In [1]:
```
# Import libraries,pandas handle tables
import pandas as pd
```

In ...:
```
#DATA COLLECTION
# daaset file create and pd.read_csv are class function,Salary_Data file s
dataset=pd.read_csv("Salary_Data.csv")
```

In [3]:
```
# Salary data set stord in dataset as a table format
dataset
```

Out[3]:

	YearsExperience	Salary
0	1.1	39343
1	1.3	46205
2	1.5	37731
3	2.0	43525
4	2.2	39891
5	2.9	56642
6	3.0	60150
7	3.2	54445
8	3.2	64445
9	3.7	57189
10	3.9	63218
11	4.0	55794
12	4.0	56957
13	4.1	57081
14	4.5	61111
15	4.9	67938
16	5.1	66029
17	5.3	83088
18	5.9	81363
19	6.0	93940
20	6.8	91738
21	7.1	98273
22	7.9	101302
23	8.2	113812
24	8.7	109431
25	9.0	105582
26	9.5	116969
27	9.6	112635
28	10.3	122391
29	10.5	121872

In [4]:

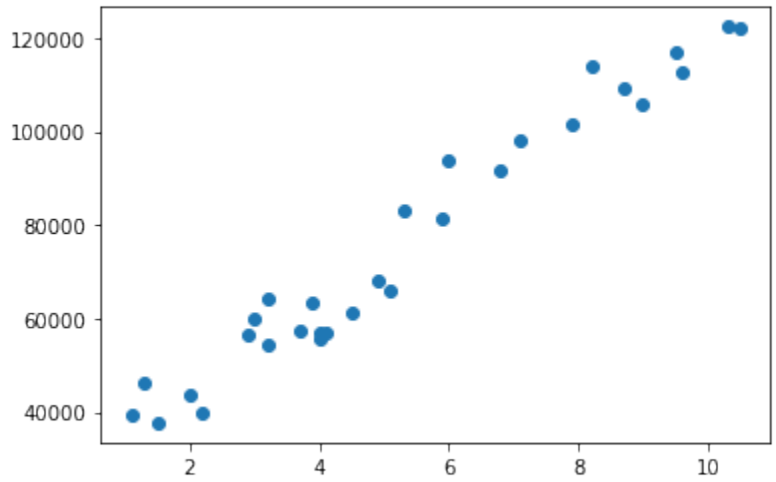
In [5]:
```
# output datas stored at dep variable
dep=dataset[["Salary"]]
```

In [6]:
```
# matplot lib imported  because to plot the graph and named as plt
import matplotlib.pyplot as plt
```

In...
```
# scatter is used to plot the graph,input and output variables are named ar
plt.scatter(indep,dep)
plt.Xlable("experience")
plt.ylable("Salary")
plt.show()
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Input In [7], in <cell line: 3>()
      1 #scatter is used to plot the graph,input and output variables are name
d and labled,to show in the graph
      2 plt.scatter(indep,dep)
----> 3 plt.Xlable("experience")
      4 plt.ylable("Salary")
      5 plt.show()

AttributeError: module 'matplotlib.pyplot' has no attribute 'Xlable'
```



In...
```
#SPLIT TRAIN AND TEST
# train and test function to be loaed in the model selection class at sk le
# 4 vriables silpit up as train and test,test size denotes for train and te
# fixed for integrity
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(indep,dep,test_size=0.30,rar
```

In [9]:
```
X_train.shape
#training set Row column defined
```

Out[9]: (21, 1)

In [10]:
```
len(X_train)
```

Out[10]: 21

In [11]:

Out[11]:
```
(9, 1)
```

In [...]:
```
# MODEL CREATION
# model create by using the formula SLR,Linerregression to be loaded in t
# variable assigned and named as regressor,loaded the formula here
# fit is used for SUBSITUTE the train set.
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)
```

Out[12]:
```
LinearRegression()
```

In [13]:
```
weight=regressor.coef_
```

In [15]:
```
# find the weight in regressor,ratio incresed
weight
```

Out[15]:
```
array([[9360.26128619]])
```

In [16]:
```
# strarting range of the pattern
bias=regressor.intercept_
print(bias)
```

```
[26777.3913412]
```

In [17]:
```
# EVALUATION METRICS
# Evaluation process.X_test variables predict and saved in y_pred
y_pred=regressor.predict(X_test)
```

In [18]:
```
# pridicted values
y_pred
```

Out[18]:
```
array([[ 40817.78327049],
       [123188.08258899],
       [ 65154.46261459],
       [ 63282.41035735],
       [115699.87356004],
       [108211.66453108],
       [116635.89968866],
       [ 64218.43648597],
       [ 76386.77615802]])
```

In [...]:
```
# New data table created for actual value and predicted value (Empty tabl
predActual=pd.DataFrame(index=range(0,10))
```

In [21]:
```
predActual["Actualvalue"]=y_test
```

In [22]:
```
y_test.index=range(0,9)
```

In [23]:
```
# Test set tabulated
y_test
```

Out[23]:

	Salary
0	37731
1	122391
2	57081
3	63218
4	116969
5	109431
6	112635
7	55794
8	83088

In [24]:
```
predActual["Actualvalue"]=y_test
```

In [25]:
```
y_pred_table=pd.DataFrame(y_pred,columns=["pred"])
```

In [26]:
```
# Predicted value tabulated
y_pred_table
```

Out[26]:

	pred
0	40817.783270
1	123188.082589
2	65154.462615
3	63282.410357
4	115699.873560
5	108211.664531
6	116635.899689
7	64218.436486
8	76386.776158

In [27]:
```
predActual["predvalue"]=y_pred_table
```

In [28]:
```
y_pred
```

Out[28]:
```
array([[ 40817.78327049],
       [123188.08258899],
       [ 65154.46261459],
       [ 63282.41035735],
       [115699.87356004],
       [108211.66453108],
       [116635.89968866],
       [ 64218.43648597],
       [ 76386.77615802]])
```

Out[29]:

	Actualvalue	predvalue
0	37731.0	40817.783270
1	122391.0	123188.082589
2	57081.0	65154.462615
3	63218.0	63282.410357
4	116969.0	115699.873560
5	109431.0	108211.664531
6	112635.0	116635.899689
7	55794.0	64218.436486
8	83088.0	76386.776158
9	NaN	NaN

In [...
```
# Evaluation metrics  to be done between test set and predicted value and
from sklearn.metrics import r2_score
r=r2_score(y_test,y_pred)
```

In [31]:
```
# preicted Model is good
r
```

Out[31]:
```
0.9740993407213511
```

In [32]:
```
# SAV THE BEST MODEL
# pickle library imported and final model saved in the library to write
import pickle
filename="finalModel.sav"
pickle.dump(regressor,open(filename,'wb'))
```

In [33]:
```
# final model saved as sav file and assigned to load model
load_model=pickle.load(open("finalModel.sav",'rb'))
```

In [34]:
```
# result derived from load model
result=load_model.predict([[15]])
```

```
C:\Users\Hi\anaconda3\lib\site-packages\sklearn\base.py:450: UserWarning: X do
es not have valid feature names, but LinearRegression was fitted with feature
names
  warnings.warn(
```

In [35]:
```
# final result of the salary for 15 years of experience
result
```

Out[35]:
```
array([[167181.3106341]])
```

In [ ]: