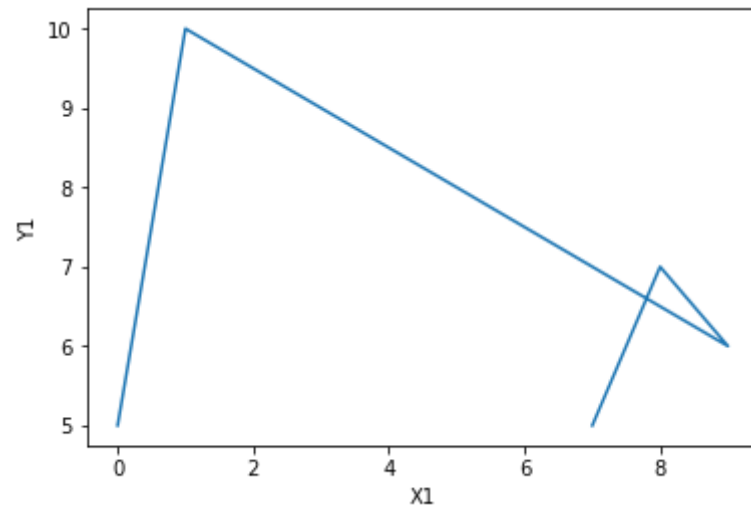
In [15]:

```
1 x1 = np.array([0,1,9,8,7])
2 y1 = np.array([5,10,6,7,5])
3 x2 = np.array([0,6,3,5,8])
4 y2 = np.array([7,2,4,3,7])
5
6
7 plt.plot(x1,y1)
8 plt.plot(x2,y2)
9 plt.show()
```



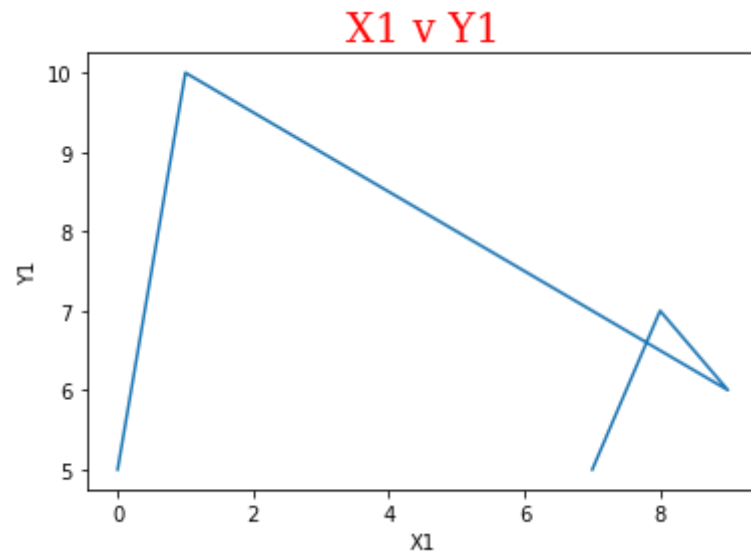
In [16]:

```
1  ## Name the axes:  
2  x1 = np.array([0,1,9,8,7])  
3  y1 = np.array([5,10,6,7,5])  
4  
5  plt.plot(x1,y1)  
6  
7  plt.xlabel("X1")  
8  plt.ylabel("Y1")  
9  plt.show()
```



In [17]:

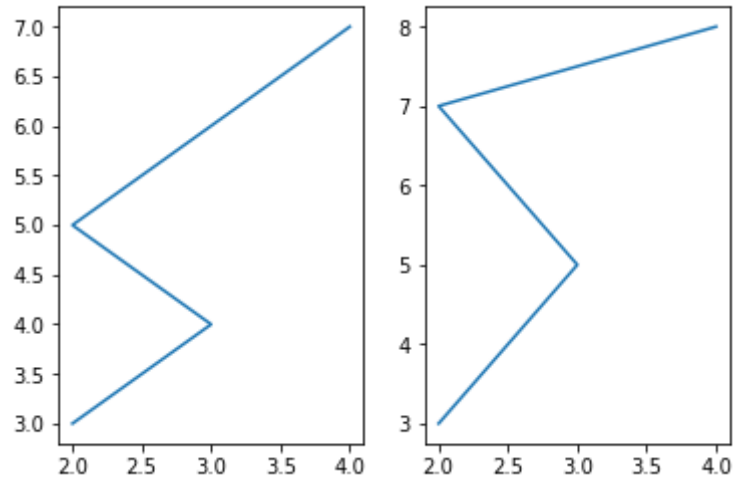
```
1  ## Name the axes and titles:
2  x1 = np.array([0,1,9,8,7])
3  y1 = np.array([5,10,6,7,5])
4
5  plt.plot(x1,y1)
6
7  fd = {'family':'serif','color':'red','size':20}
8
9  plt.xlabel("X1")
10 plt.ylabel("Y1")
11 plt.title("X1 v Y1",fd)
12 plt.show()
```





```
In [18]: 1  ## 2 plots one after the other
2
3  ##
4
5  x = np.array([2,3,2,4])
6  y = np.array([3,4,5,7])
7
8  plt.subplot(1,2,1)
9  plt.plot(x,y)
10 ##
11 x = np.array([2,3,2,4])
12 y = np.array([3,5,7,8])
13
14 plt.subplot(1,2,2)
15 plt.plot(x,y)
16
17
```

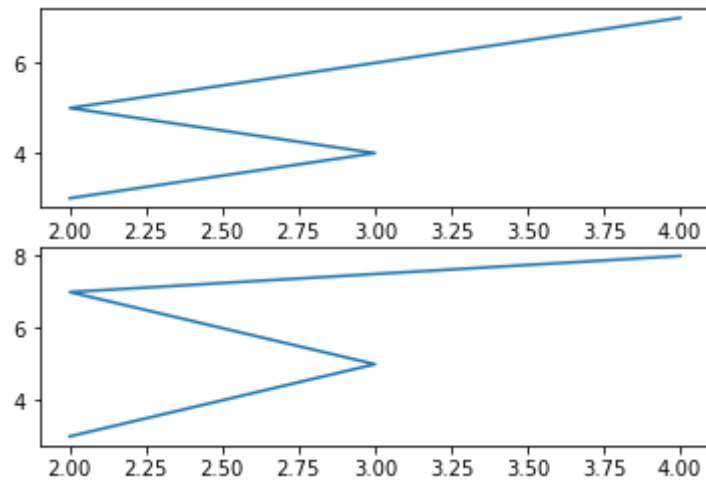
Out[18]: [<matplotlib.lines.Line2D at 0x1edff1e1148>]



In [19]:

```
1  ## 2 plots one after the other
2
3  ##
4
5  x = np.array([2,3,2,4])
6  y = np.array([3,4,5,7])
7
8  plt.subplot(2,1,1) ## Row , Column , plot number
9  plt.plot(x,y)
10 ##
11 x = np.array([2,3,2,4])
12 y = np.array([3,5,7,8])
13
14 plt.subplot(2,1,2)
15 plt.plot(x,y)
```

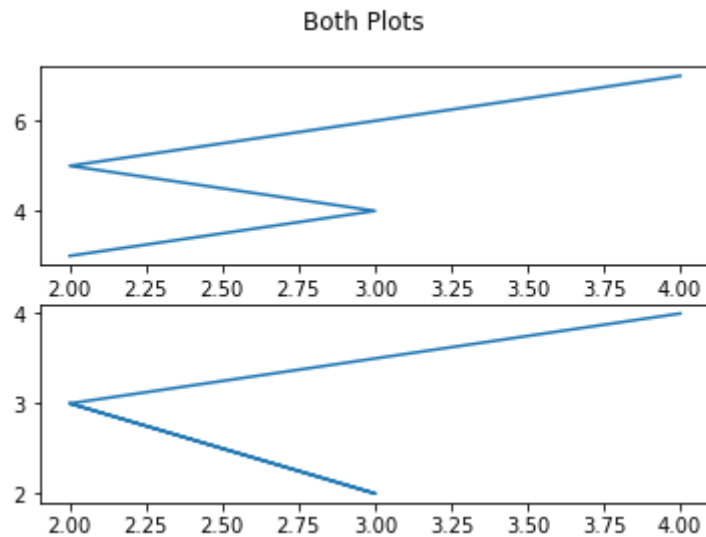
Out[19]: [&lt;matplotlib.lines.Line2D at 0x1edff0b7248&gt;]



In [20]:

```
1  ## 2 plots one after the other
2
3  ##
4
5  x = np.array([2,3,2,4])
6  y = np.array([3,4,5,7])
7
8  plt.subplot(2,1,1) ## Row , Column , plot number
9  plt.plot(x,y)
10 ##
11 x = np.array([2,3,2,4])
12 y = np.array([3,2,3,4])
13
14 plt.subplot(2,1,2)
15 plt.plot(x,y)
16
17 plt.suptitle("Both Plots")
```

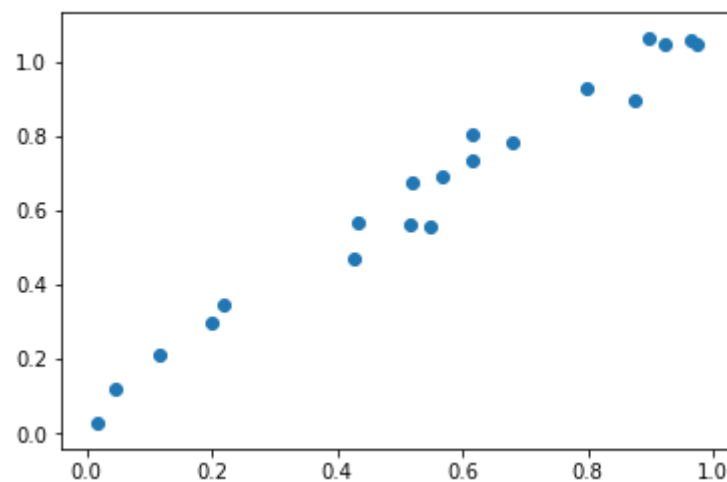
Out[20]: Text(0.5,0.98,'Both Plots')



```
In [21]: 1 ##
2 x = np.array([2,3,2,4])
3 y = np.array([3,4,5,7])
4 ##
5 x = np.array([2,3,2,4])
6 y = np.array([3,5,7,8])
7 ##
8 x1 = np.array([0,1,9,8,7])
9 y1 = np.array([5,10,6,7,5])
10
```

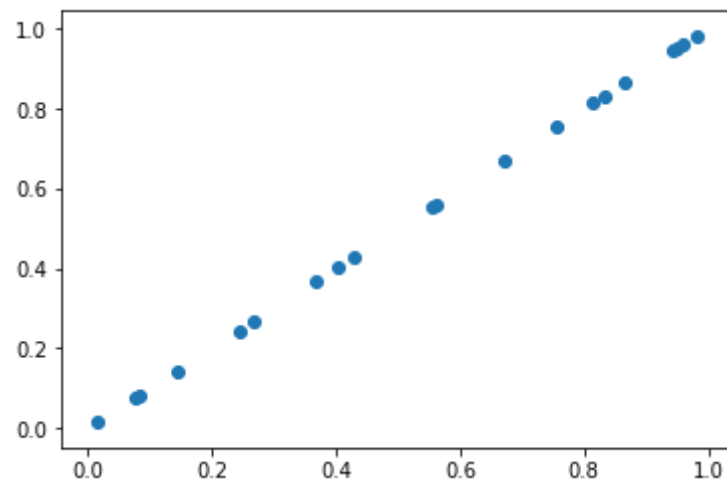
```
In [33]: 1 ## Scatter Plot
2
3 x = np.random.rand(20)
4 y = 0.1*np.random.rand(20) + x + 0.1*np.random.rand(20)
5
6 plt.scatter(x,y)
7
8 ## Correlation -> Linear relationship between 2 variable
```

Out[33]: <matplotlib.collections.PathCollection at 0x1edfefe19c8>



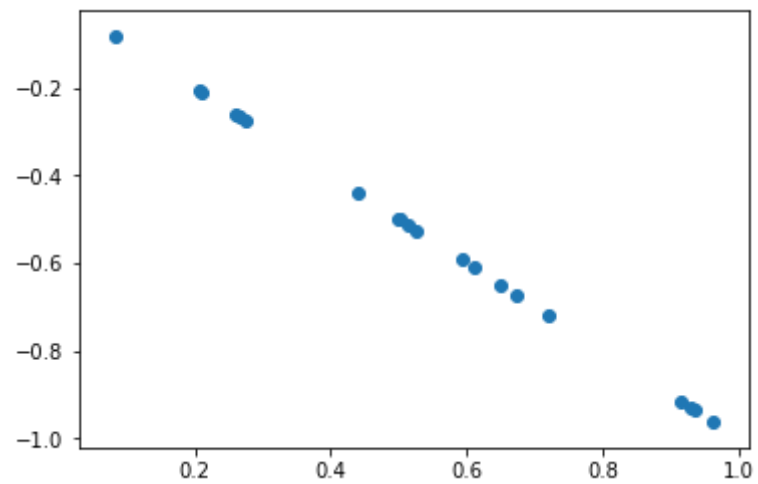
```
In [34]: 1  ## Scatter Plot corr ~ 1
2
3  x = np.random.rand(20)
4  y = x
5
6  plt.scatter(x,y)
7
8
9
```

Out[34]: <matplotlib.collections.PathCollection at 0x1edff293188>



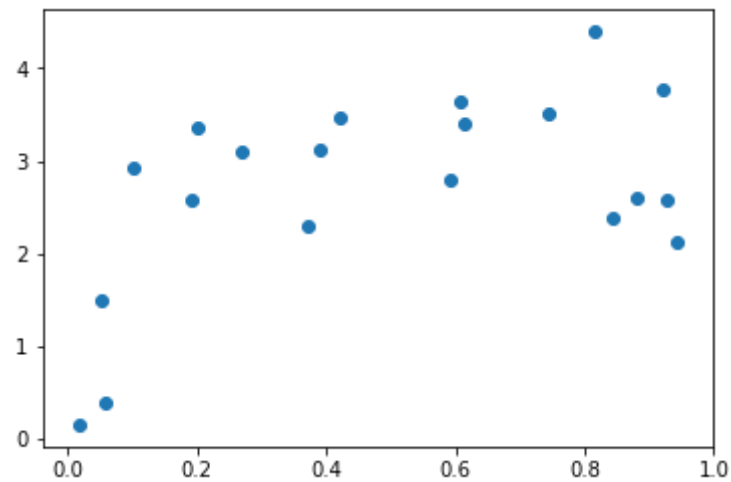
```
In [35]: 1  ## Scatter Plot corr ~ -1
2
3  x = np.random.rand(20)
4  y = -x
5
6  plt.scatter(x,y)
7
```

Out[35]: <matplotlib.collections.PathCollection at 0x1edff2eaf48>



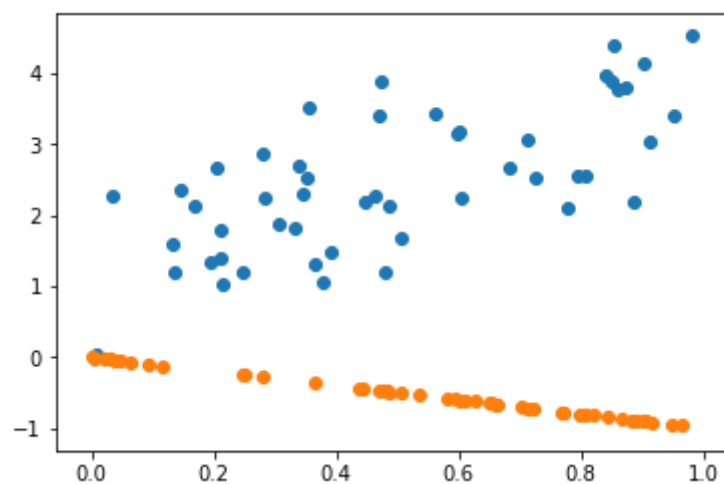
```
In [44]: 1  ## Scatter Plot corr ~ 0
2
3  x = np.random.rand(20)
4  y = 2*x + 3*np.random.rand(20)
5
6  plt.scatter(x,y)
7
```

Out[44]: <matplotlib.collections.PathCollection at 0x1edff63eb88>



```
In [39]: 1  ## Scatter Plot
2  x = np.random.rand(50)
3  y = 2*x + 3*np.random.rand(50)
4
5  plt.scatter(x,y)
6
7
8  x = np.random.rand(50)
9  y = -x
10
11 plt.scatter(x,y)
```

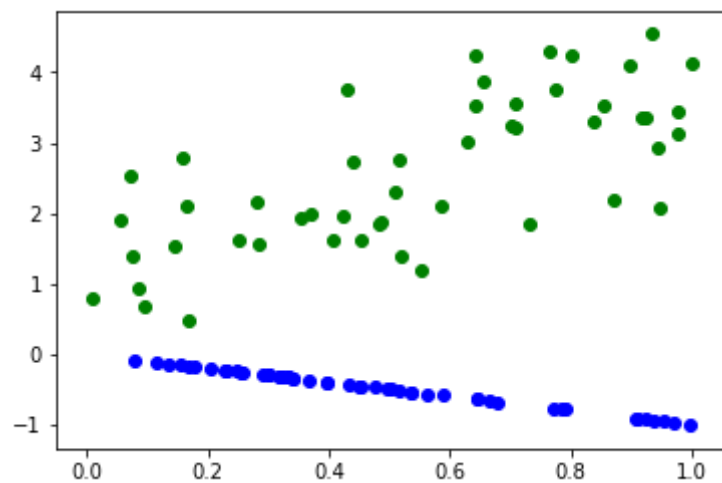
Out[39]: <matplotlib.collections.PathCollection at 0x1edff46e288>





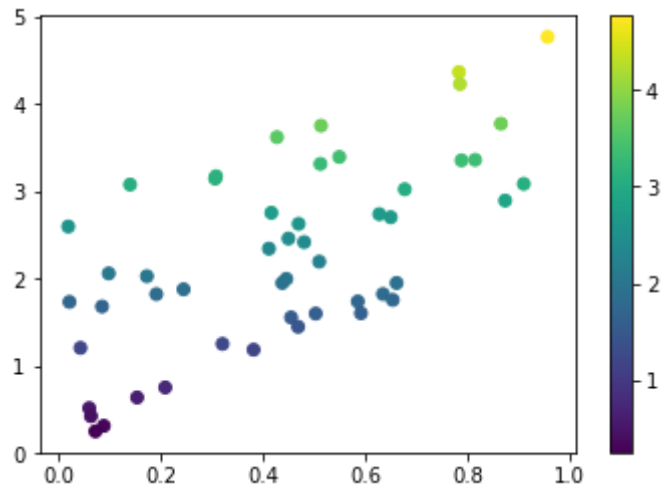
```
In [47]: 1  ## Scatter Plot
2  x = np.random.rand(50)
3  y = 2*x + 3*np.random.rand(50)
4
5  plt.scatter(x,y,color = 'green')
6
7
8  x = np.random.rand(50)
9  y = -x
10
11 plt.scatter(x,y,color = 'blue')
```

Out[47]: <matplotlib.collections.PathCollection at 0x1edff86e1c8>



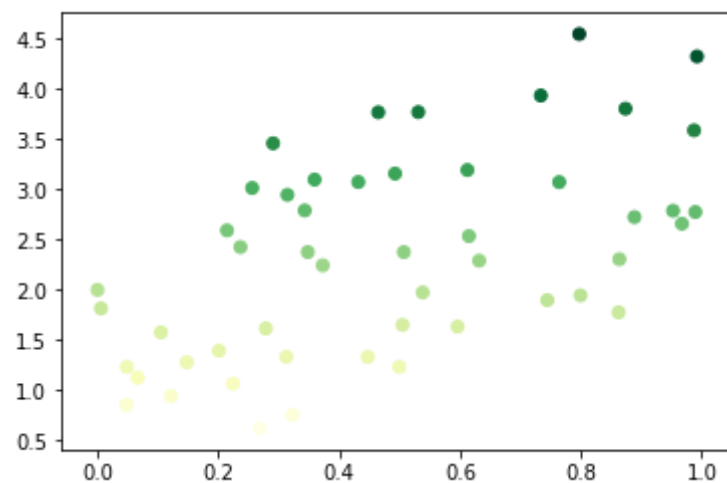
In [57]:

```
1  ## Scatter Plot with Color Map
2  ## Scatter Plot
3  x = np.random.rand(50)
4  y = 2*x + 3*np.random.rand(50)
5
6  plt.scatter(x ## Our X array
7              ,y ## Our Y array
8              ,c = y ## The variable whose values will determine the color in the colormap
9              ,cmap = 'viridis')
10 plt.colorbar()
11 plt.show()
12
```



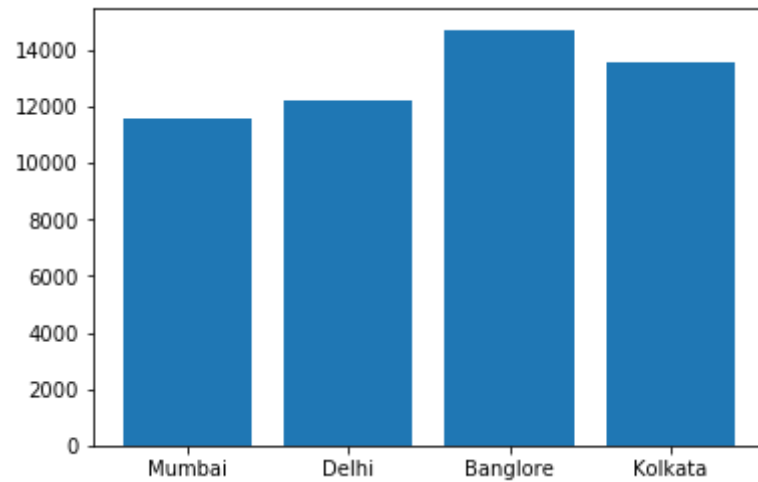
```
In [55]: 1  ## Scatter Plot with Color Map
        2  ## Scatter Plot
        3  x = np.random.rand(50)
        4  y = 2*x + 3*np.random.rand(50)
        5
        6  plt.scatter(x,y,c = y,cmap = 'YlGn')
```

Out[55]: <matplotlib.collections.PathCollection at 0x1edffb19608>



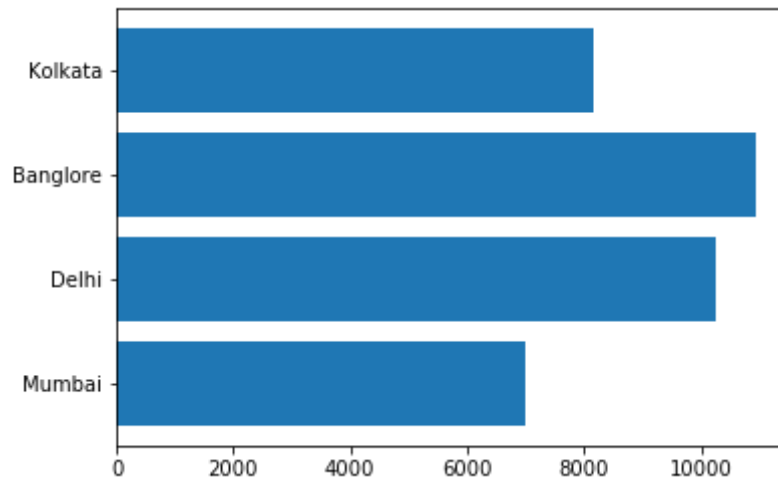
```
In [66]: 1  ## Bar Chart:
2
3  x = np.array(["Mumbai", "Delhi", "Banglore", "Kolkata"])
4  y = np.random.randint(1000,20000,4) ## Vales between 0,20000
5
6  plt.bar(x,y)
```

Out[66]: <BarContainer object of 4 artists>



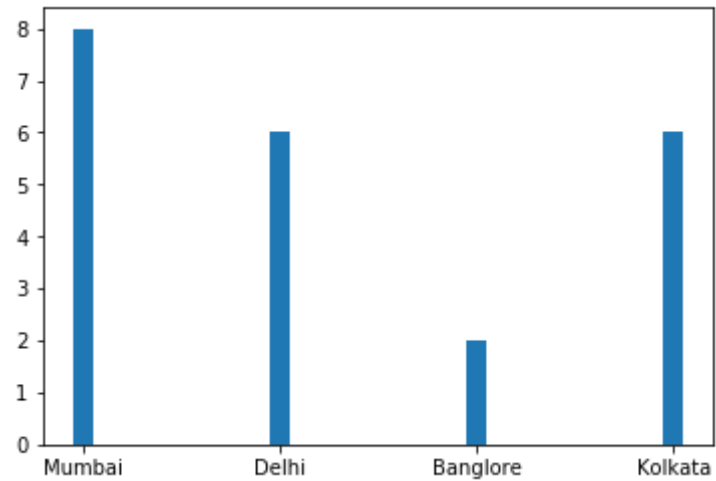
```
In [67]: 1  ## Bar Chart:
2
3  x = np.array(["Mumbai", "Delhi", "Banglore", "Kolkata"])
4  y = np.random.randint(1000, 20000, 4)  ## Vales between 0, 20000
5
6  plt.barh(x, y)
```

Out[67]: <BarContainer object of 4 artists>



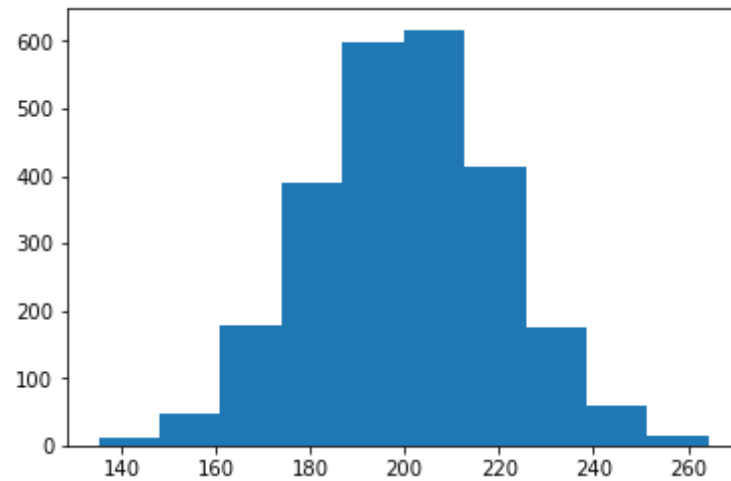
```
In [71]: 1  ## Bar Chart:
2
3  x = np.array(["Mumbai", "Delhi", "Banglore", "Kolkata"])
4  y = np.random.randint(0,10,4)  ## Vales between 0,20000
5
6  plt.bar(x,y,width = 0.1)
```

Out[71]: <BarContainer object of 4 artists>

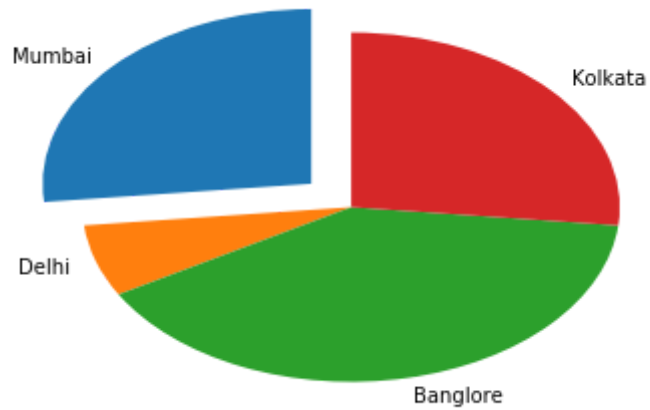


In [79]:

```
1  ## Histograms
2
3  x = np.random.normal(200,20,2500)
4  plt.hist(x)
5  plt.show()
```



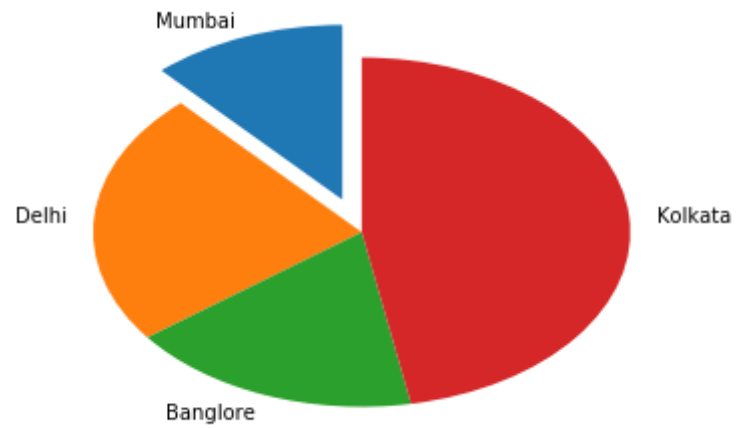
```
In [88]: 1  ## Pie:
2
3  x = np.array(["Mumbai","Delhi","Banglore","Kolkata"])
4  y = np.random.randint(0,10,4)  ## Vales between 0,20000
5  explode = np.array([0.2,0,0,0])
6
7
8  plt.pie(y,labels = x,startangle=90,explode=explode)
9  plt.show()
```





In [92]:

```
1  ## Pie:
2
3  x = np.array(["Mumbai","Delhi","Banglore","Kolkata"])
4  y = np.random.randint(0,10,4) ## Vales between 0,20000
5  explode = np.array([0.2,0,0,0])
6
7
8  plt.pie(y,labels = x,startangle=90,explode=explode,shadow = False)
9  #plt.legend()
10 plt.show()
```



In [ ]:

1

