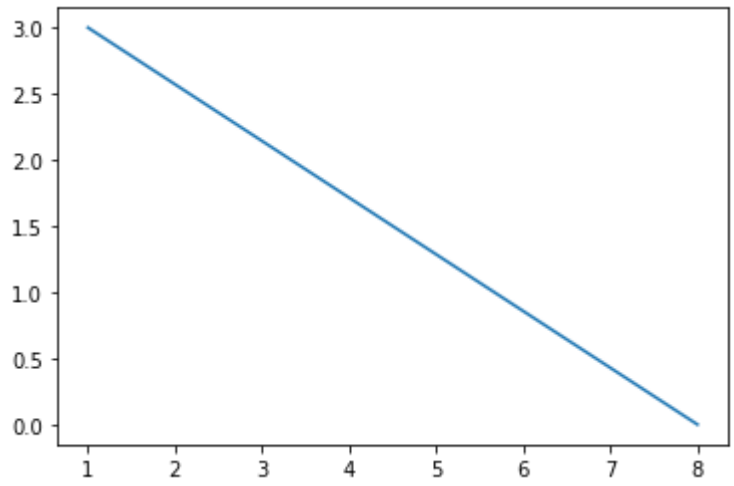
In [ ]:

```
#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([3,0])
plt.plot(xpoints,ypoints)
plt.show()
```



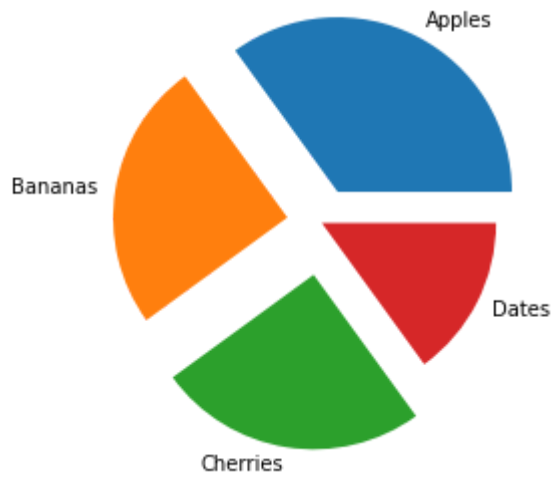
In [ ]:

```
import matplotlib.pyplot as plt
import numpy as np
print(np.arange(10))
```

```
[0 1 2 3 4 5 6 7 8 9]
```

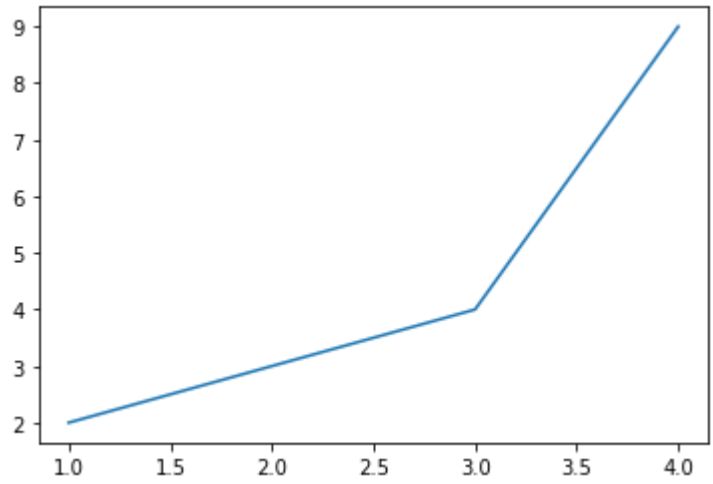
In [ ]:

```
import matplotlib.pyplot as plt
import numpy as np
y = np.array([35, 25, 25, 15])
myexplode = [0.2,0.2, 0.3,0]

plt.pie(y, labels = ["Apples", "Bananas", "Cherries", "Dates"],  explode = myexplode)
plt.show()
```



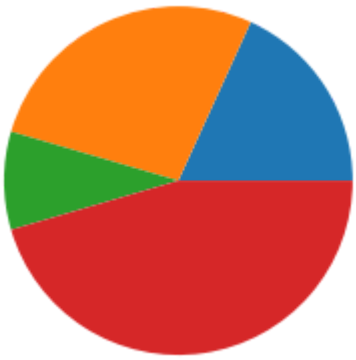
In [ ]:

```
import matplotlib.pyplot as plt
import numpy as np
plt.plot([1,2,3,4],[2,3,4,9])
plt.show()
```



In [ ]:

```
#Pie Plot
data=[20,30,10,50]
from pylab import *
pie(data)
show()
```



In [ ]:

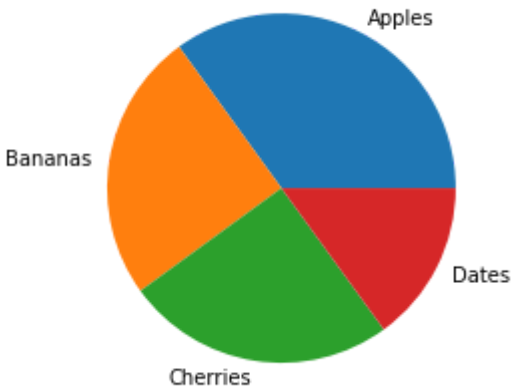
```
#pie chart with explode
import matplotlib.pyplot as plt
import numpy as np
y = np.array([35, 25, 25, 15])
myexplode = [0.2, 0.3, 0.2, 0]
plt.pie(y, labels = ["Apples", "Bananas", "Cherries", "Dates"],
        explode = myexplode)
plt.show()
```

In [ ]:

```
import matplotlib.pyplot as plt
import numpy as np
y = np.array([35, 25, 25, 15])
plt.pie(y)
plt.legend(["Apples", "Bananas","Cherries", "Dates"],title = "Four Fruits:")
plt.show()
```

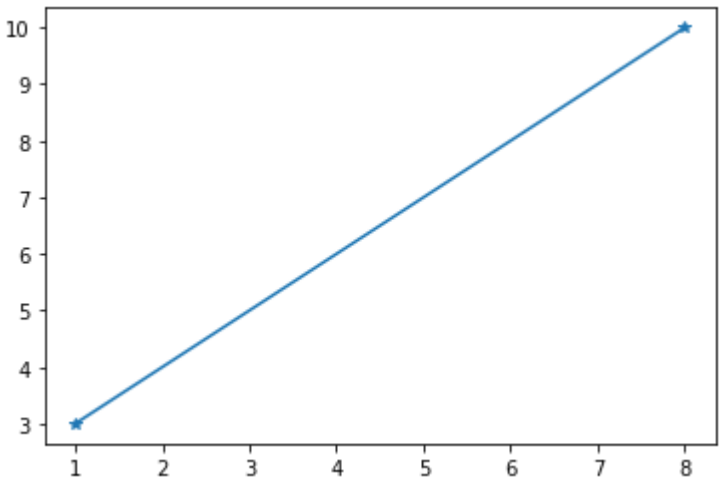
In [ ]:

```
#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()
```



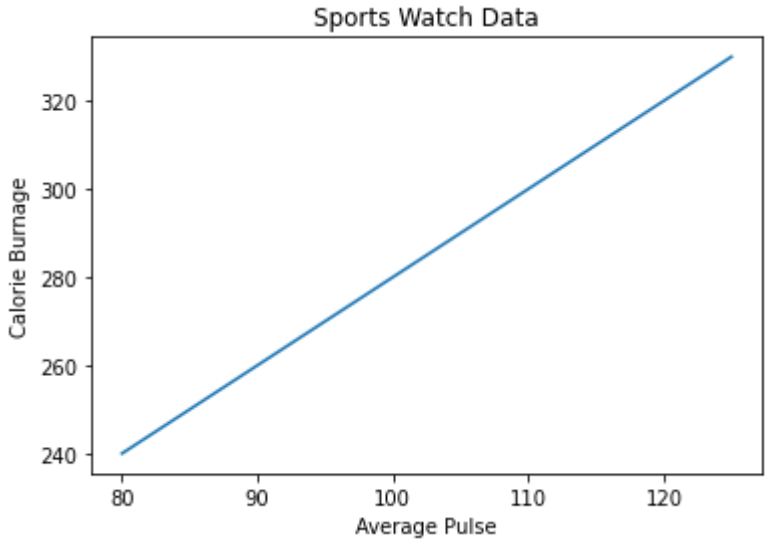
In [ ]:

```
#Draw a line in a diagram from position (1, 3) to position (8, 10): with marker
import matplotlib.pyplot as plt
import numpy as np
xpoints = np.array([1, 8])
ypoints = np.array([3, 10])
plt.plot(xpoints,ypoints,marker='*')
plt.show()
```



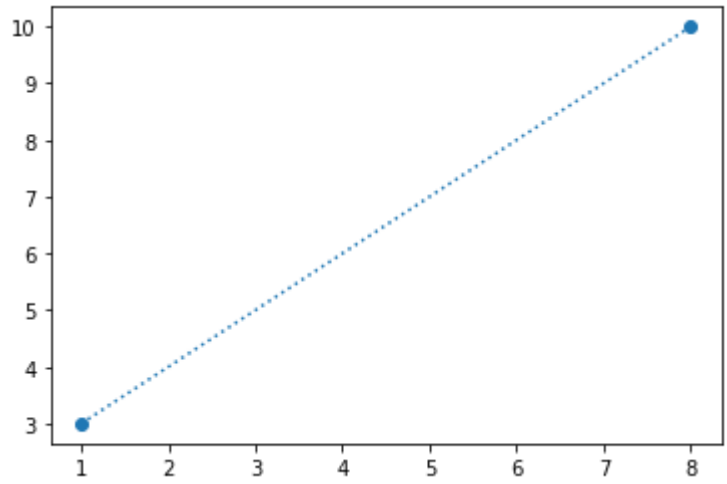
In [ ]:

```
#title-xlabel and ylabel
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.show()
```



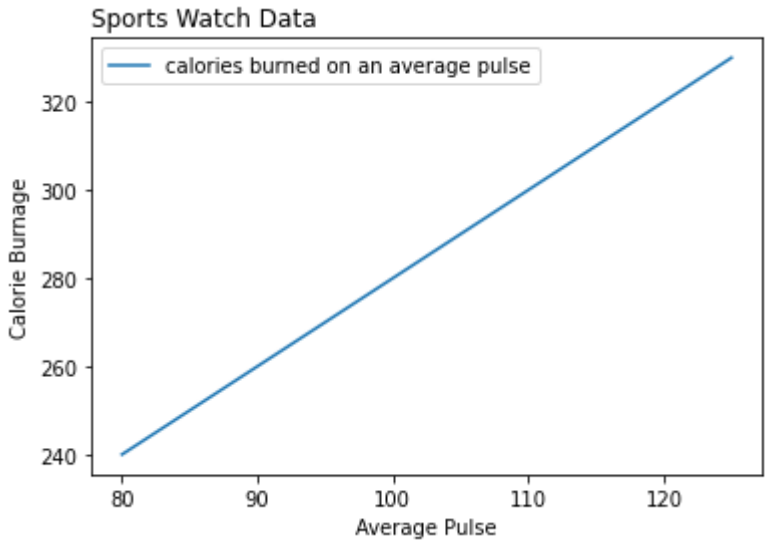
In [ ]:

```
#dotted linestyle
import matplotlib.pyplot as plt
import numpy as np
xpoints = np.array([1, 8])
ypoints = np.array([3, 10])
plt.plot(xpoints,ypoints,marker='o',linestyle="dotted")
plt.show()
```



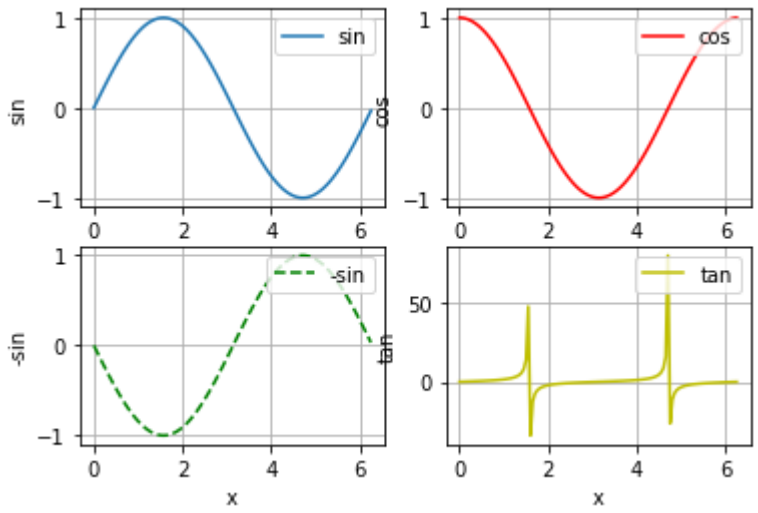
In [ ]:

```
#change location of title and legend
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,label="calories burned on an average pulse")
plt.legend()
plt.show()
```



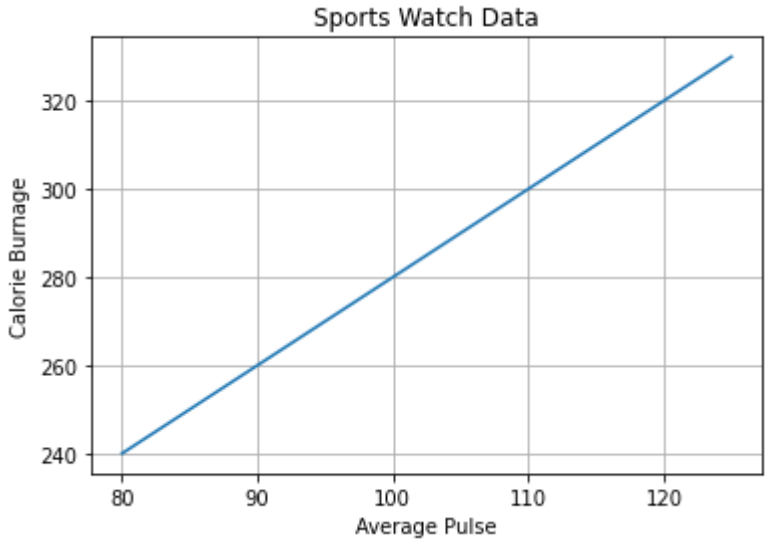
In [ ]:

```
#subplot with  4 figures
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(0, math.pi*2, 0.05)
subplot(2,2,1)
plot(x, sin(x),label='sin')
xlabel('x')
ylabel('sin')
legend(loc='upper right')
grid(True)
subplot(2,2,2)
plot(x, cos(x), 'r-',label='cos')
xlabel('x')
ylabel('cos')
legend(loc='upper right')
grid(True)
subplot(2,2,3)
xlabel('x')
ylabel('-sin')
plot(x, -sin(x), 'g--',label='-sin')
legend(loc='upper right')
grid(True)
subplot(2,2,4)
xlabel('x')
ylabel('tan')
plot(x, tan(x), 'y-',label='tan')
legend(loc='upper right')
grid(True)
show()
```



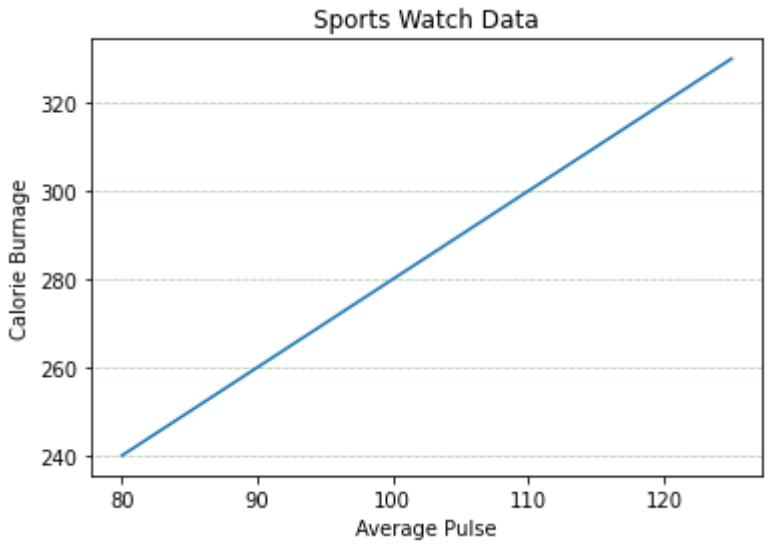
In [ ]:

```
#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 [ ]:

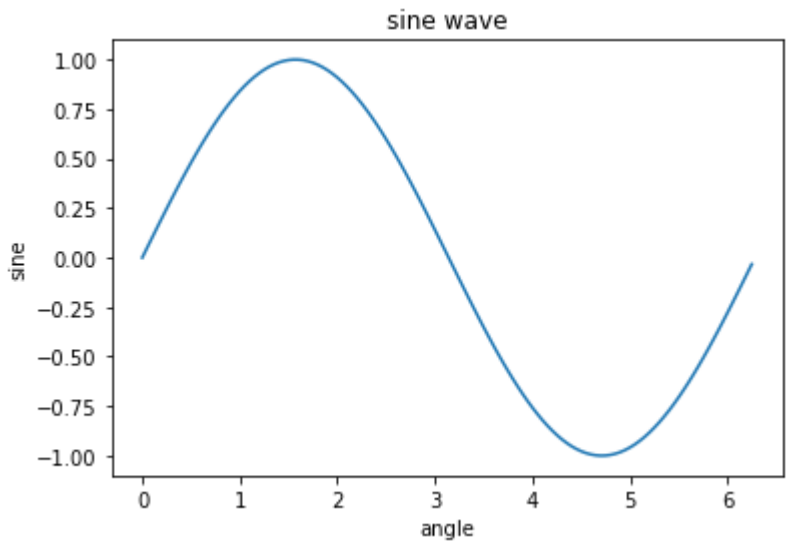
```
#Display only grid lines for the y-axis:

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(color = 'green', linestyle = 'dotted', linewidth = 0.5,axis = 'y')
plt.show()
```



In [ ]:

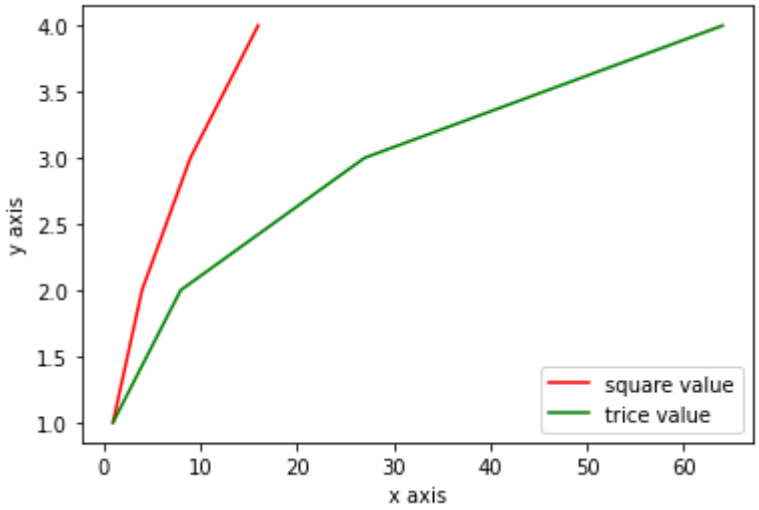
```
import  matplotlib.pyplot as plt
import numpy as np
import math #needed for definition of pi
x = np.arange(0, math.pi*2, 0.05)
print(x)
y = np.sin(x)
plt.plot(x,y)
plt.xlabel("angle")
plt.ylabel("sine")
plt.title('sine wave')
plt.show()
```

```
[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]
```



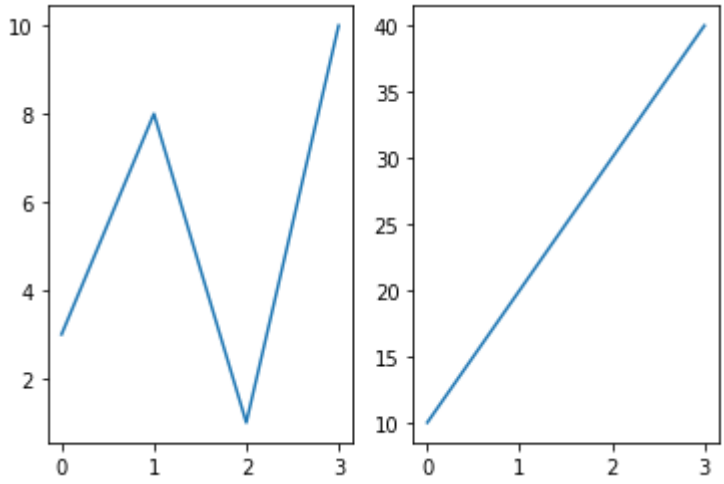
In [ ]:

```
#Legend
import  matplotlib.pyplot as plt
import numpy as np
t=np.array([1,2,3,4])
plt.plot(t**2,t,color="red")
plt.plot(t**3,t,color="green")
plt.xlabel("x axis")
plt.ylabel("y axis")
plt.legend(["square value","trice value"])
plt.show()
```



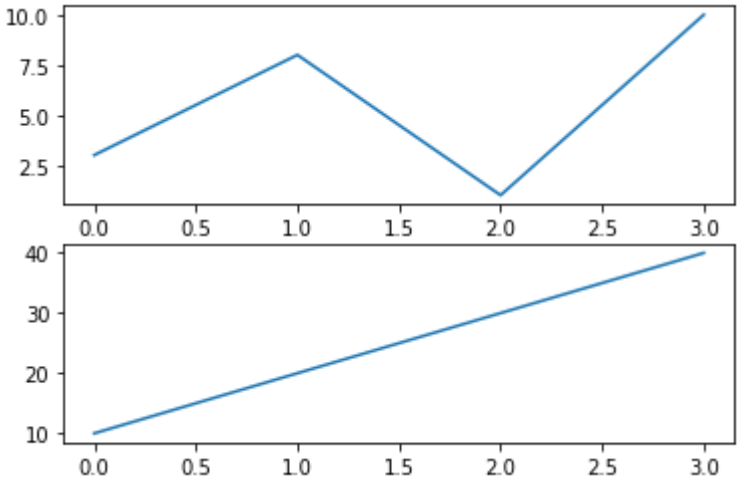
In [ ]:

```
#subplot function
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)
#the figure has 1 row, 2 columns, and this plot is the first plot.
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)
#the figure has 1 row, 2 columns, and this plot is the second plot.
plt.plot(x,y)
plt.show()
```



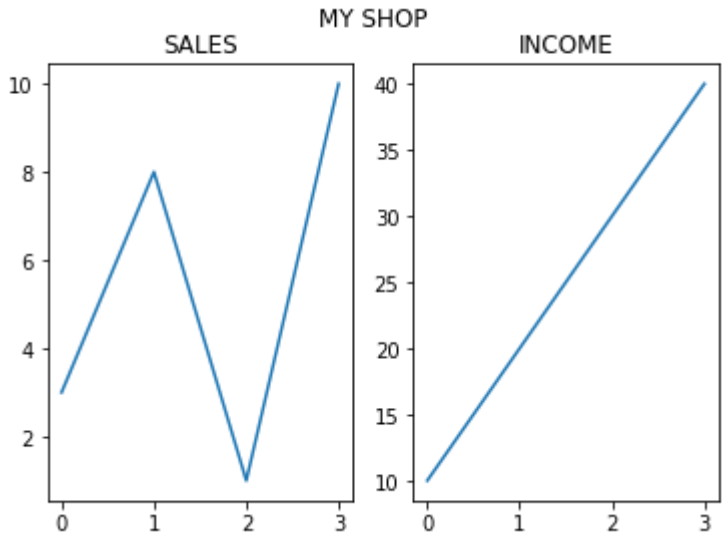
In [ ]:

```
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()
```



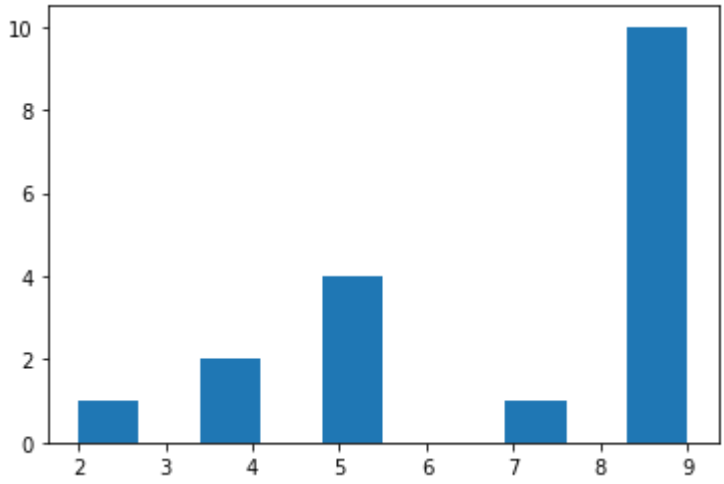
In [ ]:

```
#subtitle and supertitle
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)
plt.title("SALES")
#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.title("INCOME")
plt.suptitle("MY SHOP")
plt.show()
```



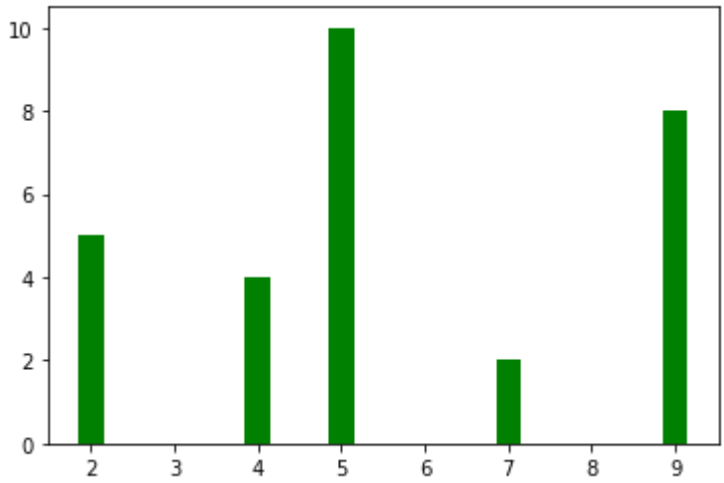
In [ ]:

```
#Creating a histogram
import  matplotlib.pyplot as plt
# x-axis values
x = [5, 2, 9, 4, 7,5,5,5,4,9,9,9,9,9,9,9,9,9]
# Function to plot the histogram
plt.hist(x)
# function to show the plot
plt.show()
```



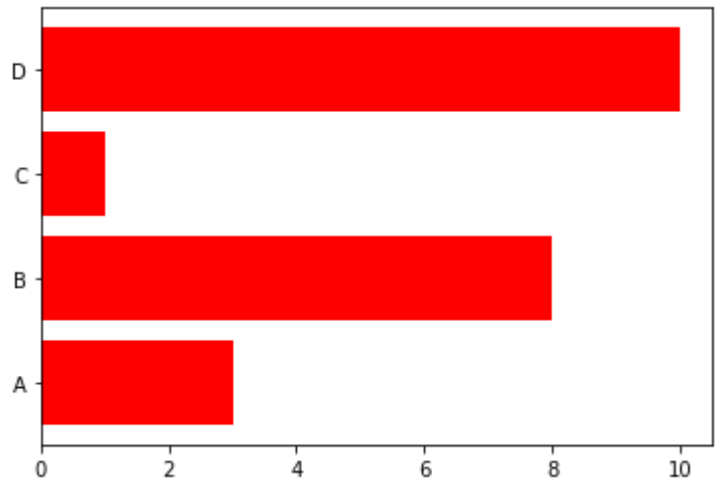
In [ ]:

```
#bar graph with different width
import  matplotlib.pyplot as plt
x = [5, 2, 9, 4, 7]
y = [10, 5, 8, 4, 2]
# Function to plot the bar
plt.bar(x,y,width=0.3,color="green")
# function to show the plot
plt.show()
```



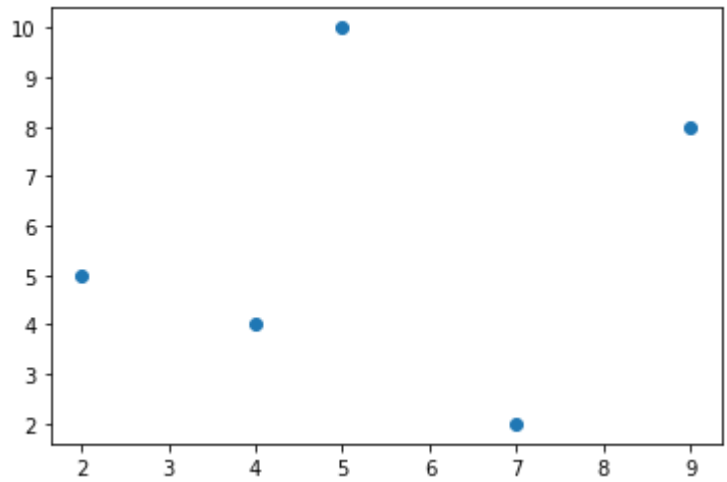
In [ ]:

```
#bar graph along the vertical axis
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, color="red")
plt.show()
```



In [ ]:

```
#Scatter Plot
from matplotlib import pyplot as plt
x = [5, 2, 9, 4, 7]
y = [10, 5, 8, 4, 2]
# Function to plot scatter
plt.scatter(x, y)
plt.show()
```

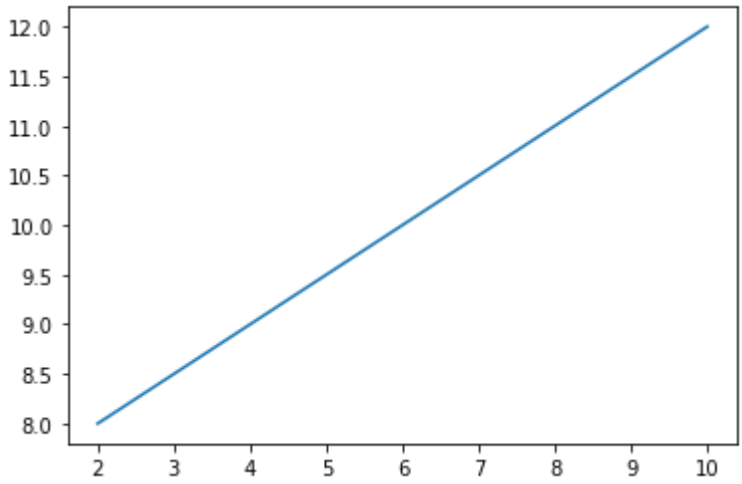


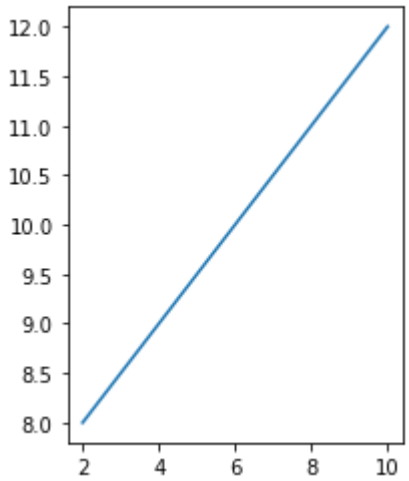
In [ ]:

```
# figure
import matplotlib.pyplot as plt
a = [2, 4, 6, 8, 10]
b = [8, 9, 10, 11, 12]
# Defailt figure size will be shown here
display(plt.plot(a, b))
# Altering the figure size to 3 x 4
plt.figure(figsize = (3, 4))
display(plt.plot(a, b))
```

```
[<matplotlib.lines.Line2D at 0x7f5f8c30d6d0>]
```

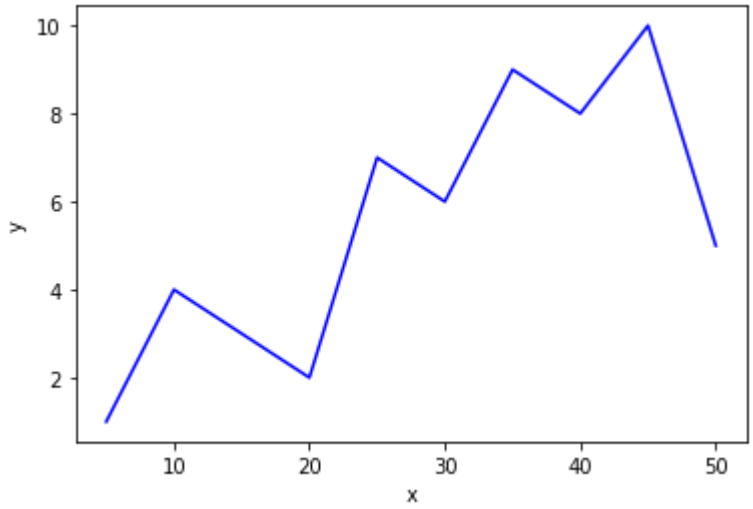
```
[<matplotlib.lines.Line2D at 0x7f5f8c2cd9d0>]
```

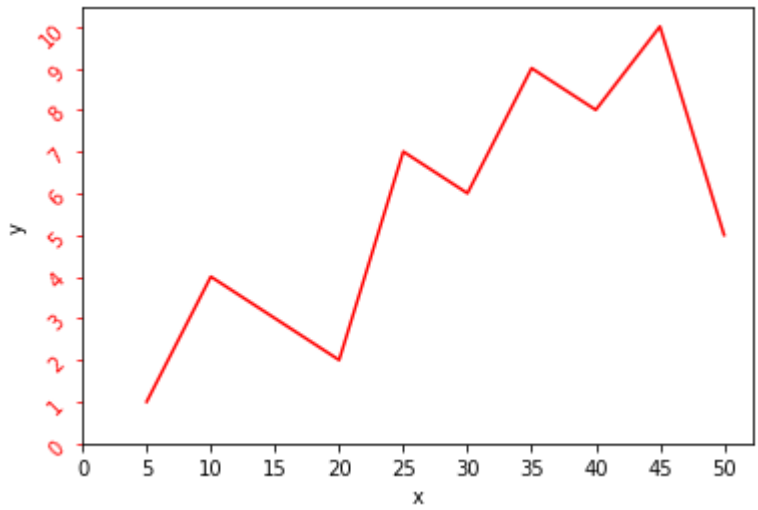




In [ ]:

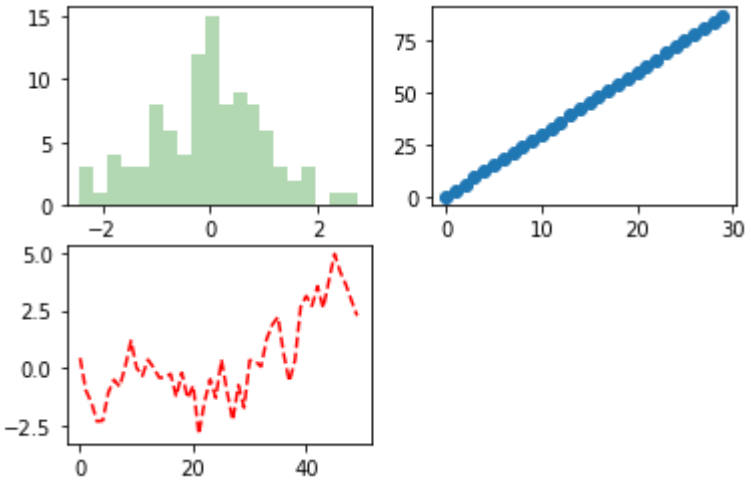
```
#xticks and Yticks
import matplotlib.pyplot as plt
import numpy as np
 # values of x and y axes
x = [5, 10, 15, 20, 25, 30, 35, 40, 45, 50]
y = [1, 4, 3, 2, 7, 6, 9, 8, 10, 5]
plt.figure(1)
plt.plot(x, y, 'b')
plt.xlabel('x')
plt.ylabel('y')
plt.figure(2)
plt.plot(x, y, 'r')
plt.xlabel('x')
plt.ylabel('y')
plt.xticks(np.arange(0, 51, 5))
plt.yticks(np.arange(0, 11, 1))
plt.tick_params(axis='y',colors='red',
                rotation=45)
plt.show()
```





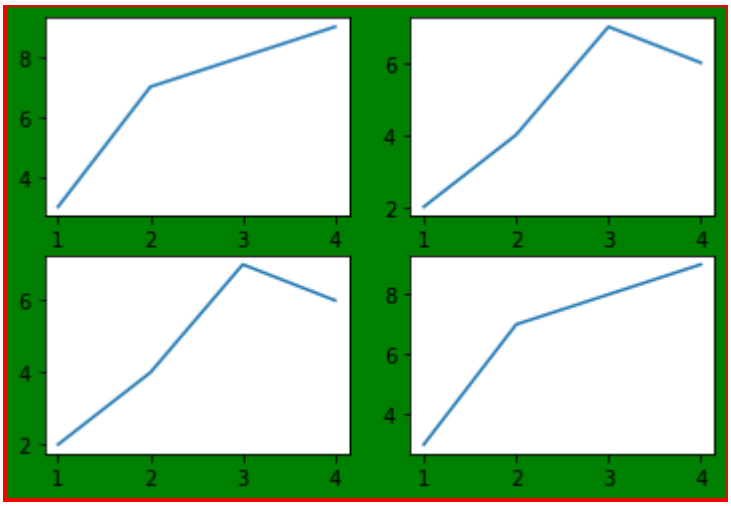
In [ ]:

```
import matplotlib.pyplot as plt
from numpy.random import randn
import numpy as np
fig = plt.figure()
ax1 = fig.add_subplot(2, 2, 1)
ax2 = fig.add_subplot(2, 2, 2)
ax3 = fig.add_subplot(2, 2, 3)
ax3.plot(randn(50).cumsum(), 'r--')
ax1.hist(randn(100), bins=20, color='g', alpha=0.3)
ax2.scatter(np.arange(30), np.arange(30) + 3 * randn(30))
plt.show()
```



In [ ]:

```
# Adding Subplot to the figure
import matplotlib.pyplot as plt
import numpy as np
t = [1,2,3,4]
s1 = [3,7,8,9]
s2 = [2,4,7,6]
fig=plt.figure(facecolor="green",linewidth=4)# open a new figure
fig.set_edgecolor('red')
plt.subplot(2,2,1)
plt.xticks(t)
plt.plot(t, s1)
# Taking another sub plot
plt.subplot(2,2,2)
plt.plot(t, s2)
plt.xticks(t)
# Taking third sub plot
plt.subplot(2,2,3)
plt.plot(t, s2)
plt.xticks(t)
#taking 4th subplot
plt.subplot(2,2,4)
plt.xticks(t)
plt.plot(t, s1)
plt.show()
```



In [ ]:

```
#add_patch() adding triangle rectangle to a figure
fig = plt.figure()
ax = fig.add_subplot(1, 1, 1)
rect = plt.Rectangle((0.2, 0.75), 0.4, 0.15, color='b', alpha=.5)
circ = plt.Circle((0.7, 0.2), 0.15, color='b', alpha=0.3)
pgon = plt.Polygon([[0.15, 0.15], [0.35, 0.4], [0.2, 0.6]],
color='g', alpha=0.5)
ax.add_patch(rect)
ax.add_patch(circ)
ax.add_patch(pgon)
```

Out[ ]:

```
<matplotlib.patches.Polygon at 0x7f94bcb9ca10>
```