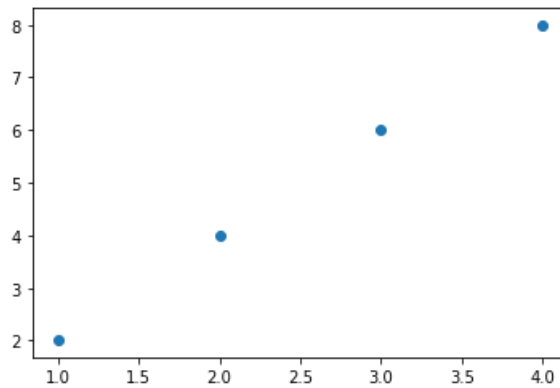


```
In [1]: #This is a supplementary material to the lecture "Plotting Graphs" to quickly revise, whenever needed
```

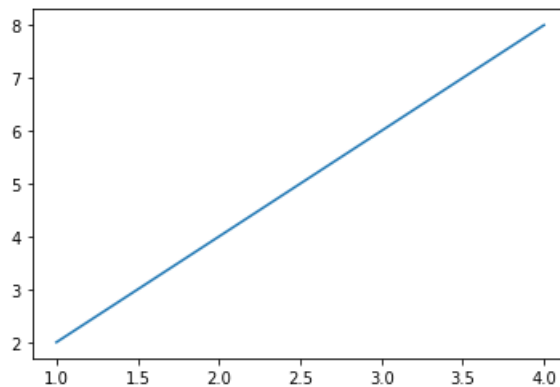
```
In [2]: #We will use a library called Matplotlib for plotting to further analyze our data  
#Matplotlib is a plotting library for the Python
```

```
In [3]: #Let's import the package  
#we will particularly focus on pyplot interface in the matplotlib  
import matplotlib.pyplot as plt
```

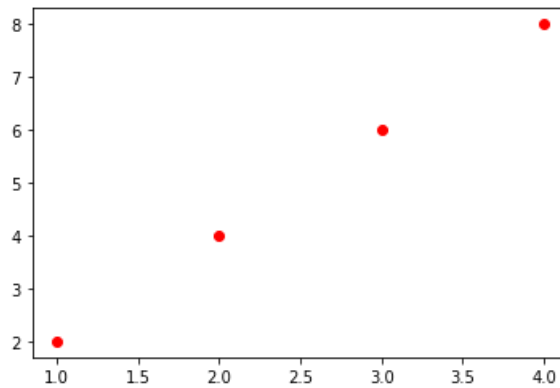
```
In [4]: #simple example  
x = [1, 2, 3, 4]  
y = [2, 4, 6, 8]  
plt.scatter(x, y)      #plotting the points  
plt.show()             #to show the plot
```



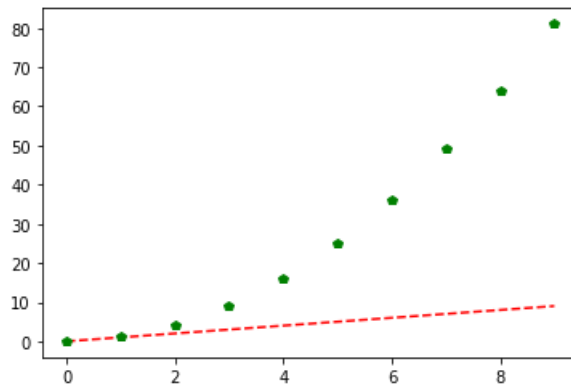
```
In [5]: plt.plot(x,y)  
plt.show()
```



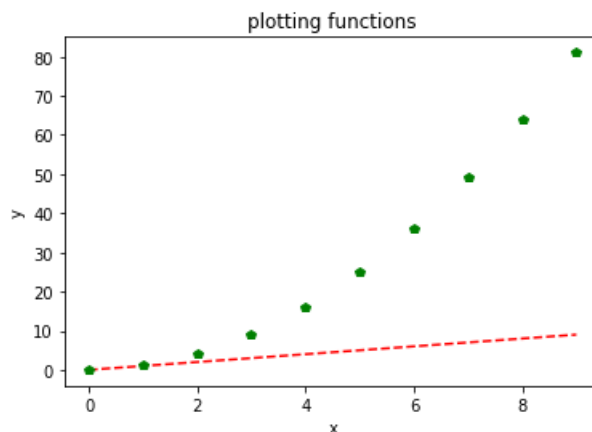
```
In [6]: #to format the style of your plot, there is a third argument, which can be passed and that is format string
plt.plot(x, y, 'ro')    #here, in format string r means red color and o means circles shaped points, try experimenting (--, +, ^, p, s for shapes of points)
plt.show()
```



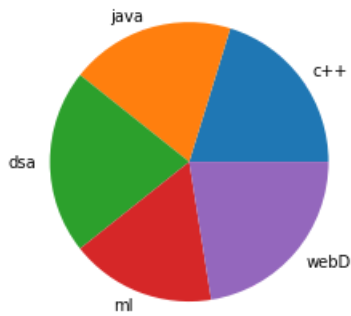
```
In [7]: import numpy as np
#say we want to plot a function like y = x^2
x = np.arange(0, 10, 1)
y1 = x
y2 = x ** 2
plt.plot(x, y1, 'r--', x, y2, 'gp')
plt.show()
```



```
In [8]: x = np.arange(0, 10, 1)
y1 = x
y2 = x ** 2
plt.plot(x, y1, 'r--', x, y2, 'gp')
plt.ylabel('y')
plt.xlabel('x')
plt.title('plotting functions')
plt.show()
```

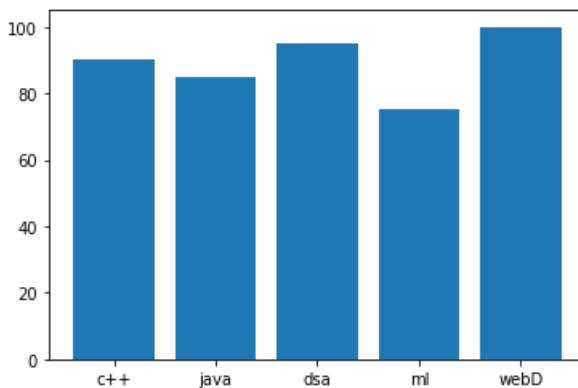


```
In [9]: #pie graph
marks = [90, 85, 95, 75, 100]
subjects = ['c++', 'java', 'dsa', 'ml', 'webD']
plt.pie(marks, labels=subjects)
plt.show()
```



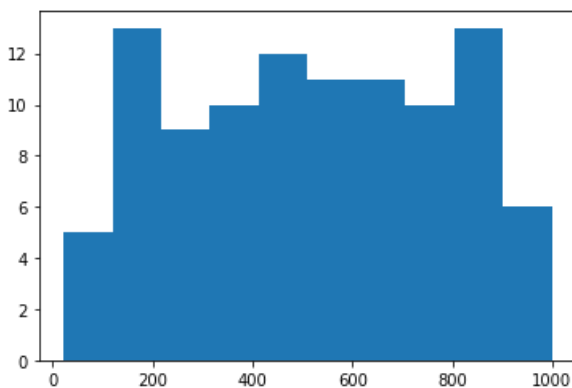
```
In [10]: plt.bar(subjects,marks)
```

Out[10]: <BarContainer object of 5 artists>



```
In [11]: #histogram
#histograms are useful to visualize how the data is distributed in some buckets of particular range
from random import sample
data = sample(range(1, 1000), 100) #here, we are sampling 100 values between 1 and 1000
#let's say we want to see how many values are sampled between 0-100 or 100-200 (buckets) and so on
#we can use histogram for this
plt.hist(data) #by default, it will plot the data distribution in 10 buckets and on y axis, we
have the number of values in a particular bucket
```

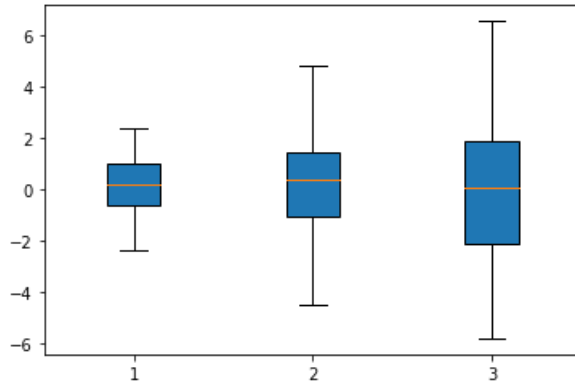
Out[11]: (array([5., 13., 9., 10., 12., 11., 11., 10., 13., 6.]),
array([22., 119.7, 217.4, 315.1, 412.8, 510.5, 608.2, 705.9, 803.6,
901.3, 999.]),
<a list of 10 Patch objects>)



In [12]: *#from above, we can clearly say that there are approx 9 values sampled in the bucket (0-100) and so on*

```
In [13]: #boxplot
data = [np.random.normal(0, std, 100) for std in range(1, 4)]

# rectangular box plot
plt.boxplot(data,vert=True,patch_artist=True);
```



In [14]: *#Thanks, happy Coding!*

In []: *#To download .ipynb notebook, right click the following link and click save as*
<https://ninjasfiles.s3.amazonaws.com/0000000000003222.ipynb>