notes created by swapan chetri

Libraries in python

python library is a collection of functions and methods that allows us to perform many actions without writing our code.

pandas -> data manipulation

numpy -> numerical computing

matplotlib -> visualization

Matplotlib library

-> matplotlib is a python library used for data visualization

-> we can create Bar plots,Scatterplots,Histogram and a lot more with matplotlib

matplotlib

importing matplotlib and other libraries

In [3]:

```
#if matplotlib is not installed try this in the cell(!pip install matplotlib)
from matplotlib import pyplot as plt
# or we can write
# import matplotlib.pyplot as plt
# importing another required libraries
import numpy as np
import pandas as pd
```

lineplot

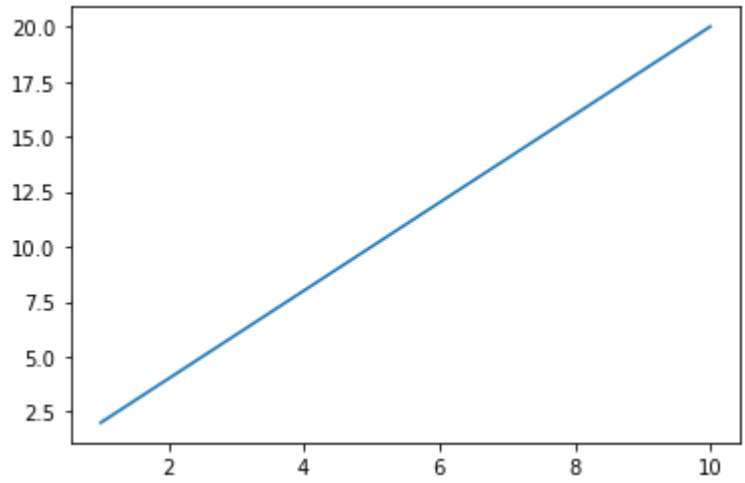
In [6]:

```
x=np.arange(1,11)
print("x is ",x)
y=2*x
print("y is ",y)
```

```
x is  [ 1  2  3  4  5  6  7  8  9 10]
y is  [ 2  4  6  8 10 12 14 16 18 20]
```

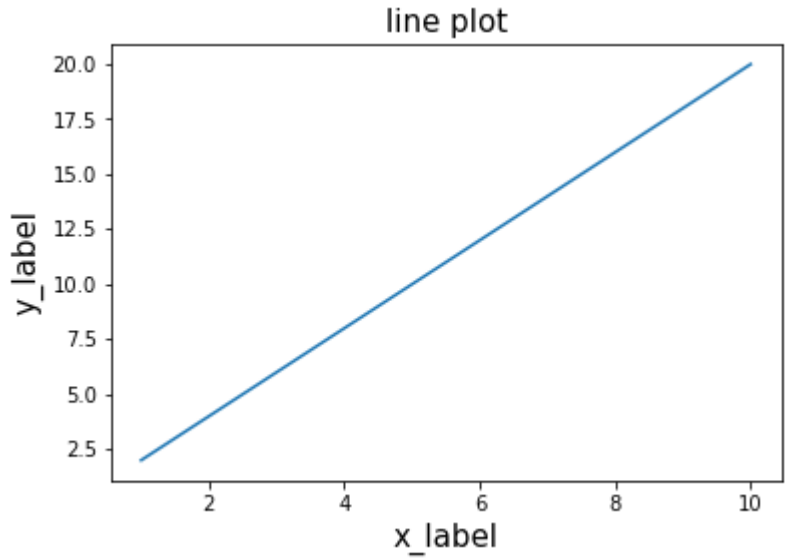
In [7]:

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



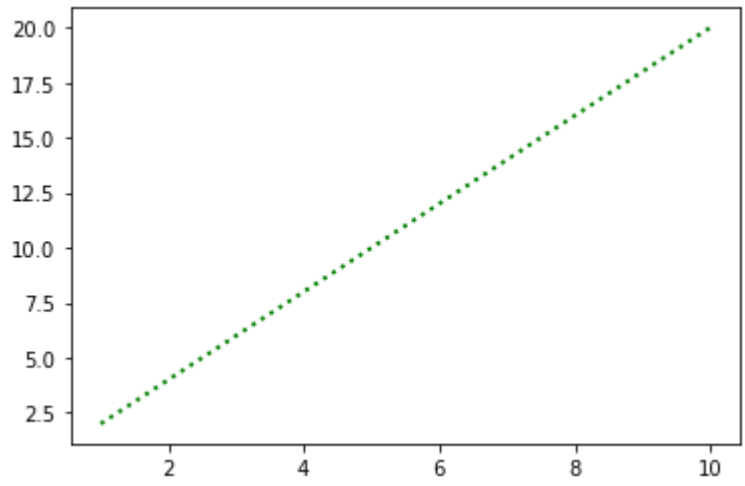
In [11]:

```
#adding labels into it
plt.plot(x,y)
plt.title("line plot",size=15)
plt.xlabel("x_label",size=15)
plt.ylabel("y_label",size=15)
plt.show()
```



In [12]:

```
# changing line asthetics
plt.plot(x,y,color="green",linestyle=":",linewidth=2)
plt.show()
```



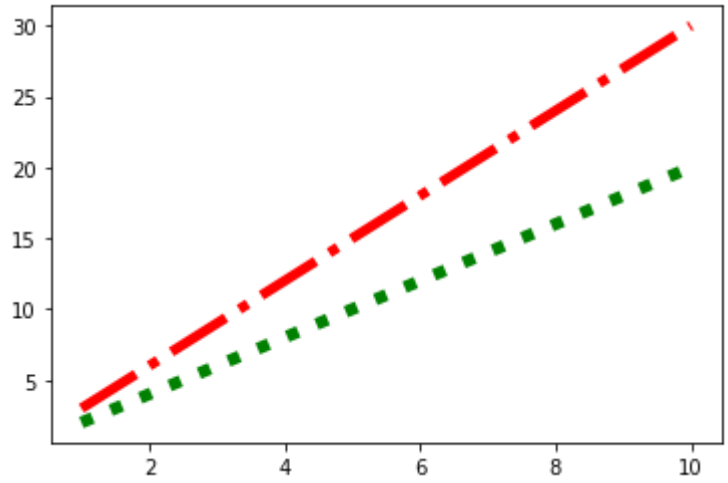
In [13]:

```
# adding two lines in the same plot
x=np.arange(1,11)
print("x is ",x)
y=2*x
print("y is ",y)
z=3*x
print("z is ",z)
```

```
x is  [ 1  2  3  4  5  6  7  8  9 10]
y is  [ 2  4  6  8 10 12 14 16 18 20]
z is  [ 3  6  9 12 15 18 21 24 27 30]
```

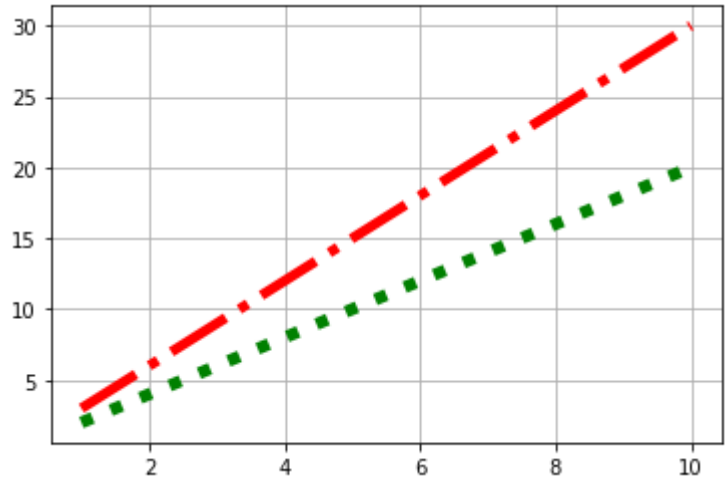
In [15]:

```
# adding two lines in the same plot
plt.plot(x,y,color="green",linestyle=":",linewidth=6)
plt.plot(x,z,color="red",linestyle="-.",linewidth=5)
plt.show()
```



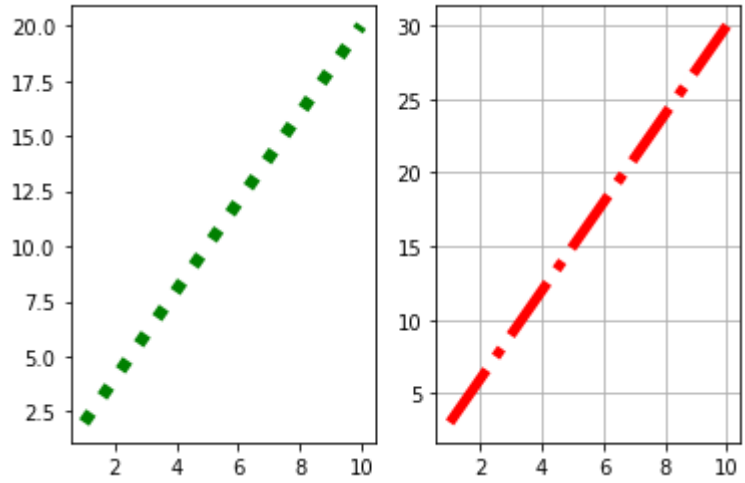
In [16]:

```
# adding grid into the graph
plt.plot(x,y,color="green",linestyle=":",linewidth=6)
plt.plot(x,z,color="red",linestyle="-.",linewidth=5)
plt.grid(True)
plt.show()
```



In [17]:

```
# adding subplots
# the arguments are like "subplot(row,column,index_no)"
plt.subplot(1,2,1)
plt.plot(x,y,color="green",linestyle=":",linewidth=6)
plt.subplot(1,2,2)
plt.plot(x,z,color="red",linestyle="-.",linewidth=5)
plt.grid(True)
plt.show()
```



Barplot

In [23]:

```
# creating a dictionary
students={"Harsh Rajesh Tiwari":90,"Dungala Gangaraju":91,"Ch. Jhansi Priya":95}
names=list(students.keys())
marks=list(students.values())
print(names)
marks
```

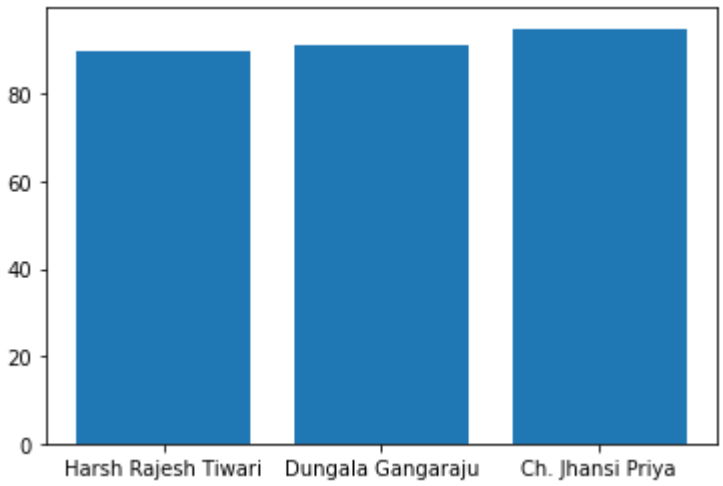
```
['Harsh Rajesh Tiwari', 'Dungala Gangaraju', 'Ch. Jhansi Priya']
```

Out[23]:

```
[90, 91, 95]
```

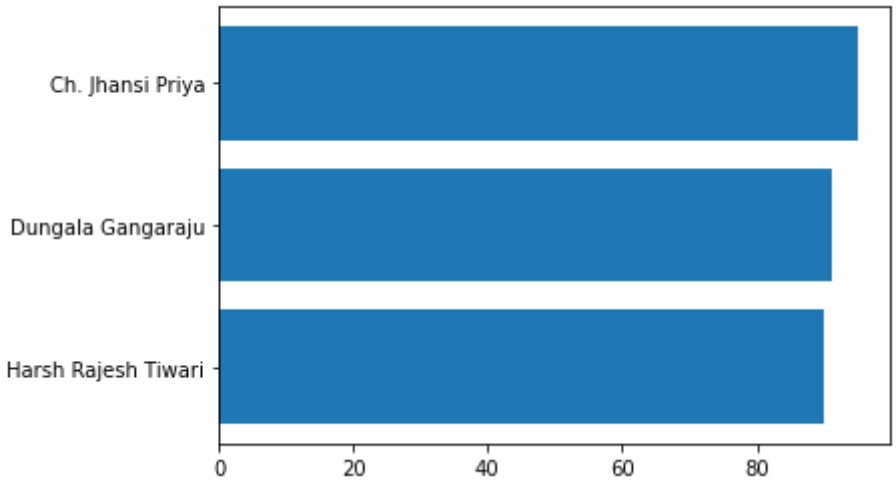
In [28]:

```
# barplot (x=categorical,y=numerical)
plt.bar(names,marks)
plt.show()
```



In [29]:

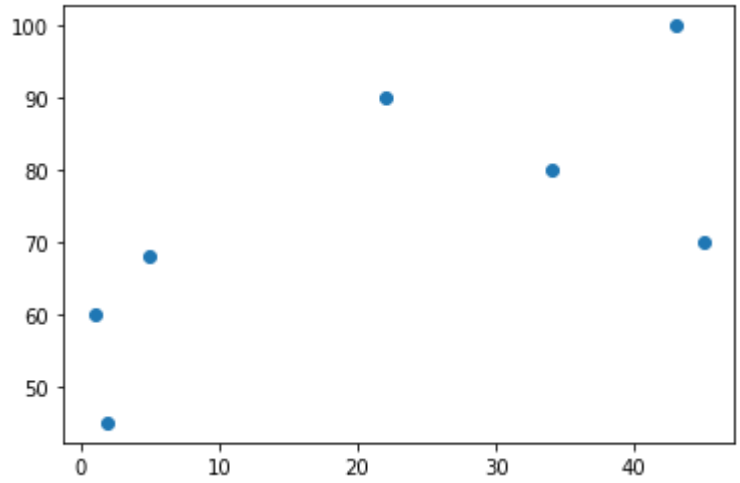
```
#horizontal barplot
plt.barh(names,marks)
plt.show()
```



scatterplot

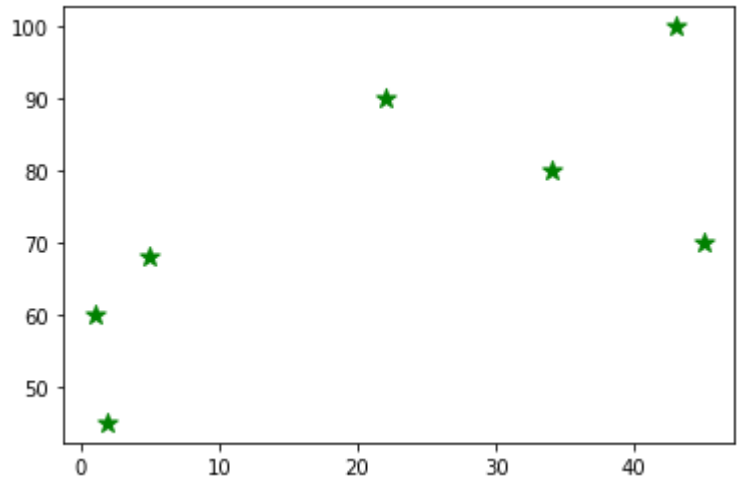
In [31]:

```
x=[1,2,5,45,34,22,43]
y=[60,45,68,70,80,90,100]
plt.scatter(x,y)
plt.show()
```



In [33]:

```
#changing mark asthetics
# here c="g" means color="green" and s=100 means size =100
plt.scatter(x,y,marker="*",c="g",s=100)
plt.show()
```



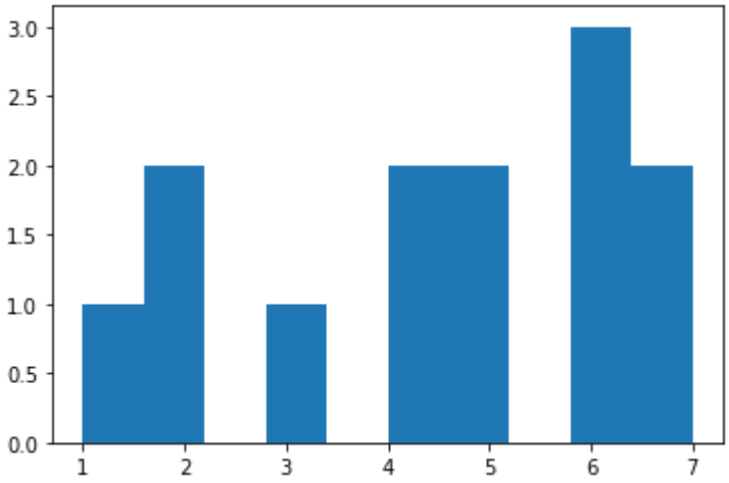
In [34]:

```
# we can add titles xlabel ylabel subplots same in all the plots in the matplotlib
```

histogram

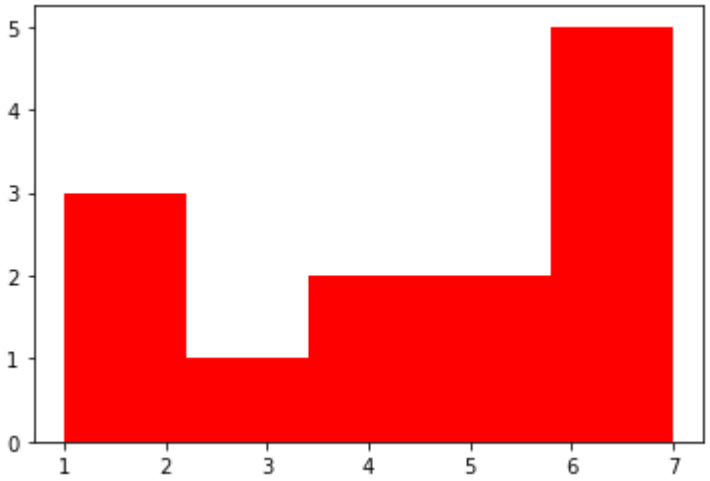
In [38]:

```
# barplot=we can use for categorical values
# histogram=numerical values
data=[1,2,3,2,4,5,6,7,7,6,6,4,5]
plt.hist(data)
plt.show()
```



In [41]:

```
# changing asthetics
data=[1,2,3,2,4,5,6,7,7,6,6,4,5]
# bins=5 means no. of bins will be equal to 5
plt.hist(data,color="red",bins=5)
plt.show()
```



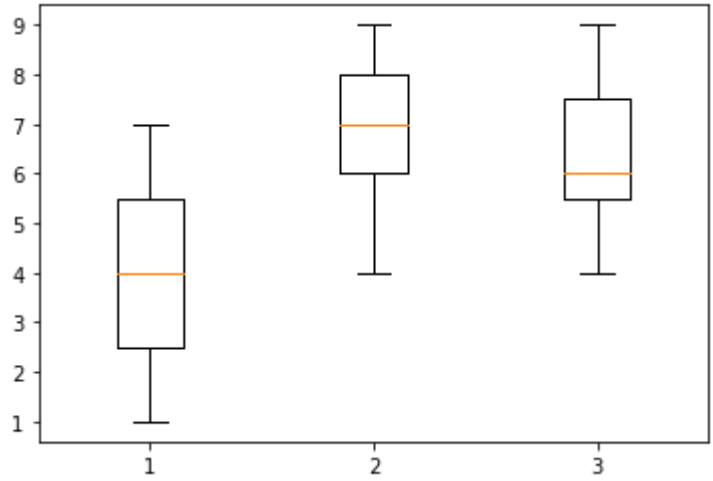
boxplot

In [43]:

```
one=[1,2,3,4,5,6,7]
two=[6,7,8,9,4,6,8]
three=[6,5,7,8,9,6,4]
data=list([one,two,three])
```

In [44]:

```
plt.boxplot(data)
plt.show()
```



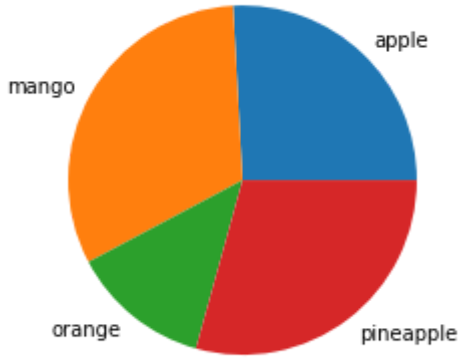
pie chart

In [45]:

```
fruit=["apple","mango","orange","pineapple"]
quantity=[60,74,30,68]
```

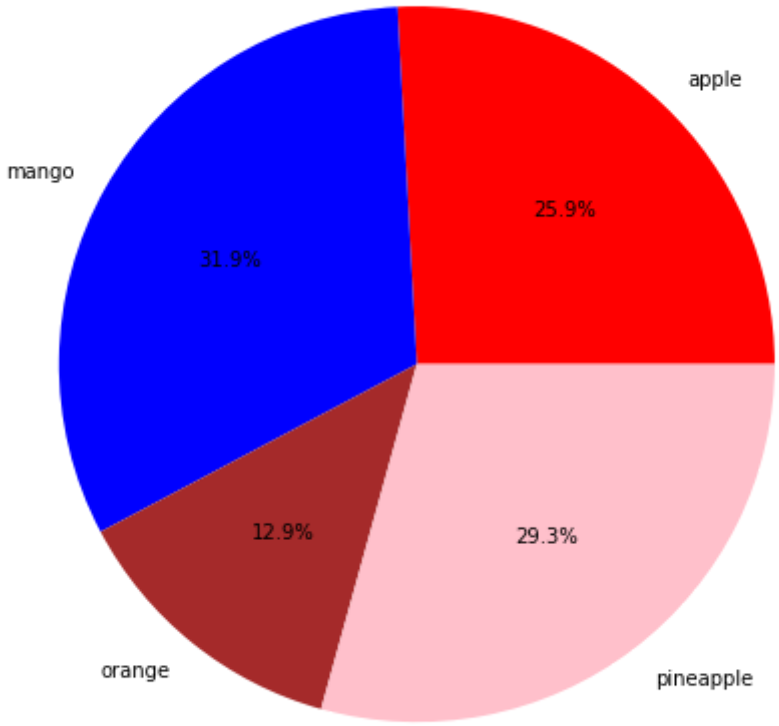
In [47]:

```
plt.pie(quantity,labels=fruit)
plt.show()
```



In [52]:

```
# changing asthetics
# changing the size of graph
plt.figure(figsize=(8,9))
plt.pie(quantity,labels=fruit,autopct='%0.1f%%',colors=["red","blue","brown","pink"])
plt.show()
```



In [ ]:

```

```