In [77]:
```
#Name:Khushi Chandrashekhar Satpute
#Aim: To Perform and analysis of Linear Regression Algorithm
#Roll No:43
#Sec:B
#Sub:ET-II
```

In [79]:
```
import pandas as pd
import numpy as np
import os
```

In [81]:
```
os.getcwd()
```

Out[81]:
```
'C:\\Users\\asus\\Desktop'
```

In [91]:
```
os.chdir("C:\\Users\\asus\\Desktop")
```

In [93]:
```
data=pd.read_csv("salary.csv")
```

In [95]:
```
data.head()
```

Out[95]:

	YearsExperience	Salary
0	1.1	39343
1	1.3	46205
2	1.5	37731
3	2.0	43525
4	2.2	39891

In [97]:
```
data.tail()
```

Out[97]:

	YearsExperience	Salary
30	11.2	127345
31	11.5	126756
32	12.3	128765
33	12.9	135675
34	13.5	139465

In [99]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 35 entries, 0 to 34
Data columns (total 2 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   YearsExperience  35 non-null     float64
 1   Salary           35 non-null     int64  
dtypes: float64(1), int64(1)
memory usage: 692.0 bytes
```

In [101…
```
data.describe()
```

Out[101…

	YearsExperience	Salary
count	35.000000	35.000000
mean	6.308571	83945.600000
std	3.618610	32162.673003
min	1.100000	37731.000000
25%	3.450000	57019.000000
50%	5.300000	81363.000000
75%	9.250000	113223.500000
max	13.500000	139465.000000

In [103…
```
data.shape
```

Out[103…
```
(35, 2)
```

In [105…
```
data.size
```

Out[105… 70

In [107…
```
data.ndim
```

Out[107… 2

data.isnull()

In [111…
```
data.isnull().any()
```

Out[111…
```
YearsExperience    False
Salary             False
dtype: bool
```

In [113…
```
data.isna().sum()
```

Out[113…
```
YearsExperience    0
Salary             0
dtype: int64
```

In [119…
```
X = data.iloc[:, :-1].values
y = data.iloc[:, -1].values



#X = data['YearsExperience']
#y = data['Salary']
```

In [121…
```
print(X)
```

```
[[ 1.1]
 [ 1.3]
 [ 1.5]
 [ 2. ]
 [ 2.2]
 [ 2.9]
 [ 3. ]
 [ 3.2]
 [ 3.2]
 [ 3.7]
 [ 3.9]
 [ 4. ]
 [ 4. ]
 [ 4.1]
 [ 4.5]
 [ 4.9]
 [ 5.1]
 [ 5.3]
 [ 5.9]
 [ 6. ]
 [ 6.8]
 [ 7.1]
 [ 7.9]
 [ 8.2]
 [ 8.7]
 [ 9. ]
 [ 9.5]
 [ 9.6]
 [10.3]
 [10.5]
 [11.2]
 [11.5]
 [12.3]
 [12.9]
 [13.5]]
```

In [125…
```
#splitting the data into training and testing data sets
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(X,y,test_size=0.3 ,random_state=42)
```

In [129…
```
print(x_train)
```

```
[[12.9]
 [ 1.1]
 [ 2.2]
 [ 5.3]
 [ 9.6]
 [ 2.9]
 [ 4. ]
 [ 1.3]
 [ 1.5]
 [12.3]
 [ 2. ]
 [11.2]
 [ 8.2]
 [11.5]
 [ 3.9]
 [ 7.9]
 [ 5.9]
 [ 9. ]
 [ 3. ]
 [ 6.8]
 [13.5]
 [ 3.2]
 [ 4.5]
 [10.3]]
```

In [133…
```
x_test
```

Out[133…
```
array([[ 9.5],
       [ 4.1],
       [ 8.7],
       [ 7.1],
       [ 4.9],
       [10.5],
       [ 6. ],
       [ 4. ],
       [ 3.2],
       [ 5.1],
       [ 3.7]])
```

In [135…
```
y_train
```

Out[135…
```
array([135675,  39343,  39891,  83088, 112635,  56642,  55794,  46205,
        37731, 128765,  43525, 127345, 113812, 126756,  63218, 101302,
        81363, 105582,  60150,  91738, 139465,  54445,  61111, 122391],
      dtype=int64)
```

In [137…
```
y_test
```

Out[137…
```
array([116969,  57081, 109431,  98273,  67938, 121872,  93940,  56957,
        64445,  66029,  57189], dtype=int64)
```

In [141…
```
from sklearn.linear_model import LinearRegression
lr = LinearRegression()
lr.fit(x_train, y_train)
```

Out[141…

▾ LinearRegression

```
LinearRegression()
```

In [143…
```
#Assigning Coefficient (slope) to m
m = lr.coef_
```

In [145…
```
print("Coefficient  :" , m)
```

```
Coefficient  : [8555.33918938]
```

In [147…
```
#Assigning Y-intercept to a
c = lr.intercept_
```

In [149…
```
print("Intercept : ", c)
```

```
Intercept :  29602.07353482097
```

In [153…
```
lr.score(x_test,y_test) * 100
```

Out[153…
```
91.71426108885095
```

In [ ]: