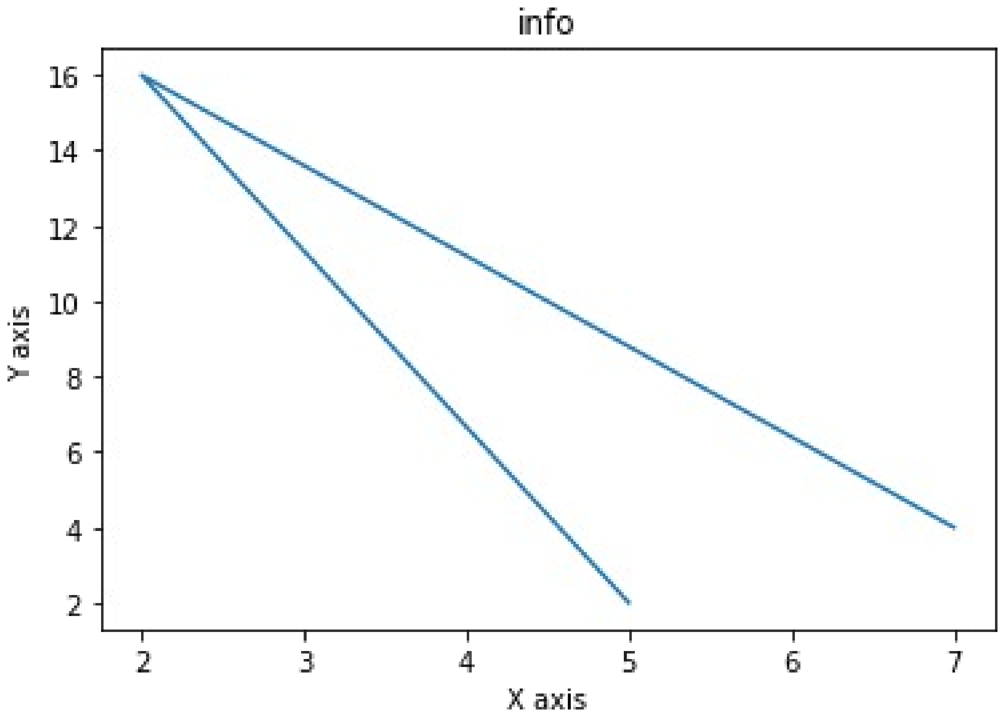
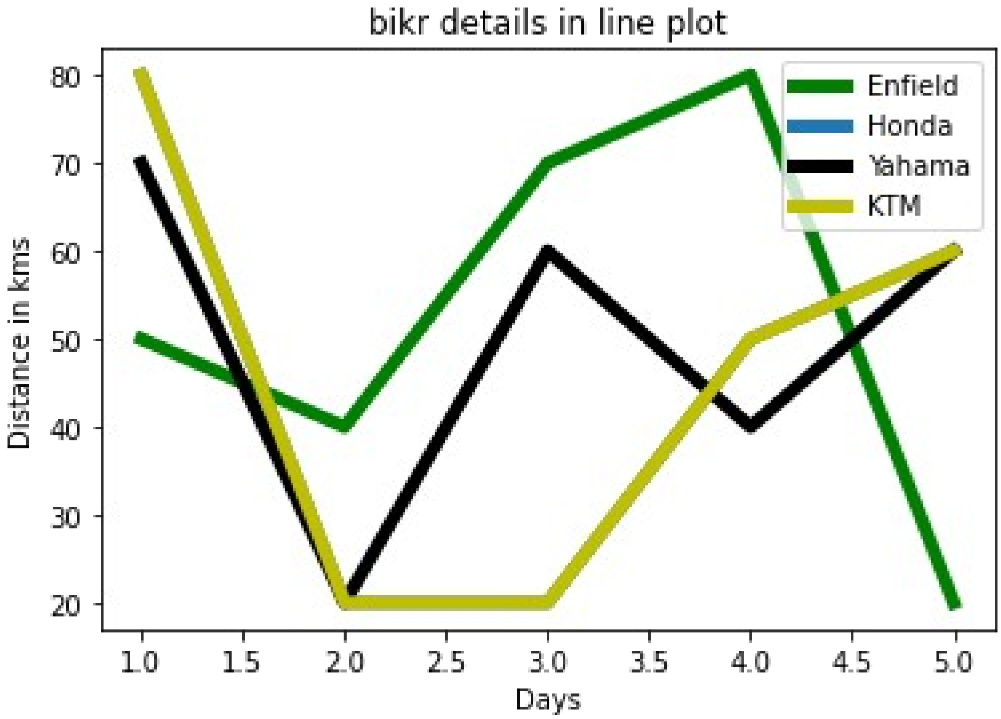
```
import matplotlib.pyplot as plt

## Line plot
x_values = [0,1,2,3,4,5]
squares = [0,1,4,9,16,25]
plt.plot(x_values, squares)
plt.savefig("testimage.jpg")

## plot a line with details
x = [5,2,7]
y = [2,16,4]
plt.plot(x,y)

plt.title("info")
plt.ylabel("Y axis")
plt.xlabel("X axis")

plt.show()
```

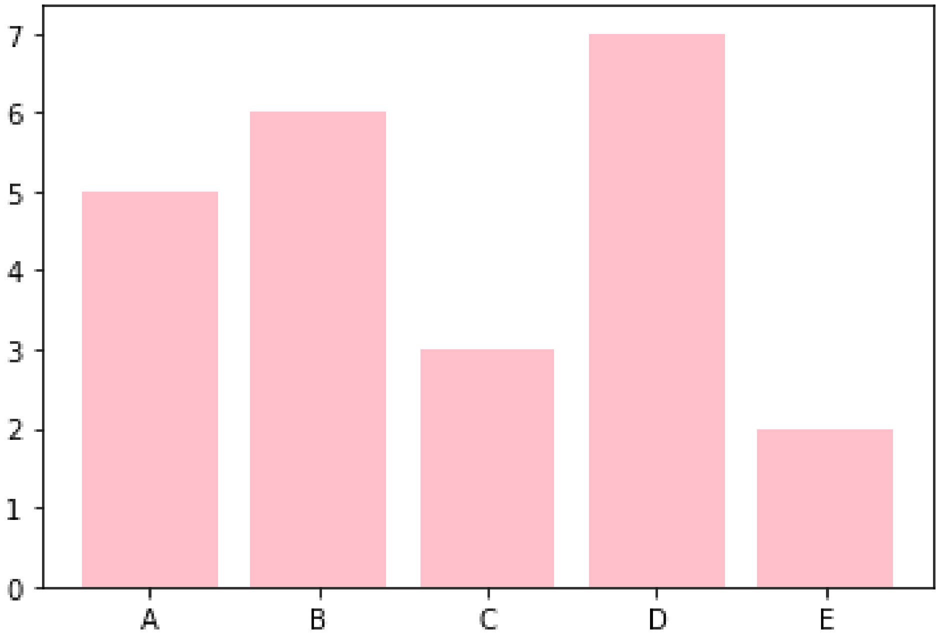


```
# plot multiple lines
x = [1,2,3,4,5]
y = [50,40,70,80,20]
y2 = [80,20,20,50,60]
y3 = [70,20,60,40,60]
y4 = [80,20,20,50,60]

plt.plot(x,y,'g', label = 'Enfield', linewidth=5)
plt.plot(x, y2,label='Honda', linewidth=5)
plt.plot(x,y3,'k',label='Yahama',linewidth=5)
plt.plot(x, y4, 'y',label='KTM', linewidth=5)
plt.title('bikr details in line plot')
plt.ylabel('Distance in kms')
plt.xlabel('Days')
plt.legend()
```

```
<matplotlib.legend.Legend at 0x219eafedbe0>
```

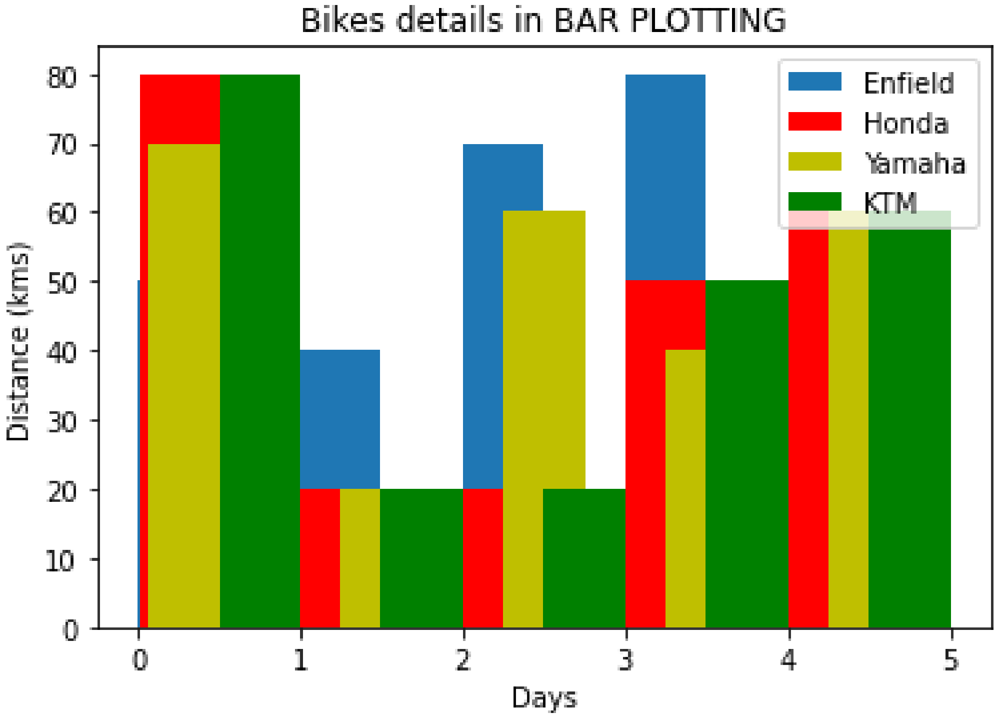


```
## BAR PLOT

## plot a bar chart
x_values = [5,6,3,7,2]
y_values = ["A","B","C","D","E"]

plt.bar(y_values, x_values, color = "pink")
plt.show()
```

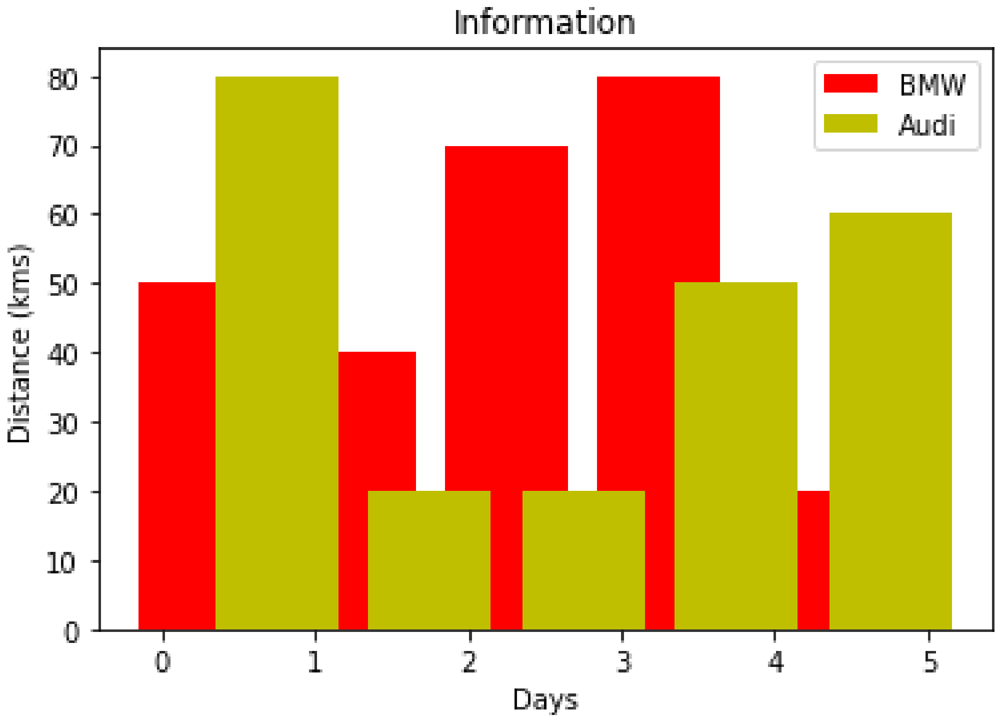


```
## multiple bar plot
## Example-1
plt.bar([0.25,1.25,2.25,3.25,4.25],[50,40,70,80,20],
label="Enfield",width=.5)
plt.bar([0.26,1.25,2.25,3.25,4.25],[80,20,20,50,60],
label="Honda", color='r',width=.5)
plt.bar([0.31,1.5,2.5,3.5,4.5],[70,20,60,40,60],
label="Yamaha", color='y',width=.5)
plt.bar([.75,1.75,2.75,3.75,4.75],[80,20,20,50,60],
label="KTM", color='g',width=.5)
plt.legend()
plt.xlabel('Days')
plt.ylabel('Distance (kms)')
plt.title('Bikes details in BAR PLOTTING')
```

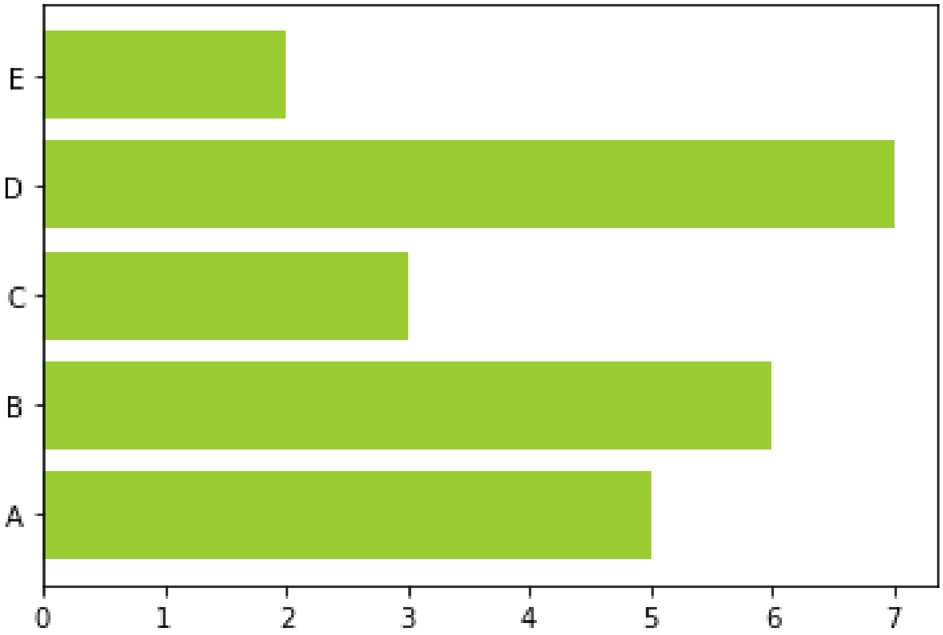
```
Text(0.5, 1.0, 'Bikes details in BAR PLOTTING')
```



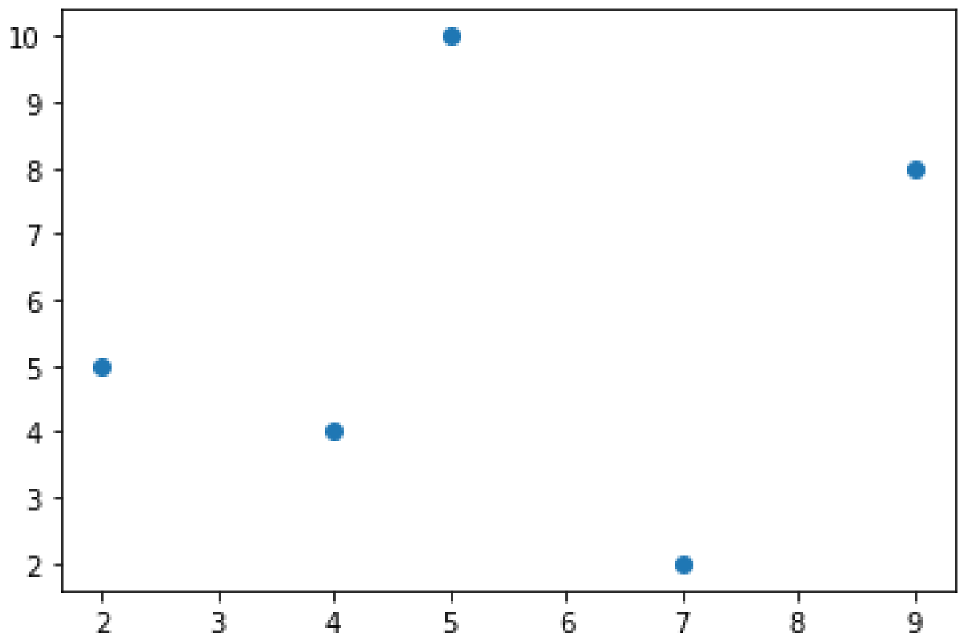
```
#Example 2
#from matplotlib import pyplot as plt
plt.bar([0.25,1.25,2.25,3.25,4.25],[50,40,70,80,20], label="BMW",
color='r')
plt.bar([.75,1.75,2.75,3.75,4.75],[80,20,20,50,60], label="Audi",
color='y')
plt.legend()
plt.xlabel('Days')
plt.ylabel('Distance (kms)')
plt.title('Information')
plt.show()
```



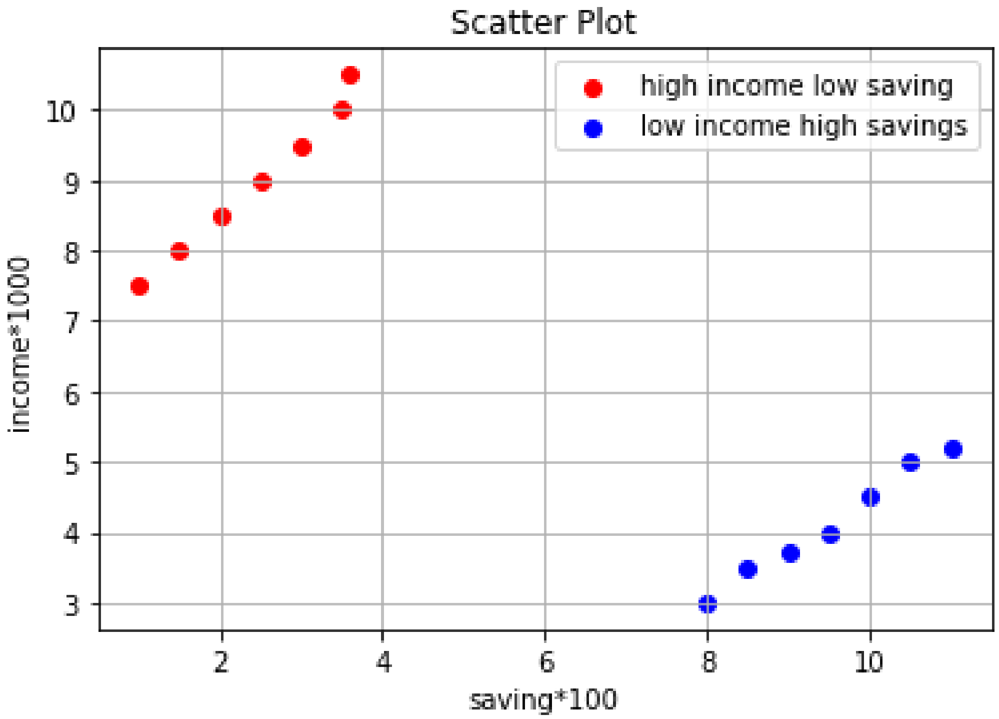
```
## Horizontal Plot
x_values = [5,6,3,7,2]
y_val = ["A", "B", "C", "D", "E"]
plt.barh(y_val,x_values, color ="yellowgreen")
# Adding an "h" after bar will flip the graph
plt.show()
```



```
## Scattered Plot
## Scattered Plot without details
x = [5, 2, 9, 4, 7] # x-axis values
y = [10, 5, 8, 4, 2] # Y-axis values
plt.scatter(x, y) # Function to plot scatter
plt.show() # function to show the plot
```

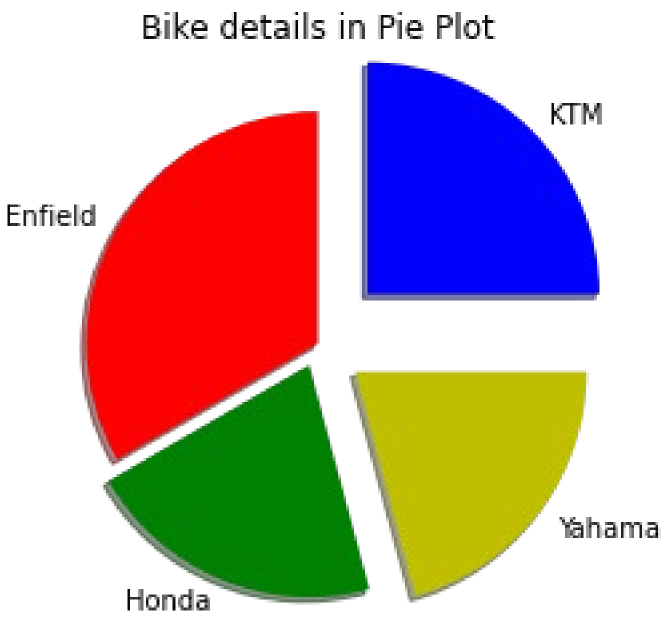


```
# Multiple scattered Plot
x = [1,1.5,2,2.5,3,3.5,3.6]
y = [7.5,8,8.5,9,9.5,10,10.5]
x1=[8,8.5,9,9.5,10,10.5,11]
y1=[3,3.5,3.7,4,4.5,5,5.2]
plt.scatter(x,y, label='high income low saving',color='r')
plt.scatter(x1,y1,label='low income high savings',color='b')
plt.xlabel('saving*100')
plt.ylabel('income*1000')
plt.title('Scatter Plot')
plt.legend()
#plt.show()
plt.grid()
```



```
## 4 pie plot
days = [1,2,3,4,5]
Enfield =[50,40,70,80,20]
Honda = [80,20,20,50,60]
Yahama =[70,20,60,40,60]
KTM = [80,20,20,50,60]
slices = [8,5,5,6]
activities = ['Enfield','Honda','Yahama','KTM']
cols = ['r','g','y','b']
plt.pie(slices, labels=activities, colors=cols, startangle=90, shadow=
True,explode=(0,0.1,0.2,0.3))
plt.title('Bike details in Pie Plot')
```

```
Text(0.5, 1.0, 'Bike details in Pie Plot')
```

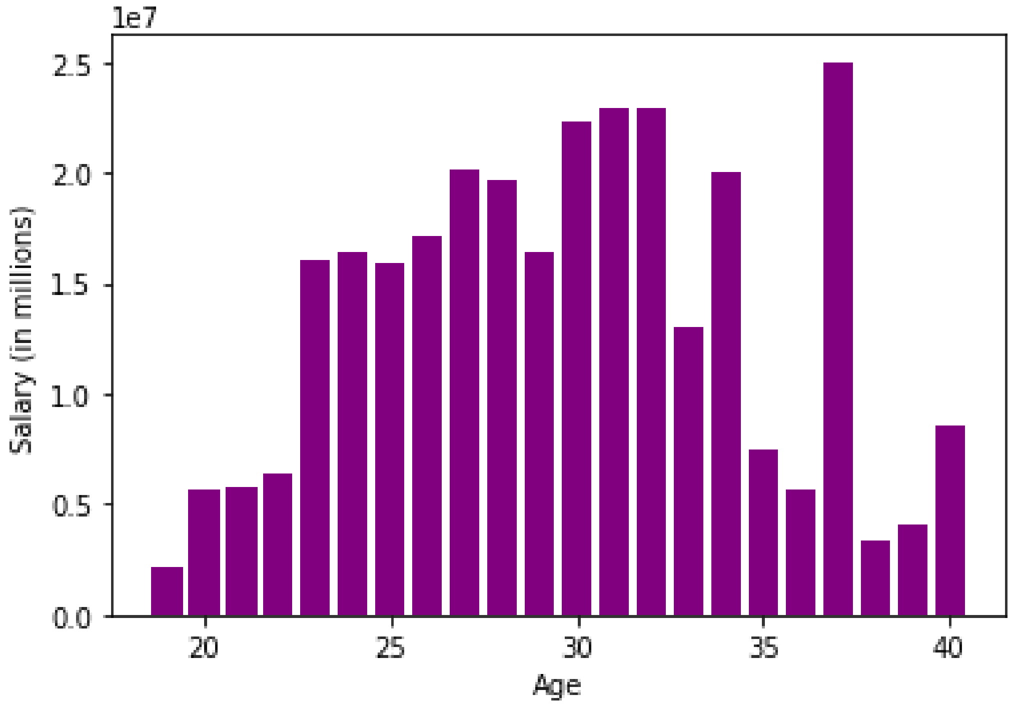


```
#matplotlib plots on NBA data set
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv(r'C:\Users\GLAU\Downloads\nba.csv')
df
```

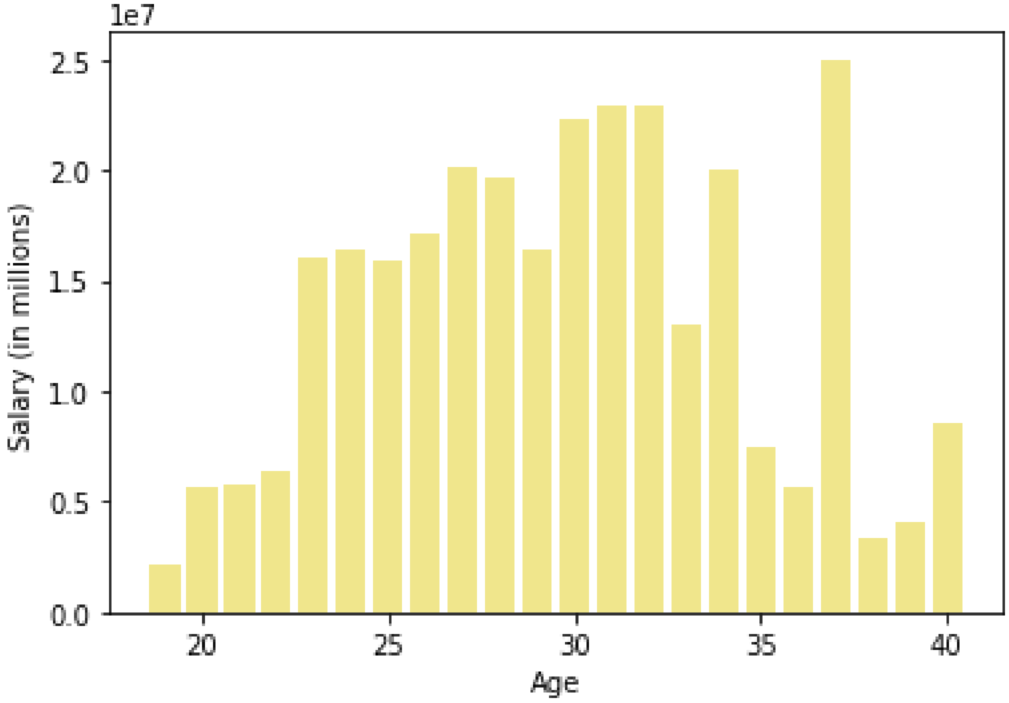
```
              Name            Team  Number Position   Age Height
Weight  \
0    Avery Bradley  Boston Celtics     0.0       PG  25.0    6-2
180.0
1      Jae Crowder  Boston Celtics    99.0       SF  25.0    6-6
235.0
2     John Holland  Boston Celtics    30.0       SG  27.0    6-5
205.0
3      R.J. Hunter  Boston Celtics    28.0       SG  22.0    6-5
185.0
4    Jonas Jerebko  Boston Celtics     8.0       PF  29.0   6-10
231.0
..             ...             ...     ...      ...   ...    ...      .
..
453   Shelvin Mack       Utah Jazz     8.0       PG  26.0    6-3
203.0
454      Raul Neto       Utah Jazz    25.0       PG  24.0    6-1
179.0
455   Tibor Pleiss       Utah Jazz    21.0        C  26.0    7-3
256.0
456    Jeff Withey       Utah Jazz    24.0        C  26.0    7-0
231.0
457            NaN             NaN     NaN      NaN   NaN    NaN
NaN
```

```
               College     Salary
0                Texas  7730337.0
1            Marquette  6796117.0
2    Boston University        NaN
3        Georgia State  1148640.0
4                  NaN  5000000.0
..                 ...        ...
453             Butler  2433333.0
454                NaN   900000.0
455                NaN  2900000.0
456             Kansas   947276.0
457                NaN        NaN

[458 rows x 9 columns]
```

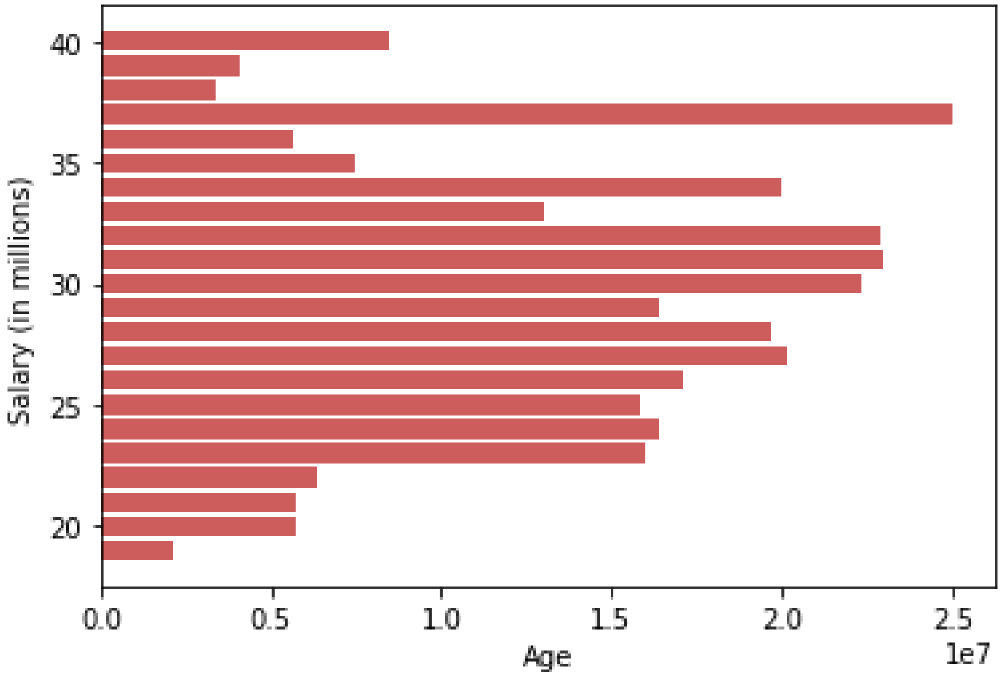
```
## BAR GRAPH
#Example 1
y_values = df['Salary']
x_values = df['Age']
plt.xlabel('Age')
plt.ylabel('Salary (in millions)')
#To plot a bar graph plt.bar() command is used
#This plots a bar graph between Age and Salaries of NBA players
plt.bar(x_values,y_values,color = "purple")
plt.show()
```



```
# Example 2
y_values = df['Salary']
x_values = df['Age']
plt.xlabel('Age')
plt.ylabel('Salary (in millions)')
# Making changes in the color field changes the colour of the graph
plt.bar(x_values,y_values,color = "khaki")
plt.show()
```

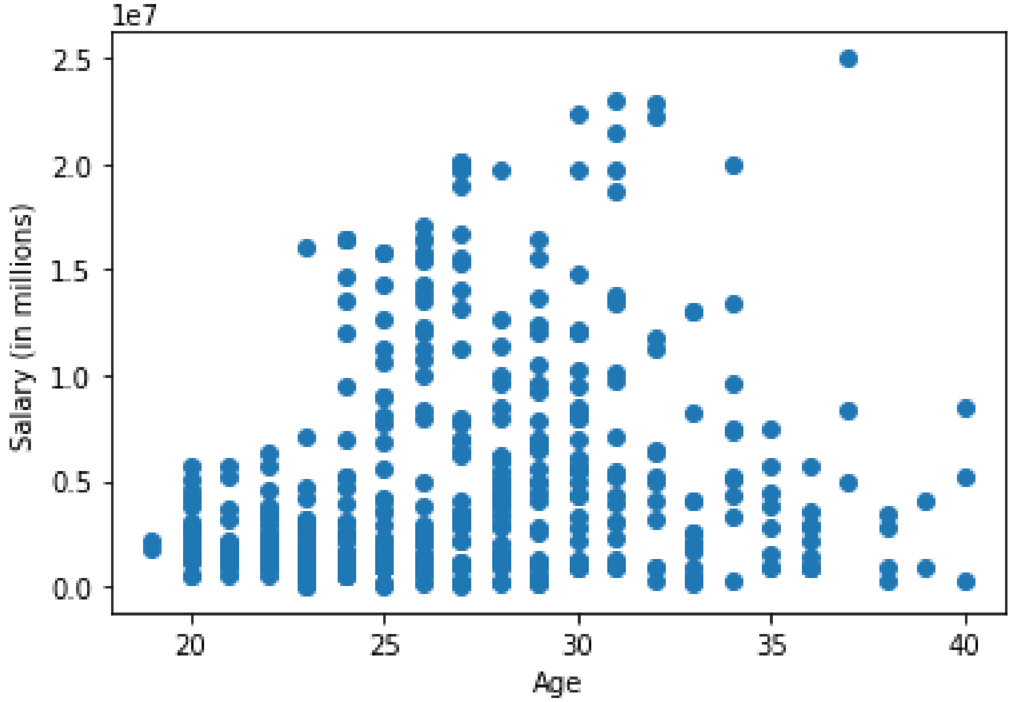


```
## Horizontal bar graph
y_values = df['Salary']
x_values = df['Age']
plt.xlabel('Age')
plt.ylabel('Salary (in millions)')
plt.barh(x_values,y_values,color = "indianred")
plt.show()
```



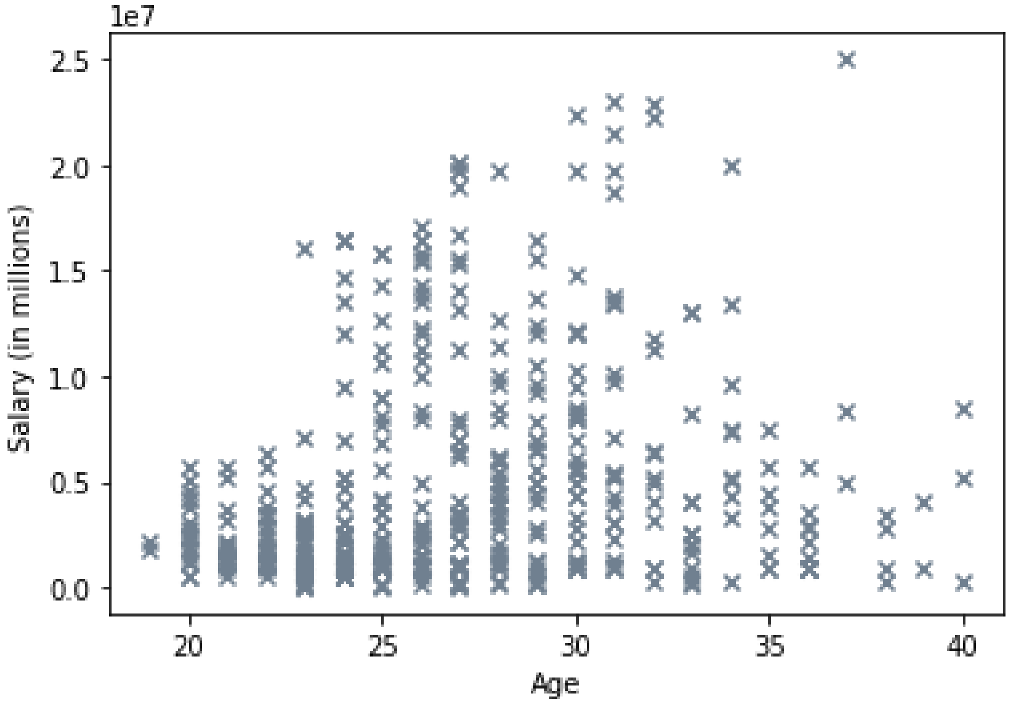
```
## Scattered Plot
## Example-1
y_value=df['Salary']
x_values=df['Age']
#To display a scatter plot we use plt.scatter() command
plt.xlabel('Age')
plt.ylabel('Salary (in millions)')
#This displays a scatter plot between Age and Salaries of NBA players
plt.scatter(x_values,y_values)
```

```
<matplotlib.collections.PathCollection at 0x219ec3a0310>
```



```
#Example 2
y_value=df['Salary']
x_values=df['Age']
plt.xlabel('Age')
plt.ylabel('Salary (in millions)')
#We can also change the type and colors of scatter plot as shown
plt.scatter(x_values,y_values, color = "slategrey", marker = "x")
```

```
<matplotlib.collections.PathCollection at 0x219ec9096d0>
```



```
## LINEAR REGRESSION USING MULTIPLE VARIABLES
import pandas as pd
import numpy as np
from sklearn import linear_model
df = pd.read_csv(r'C:\Users\GLAU\Downloads\homeprices.csv')
df
```

```
   area  bedrooms  age   price
0  2600       3.0   20  550000
1  3000       4.0   15  565000
2  3200       NaN   18  610000
3  3600       3.0   30  595000
4  4000       5.0    8  760000
5  4100       6.0    8  810000
```

```
df.bedrooms.median()
```

```
4.0
```

```
df.bedrooms = df.bedrooms.fillna(df.bedrooms.median())
df
```

```
   area  bedrooms  age   price
0  2600       3.0   20  550000
1  3000       4.0   15  565000
2  3200       4.0   18  610000
3  3600       3.0   30  595000
4  4000       5.0    8  760000
5  4100       6.0    8  810000
```

```
reg = linear_model.LinearRegression()
reg.fit(df.drop('price',axis='columns'),df.price)
```

```
LinearRegression()
```

```
reg.coef_
```

```
array([  112.06244194, 23388.88007794, -3231.71790863])
```

```
reg.intercept_
```

```
221323.00186540408
```

```
## Find price of home with 3000 sqr ft area, 3 bedrooms, 40 year old
reg.predict([[3000, 3, 40]])
```

```
array([498408.25158031])
```

```
112.06244194*3000 + 23388.88007794*3 + -3231.71790863*40 +
221323.00186540384
```

```
498408.25157402386
```

```
## Find price of home with 2500 sqr ft area, 4 bedrooms, 5 year old
reg.predict([[2500, 4, 5]])
```

```
array([578876.03748933])
```