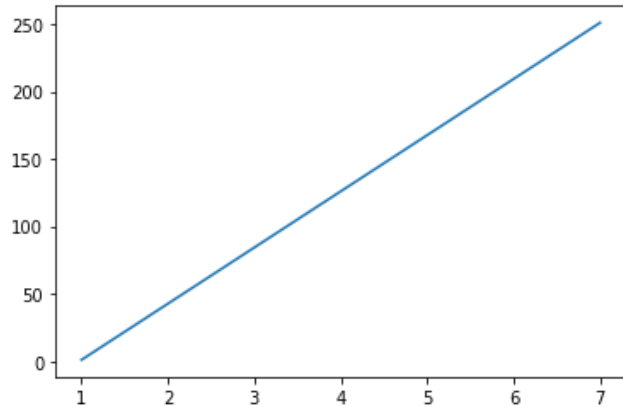


In [3]: *#Draw a Line in a diagram from position (0,0) to position (6,250):*

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

xpoints = np.array([1, 7])
ypoints = np.array([1, 251])

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

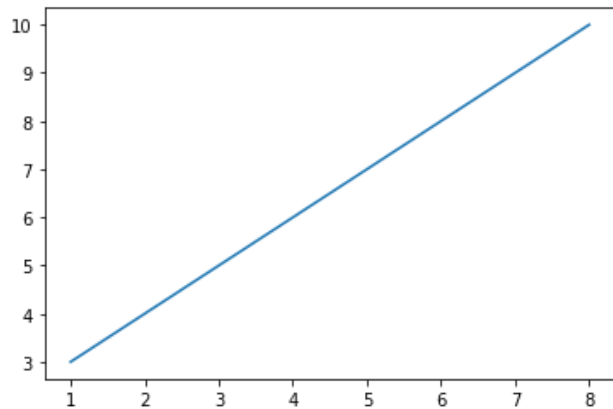


In [4]: *#Draw a Line in a diagram from position (1, 3) to position (8, 10):*

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

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

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

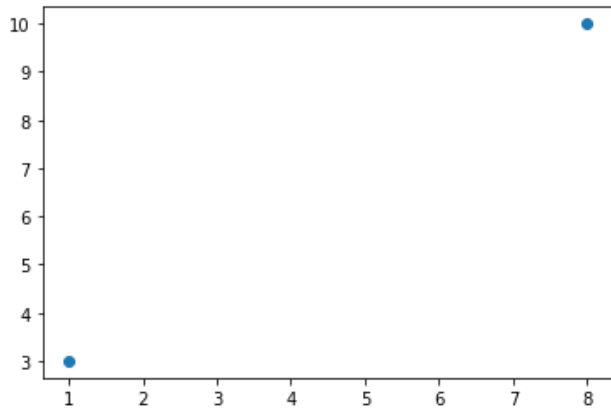


In [4]: *#Draw two points in the diagram, one at position (1, 3) and one in position (8, 10):*

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

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

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

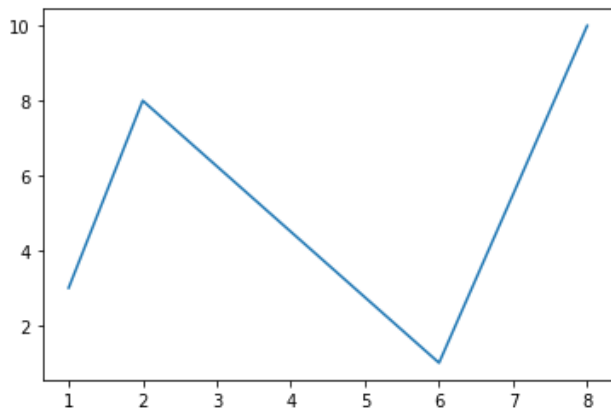


In [6]: *#Draw a Line in a diagram from position (1, 3) to (2, 8) then
#to (6, 1) and finally to position (8, 10):*

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

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

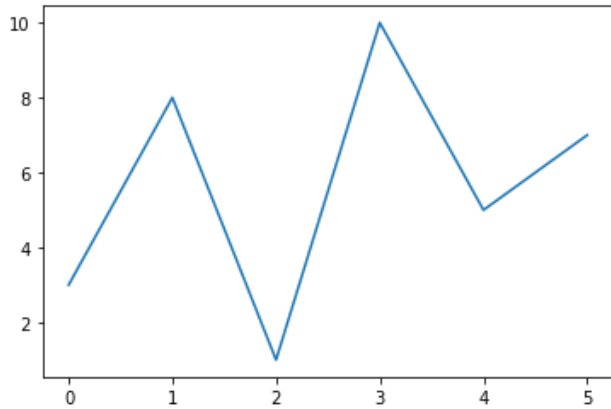
```
plt.plot(xpoints, ypoints)
plt.show()
```



```
In [7]: #Plotting without x-points:
import matplotlib.pyplot as plt
import numpy as np

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

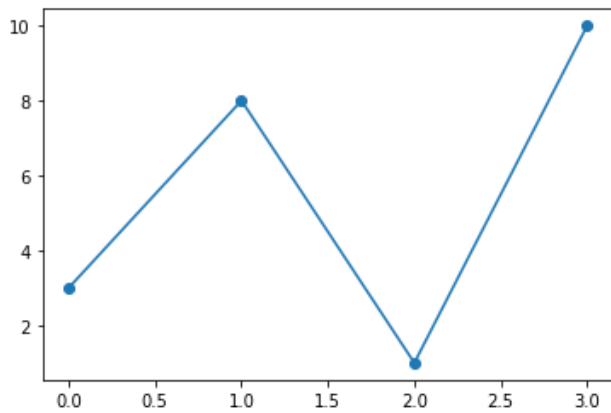
plt.plot(ypoints)
plt.show()
```



```
In [8]: #Mark each point with a circle:
import matplotlib.pyplot as plt
import numpy as np

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

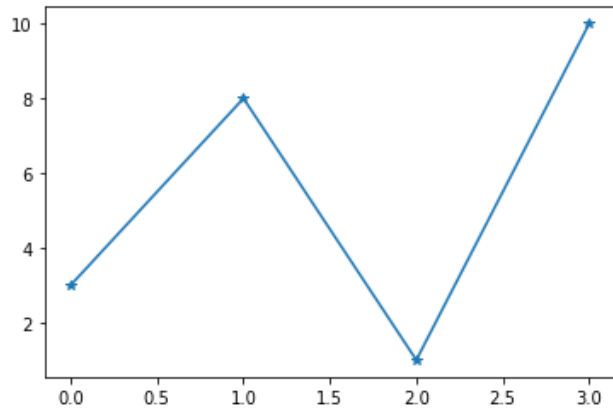
plt.plot(ypoints, marker = 'o')
plt.show()
```



```
In [11]: #Mark each point with a star:
import matplotlib.pyplot as plt
import numpy as np

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

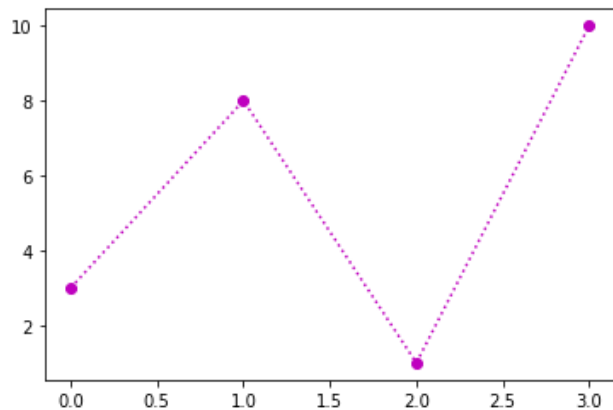
plt.plot(ypoints, marker = '*')
plt.show()
```



```
In [8]: #Mark each point with a circle and color:
import matplotlib.pyplot as plt
import numpy as np

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

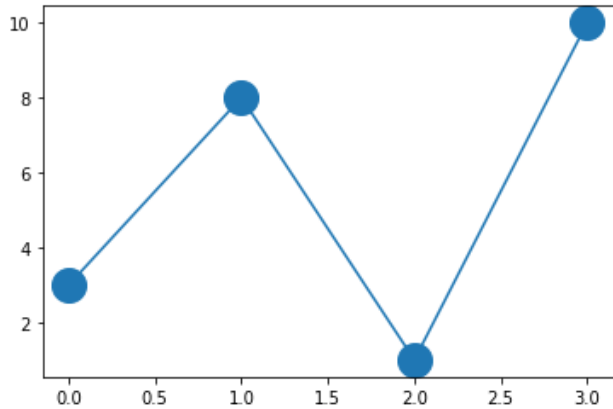
plt.plot(ypoints, 'o:m') #support only 'r'-Red, 'g'-Green, 'b'-Blue, 'c'-Cyan, 'm'-Magenta, 'y'-Yellow, 'k'-Black, 'w'-White
plt.show()
```



```
In [29]: #Set the size of the markers to 20:
import matplotlib.pyplot as plt
import numpy as np

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

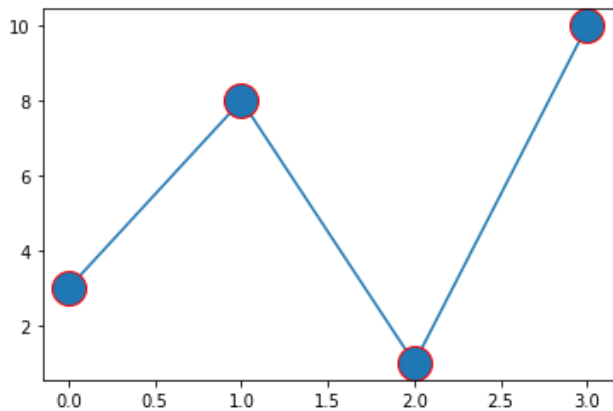
plt.plot(ypoints, marker = 'o', ms = 20)
plt.show()
```



```
In [30]: #Set the EDGE color to red:
import matplotlib.pyplot as plt
import numpy as np

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

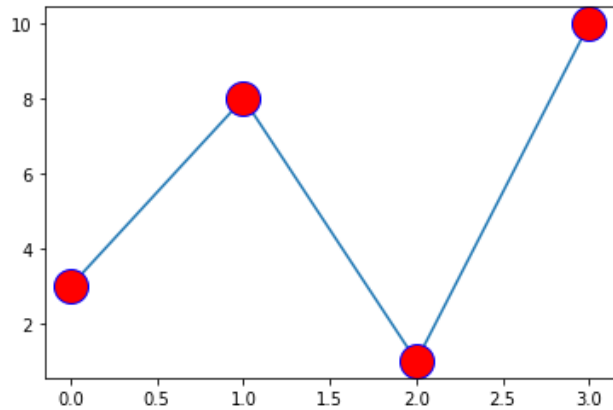
plt.plot(ypoints, marker = 'o', ms = 20, mec = 'r')
plt.show()
```



```
In [33]: #Set the FACE color to red:
import matplotlib.pyplot as plt
import numpy as np

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

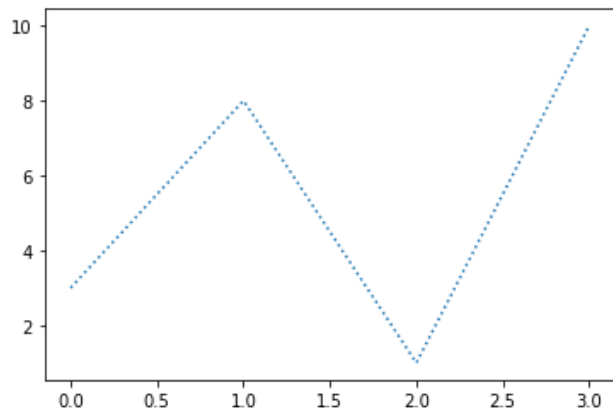
plt.plot(ypoints, marker = 'o', ms = 20, mec = 'b', mfc = 'r')
plt.show()
```



```
In [5]: #Use a dotted Line:
import matplotlib.pyplot as plt
import numpy as np

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

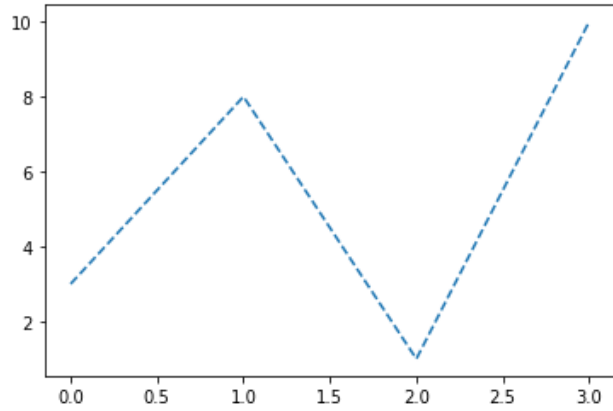
plt.plot(ypoints, linestyle = 'dotted')
plt.show()
```



```
In [6]: #Use a dashed Line:
import matplotlib.pyplot as plt
import numpy as np

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

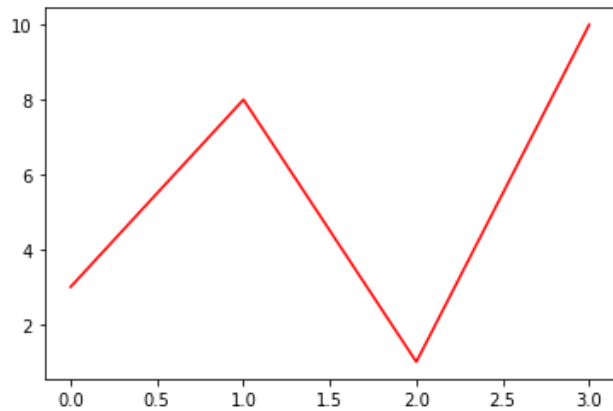
plt.plot(ypoints, linestyle = 'dashed')
plt.show()
```



```
In [7]: #Set the Line color to red:
import matplotlib.pyplot as plt
import numpy as np

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

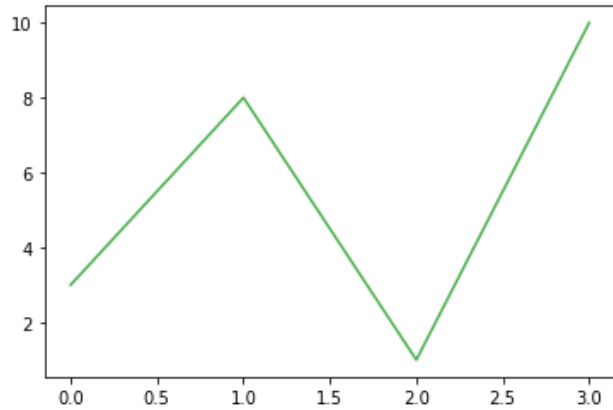
plt.plot(ypoints, color = 'r')
plt.show()
```



```
In [9]: #Set the line color code:
import matplotlib.pyplot as plt
import numpy as np

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

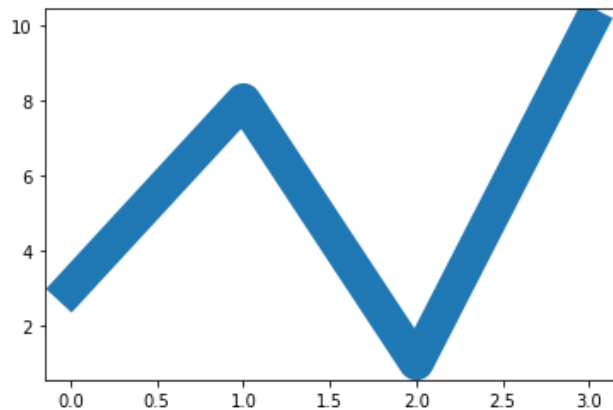
plt.plot(ypoints, c = '#4CAF50')
plt.show()
```



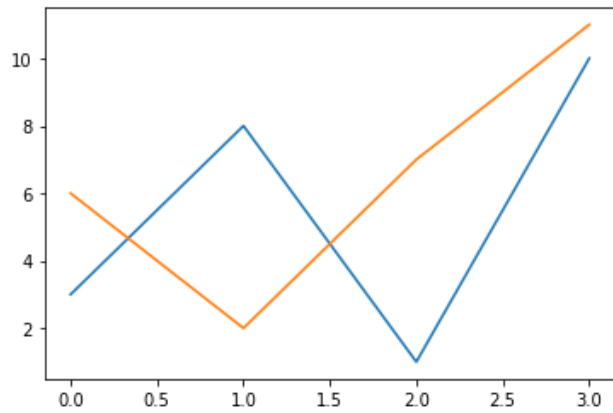
```
In [10]: #Plot with a 20.5pt wide line:
import matplotlib.pyplot as plt
import numpy as np

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

plt.plot(ypoints, linewidth = '20.5')
plt.show()
```




```
In [11]: #Multiple Lines
import matplotlib.pyplot as plt
import numpy as np
y1 = np.array([3, 8, 1, 10])
y2 = np.array([6, 2, 7, 11])
plt.plot(y1)
plt.plot(y2)
plt.show()
```

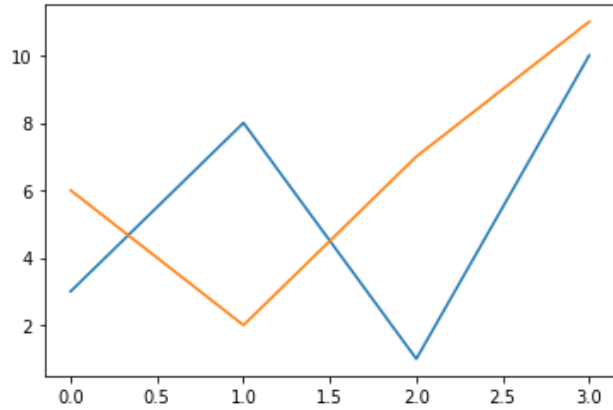


In [12]: *#Draw two lines by specifying the x- and y-point values for both lines:*

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

```
x1 = np.array([0, 1, 2, 3])
y1 = np.array([3, 8, 1, 10])
x2 = np.array([0, 1, 2, 3])
y2 = np.array([6, 2, 7, 11])
```

```
plt.plot(x1, y1, x2, y2)
plt.show()
```



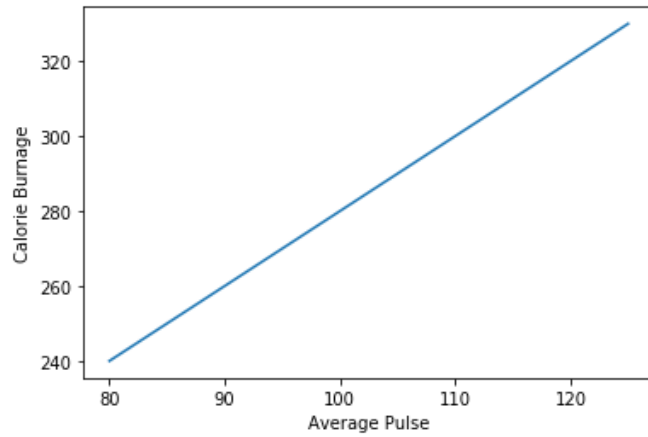
```
In [13]: #Add labels to the x- and y-axis:
import numpy as np
import matplotlib.pyplot as plt

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)

plt.xlabel("Average Pulse")
plt.ylabel("Calorie Burnage")

plt.show()
```



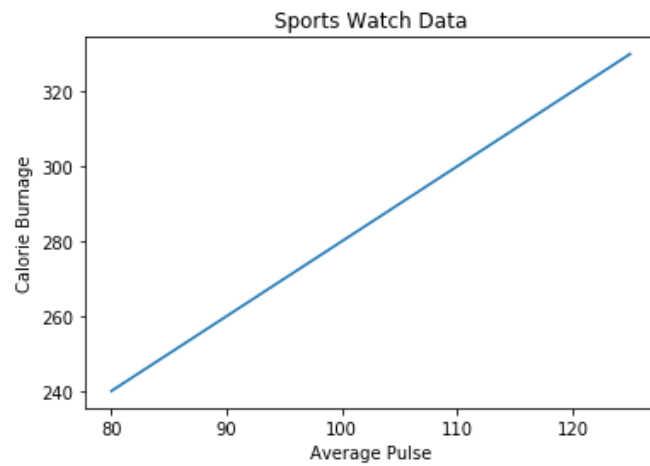
```
In [14]: #Add a plot title and labels for the x- and y-axis:
import numpy as np
import matplotlib.pyplot as plt

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)

plt.title("Sports Watch Data")
plt.xlabel("Average Pulse")
plt.ylabel("Calorie Burnage")

plt.show()
```



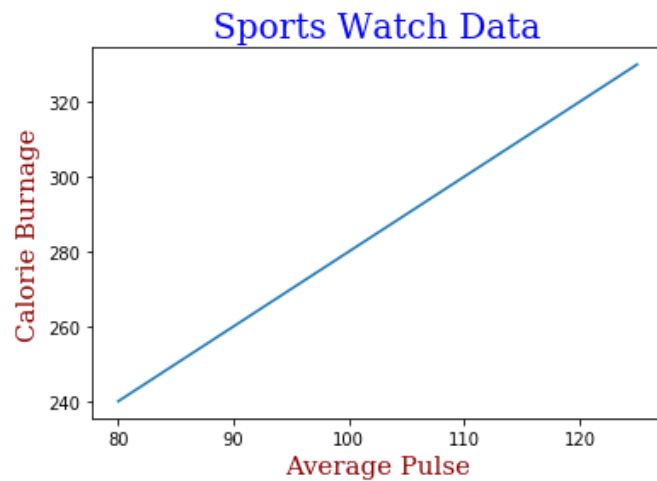
```
In [15]: #Set font properties for the title and labels:
import numpy as np
import matplotlib.pyplot as plt

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])

font1 = {'family':'serif','color':'blue','size':20}
font2 = {'family':'serif','color':'darkred','size':15}

plt.title("Sports Watch Data", fontdict = font1)
plt.xlabel("Average Pulse", fontdict = font2)
plt.ylabel("Calorie Burnage", fontdict = font2)

plt.plot(x, y)
plt.show()
```

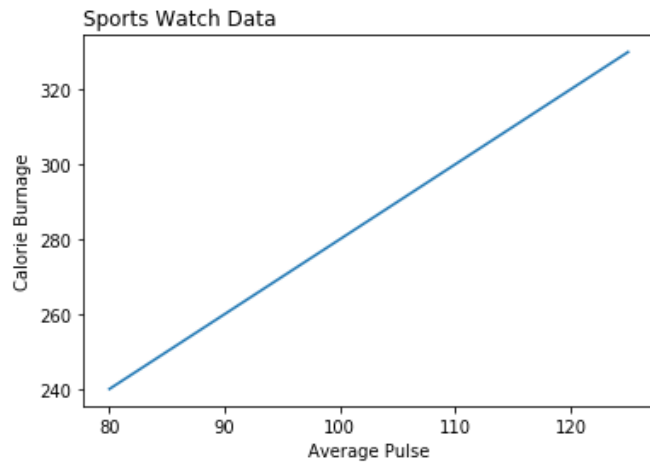


```
In [16]: #Position the title to the left:
import numpy as np
import matplotlib.pyplot as plt

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.title("Sports Watch Data", loc = 'left')
plt.xlabel("Average Pulse")
plt.ylabel("Calorie Burnage")

plt.plot(x, y)
plt.show()
```



```
In [2]: #Add grid lines to the plot:
import numpy as np
import matplotlib.pyplot as plt

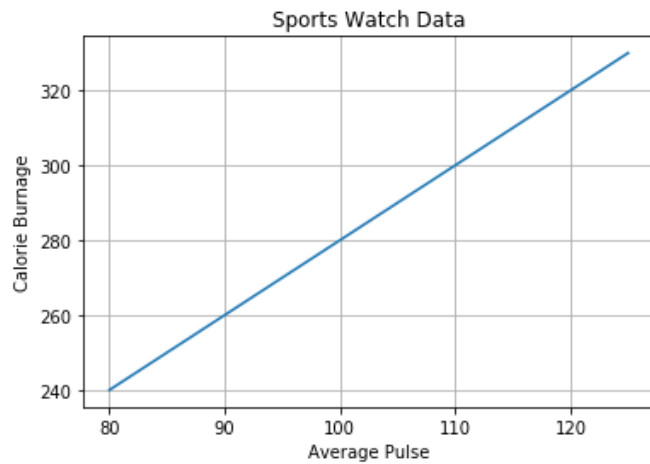
x = np.array([80, 85, 90, 95, 100, 105, 110, 115, 120, 125])
y = np.array([240, 250, 260, 270, 280, 290, 300, 310, 320, 330])

plt.title("Sports Watch Data")
plt.xlabel("Average Pulse")
plt.ylabel("Calorie Burnage")

plt.plot(x, y)

plt.grid()

plt.show()
```



```
In [3]: #Display only grid lines for the x-axis:
import numpy as np
import matplotlib.pyplot as plt

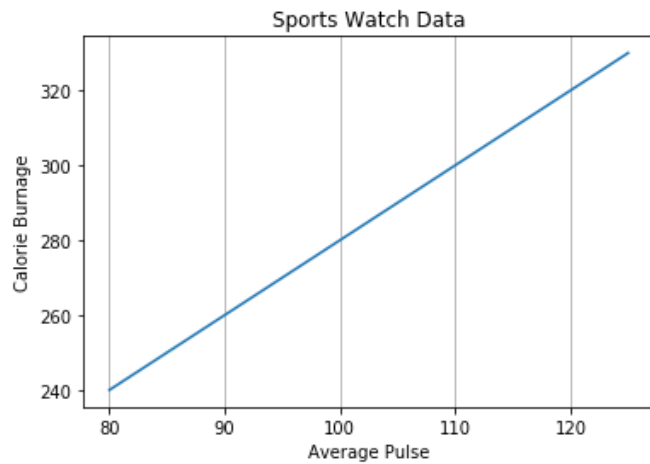
x = np.array([80, 85, 90, 95, 100, 105, 110, 115, 120, 125])
y = np.array([240, 250, 260, 270, 280, 290, 300, 310, 320, 330])

plt.title("Sports Watch Data")
plt.xlabel("Average Pulse")
plt.ylabel("Calorie Burnage")

plt.plot(x, y)

plt.grid(axis = 'x')

plt.show()
```




```
In [4]: #Display only grid lines for the y-axis:
import numpy as np
import matplotlib.pyplot as plt

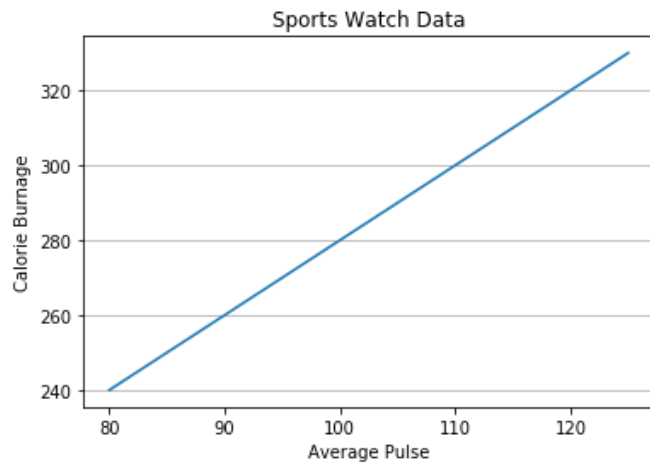
x = np.array([80, 85, 90, 95, 100, 105, 110, 115, 120, 125])
y = np.array([240, 250, 260, 270, 280, 290, 300, 310, 320, 330])

plt.title("Sports Watch Data")
plt.xlabel("Average Pulse")
plt.ylabel("Calorie Burnage")

plt.plot(x, y)

plt.grid(axis = 'y')

plt.show()
```



```
In [5]: #Set the Line properties of the grid:
import numpy as np
import matplotlib.pyplot as plt

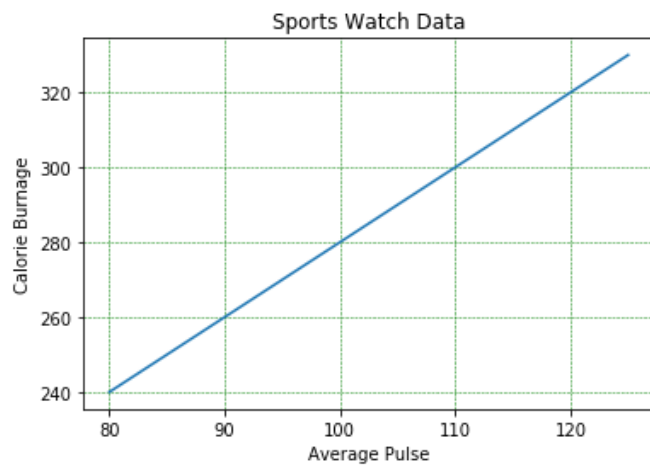
x = np.array([80, 85, 90, 95, 100, 105, 110, 115, 120, 125])
y = np.array([240, 250, 260, 270, 280, 290, 300, 310, 320, 330])

plt.title("Sports Watch Data")
plt.xlabel("Average Pulse")
plt.ylabel("Calorie Burnage")

plt.plot(x, y)

plt.grid(color = 'green', linestyle = '--', linewidth = 0.5)

plt.show()
```



```
In [6]: #Draw 2 plots:
import matplotlib.pyplot as plt
import numpy as np

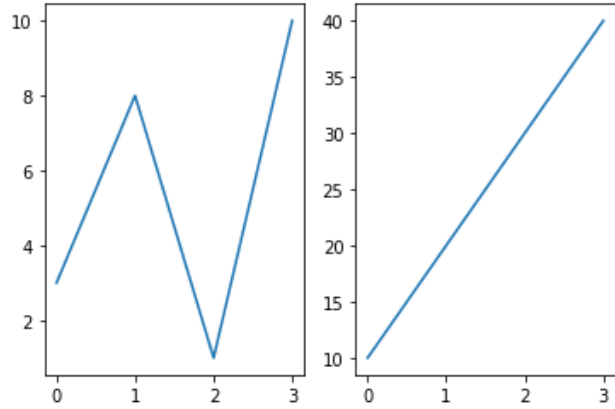
#plot 1:
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])

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

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

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

plt.show()
```



In [7]: *#Draw 2 plots on top of each other:*

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

#plot 1:

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

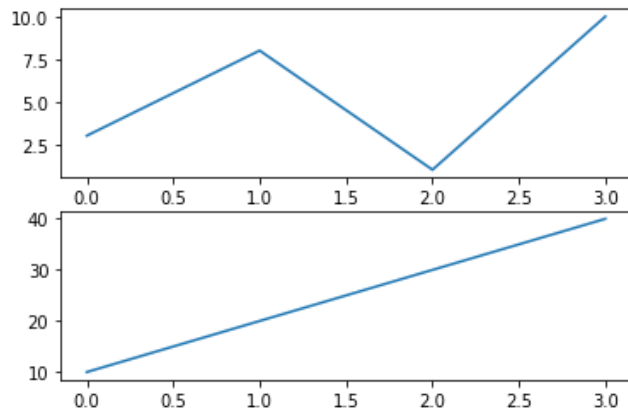
```
plt.subplot(2, 1, 1)
plt.plot(x,y)
```

#plot 2:

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

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

```
plt.show()
```



```
In [8]: #Draw 6 plots:
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)

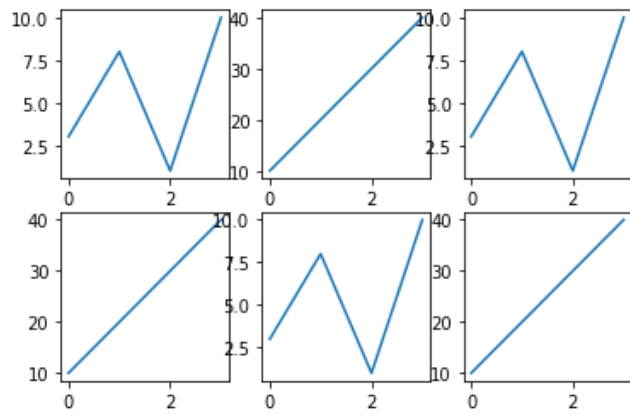
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])

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)

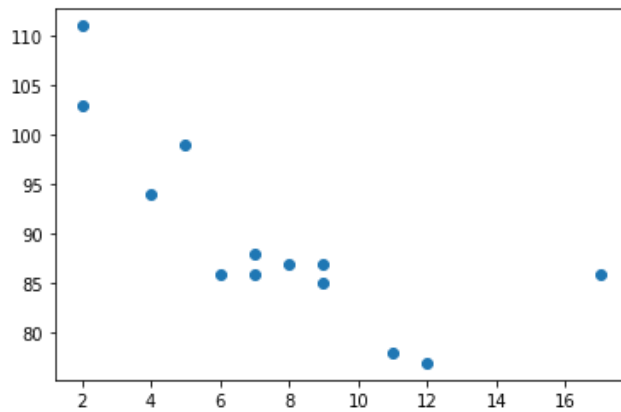
plt.show()
```



```
In [9]: #A simple scatter plot:
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)
plt.show()
```

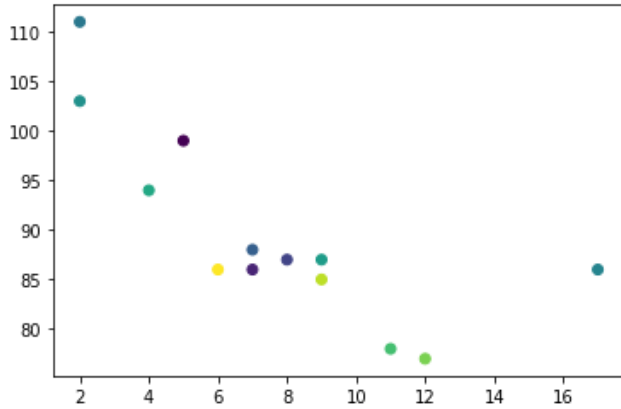


```
In [10]: #Create a color array, and specify a colormap in the scatter plot:
import matplotlib.pyplot as plt
import numpy as np

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([0, 10, 20, 30, 40, 45, 50, 55, 60, 70, 80, 90, 100])

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

plt.show()
```



In [11]: *#Create random arrays with 100 values for x-points, y-points, colors and sizes:*

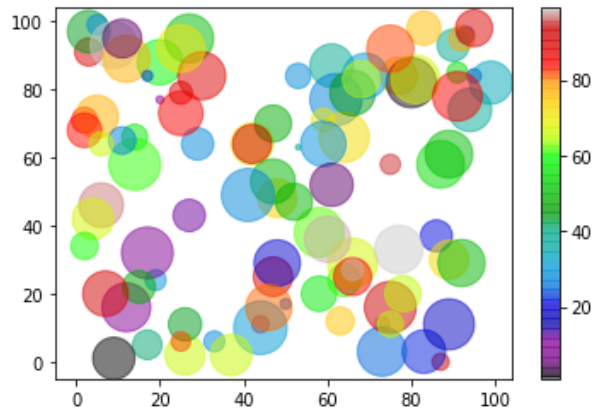
```
import matplotlib.pyplot as plt
import numpy as np

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')

plt.colorbar()

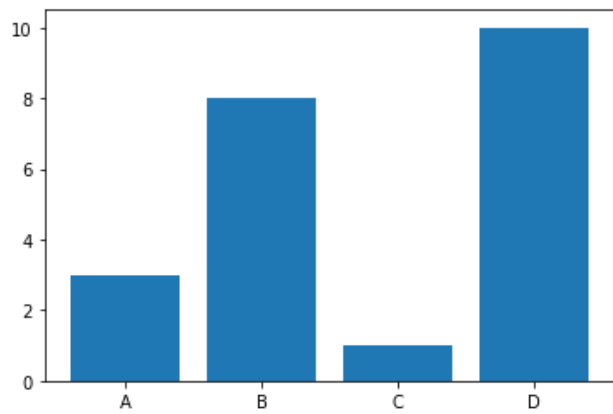
plt.show()
```




```
In [12]: #Draw 4 bars:
import matplotlib.pyplot as plt
import numpy as np

x = np.array(["A", "B", "C", "D"])
y = np.array([3, 8, 1, 10])

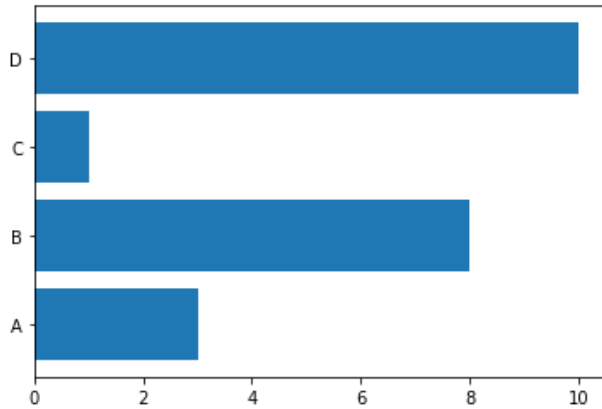
plt.bar(x,y)
plt.show()
```



```
In [13]: #Draw 4 horizontal bars:
import matplotlib.pyplot as plt
import numpy as np

x = np.array(["A", "B", "C", "D"])
y = np.array([3, 8, 1, 10])

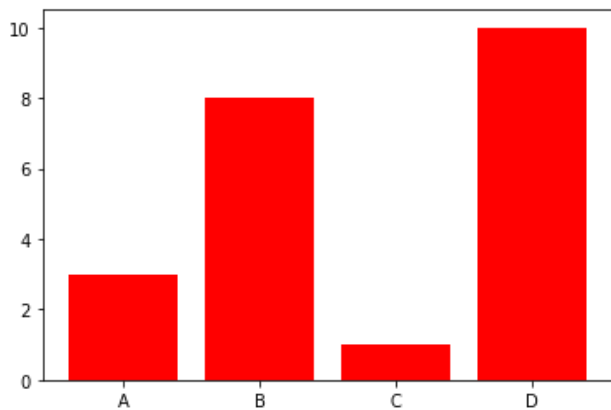
plt.barh(x, y)
plt.show()
```



```
In [14]: #Draw 4 red bars:
import matplotlib.pyplot as plt
import numpy as np

x = np.array(["A", "B", "C", "D"])
y = np.array([3, 8, 1, 10])

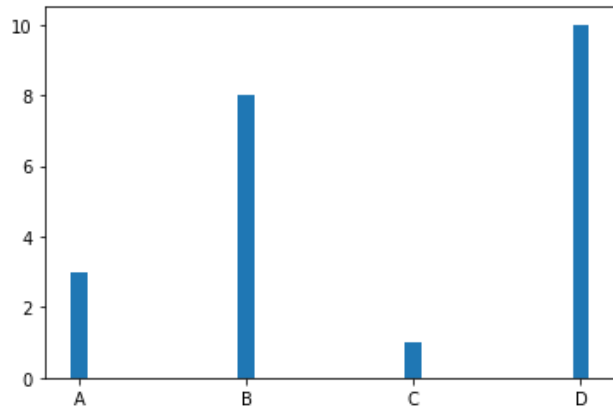
plt.bar(x, y, color = "red")
plt.show()
```



```
In [15]: #Draw 4 very thin bars:
import matplotlib.pyplot as plt
import numpy as np

x = np.array(["A", "B", "C", "D"])
y = np.array([3, 8, 1, 10])

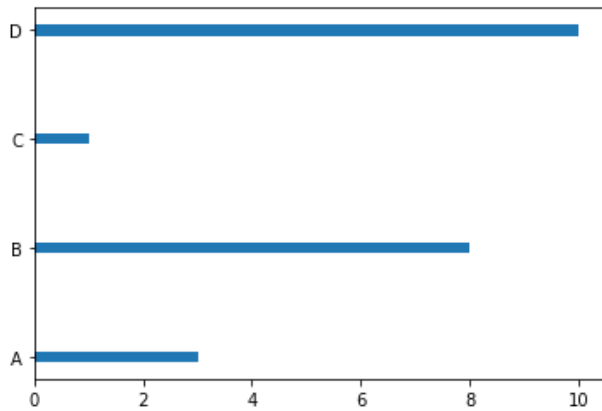
plt.bar(x, y, width = 0.1)
plt.show()
```



```
In [16]: #Draw 4 very thin bars:
import matplotlib.pyplot as plt
import numpy as np

x = np.array(["A", "B", "C", "D"])
y = np.array([3, 8, 1, 10])

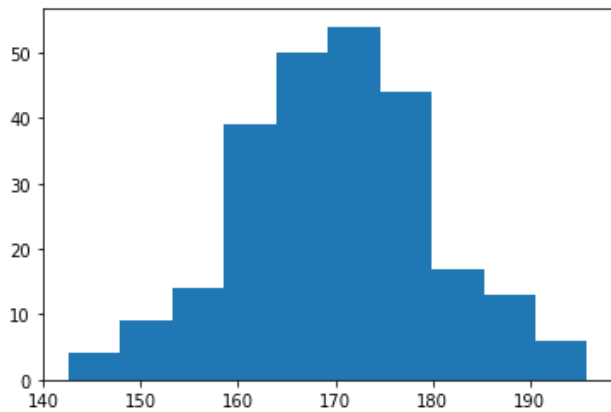
plt.barh(x, y, height = 0.1)
plt.show()
```



```
In [17]: #A simple histogram:
import matplotlib.pyplot as plt
import numpy as np

x = np.random.normal(170, 10, 250)

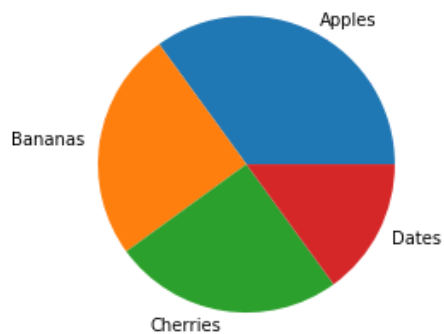
plt.hist(x)
plt.show()
```



```
In [18]: #A simple pie chart:  
import matplotlib.pyplot as plt  
import numpy as np  
  
y = np.array([35, 25, 25, 15])  
  
plt.pie(y)  
plt.show()
```



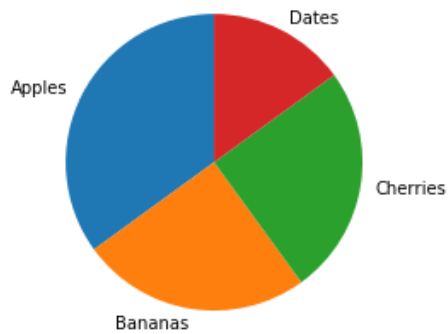
```
In [19]: #A simple pie chart:  
import matplotlib.pyplot as plt  
import numpy as np  
  
y = np.array([35, 25, 25, 15])  
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]  
  
plt.pie(y, labels = mylabels)  
plt.show()
```



```
In [20]: #Start the first wedge at 90 degrees:
import matplotlib.pyplot as plt
import numpy as np

y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]

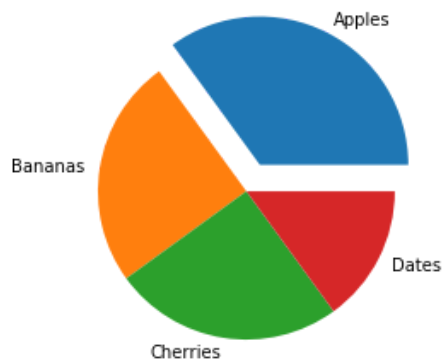
plt.pie(y, labels = mylabels, startangle = 90)
plt.show()
```



```
In [21]: #Pull the "Apples" wedge 0.2 from the center of the pie:
import matplotlib.pyplot as plt
import numpy as np

y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]
myexplode = [0.2, 0, 0, 0]

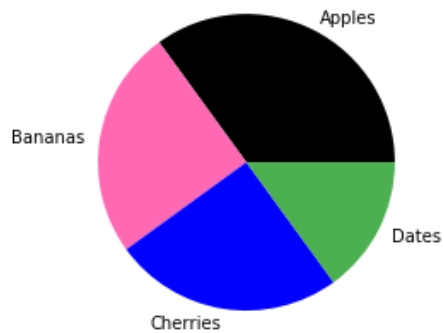
plt.pie(y, labels = mylabels, explode = myexplode)
plt.show()
```



```
In [22]: #Specify a new color for each wedge:
import matplotlib.pyplot as plt
import numpy as np

y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]
mycolors = ["black", "hotpink", "b", "#4CAF50"]

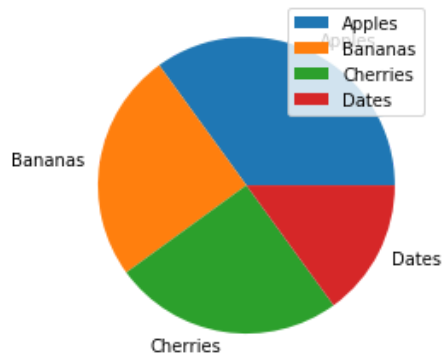
plt.pie(y, labels = mylabels, colors = mycolors)
plt.show()
```



```
In [23]: #Add a Legend:
import matplotlib.pyplot as plt
import numpy as np

y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]

plt.pie(y, labels = mylabels)
plt.legend()
plt.show()
```



In []: