

```
In [2]: #Aim : To perform Simple Linear Regression and find out the coefficients of it.
```

```
In [3]: # Name : Shruti Dhote  
# Roll no : 72  
# Sec: C  
# Subject : ET1  
# Date : 06/09/2024
```

```
In [4]: import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns  
import numpy as np
```

```
In [5]: import os
```

```
In [6]: os.getcwd()
```

```
Out[6]: 'C:\\Users\\SURUTI DHOTE'
```

```
In [9]: os.chdir("C:\\Users\\SURUTI DHOTE\\Desktop\\")
```

```
In [10]: df=pd.read_csv("Salary.csv")
```

```
In [11]: df.head()
```

```
Out[11]:
```

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

```
In [12]: df.head(10)
```

```
Out[12]:
```

	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

In [13]: `df.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 [14]: `df.tail()`

Out[14]:

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

In [15]: `df.describe()`

Out[15]:

	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 [16]: `df.shape`

Out[16]: (35, 2)

In [17]: `df.size`

Out[17]: 70

In [18]: `df.ndim`

Out[18]: 2

In [19]: `df.isnull()`

Out[19]:

	YearsExperience	Salary
0	False	False
1	False	False
2	False	False
3	False	False
4	False	False
5	False	False
6	False	False
7	False	False
8	False	False
9	False	False
10	False	False
11	False	False
12	False	False
13	False	False
14	False	False
15	False	False
16	False	False
17	False	False
18	False	False
19	False	False
20	False	False
21	False	False
22	False	False
23	False	False
24	False	False
25	False	False
26	False	False
27	False	False
28	False	False
29	False	False
30	False	False
31	False	False
32	False	False
33	False	False
34	False	False

```
In [20]: df.isnull()
```

Out[20]:

	YearsExperience	Salary
0	False	False
1	False	False
2	False	False
3	False	False
4	False	False
5	False	False
6	False	False
7	False	False
8	False	False
9	False	False
10	False	False
11	False	False
12	False	False
13	False	False
14	False	False
15	False	False
16	False	False
17	False	False
18	False	False
19	False	False
20	False	False
21	False	False
22	False	False
23	False	False
24	False	False
25	False	False
26	False	False
27	False	False
28	False	False
29	False	False
30	False	False
31	False	False
32	False	False
33	False	False
34	False	False

```
In [21]: df.isnull().sum()
```

```
Out[21]: YearsExperience    0  
Salary                  0  
dtype: int64
```

```
In [22]: df.head()
```

```
Out[22]:
```

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

```
In [23]: df.columns
```

```
Out[23]: Index(['YearsExperience', 'Salary'], dtype='object')
```

```
In [24]: df.loc[4, "Salary"]
```

```
Out[24]: 39891
```

```
In [25]: df.head(15)
```

```
Out[25]:
```

	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

```
In [26]: df.loc[2, "YearsExperience"]
```

```
Out[26]: 1.5
```

```
In [27]: df.loc[12]
```

```
Out[27]: YearsExperience    4.0  
Salary          56957.0  
Name: 12, dtype: float64
```

```
In [28]: df.loc[4]
```

```
Out[28]: YearsExperience    2.2  
Salary          39891.0  
Name: 4, dtype: float64
```

```
In [29]: a=(1,2,3,4,5,6,7,8,9,10)
```

```
In [30]: a[1:4]
```

```
Out[30]: (2, 3, 4)
```

```
In [31]: df.loc[0:3, 'YearsExperience': 'Salary']
```

```
Out[31]:   YearsExperience  Salary  
0           1.1    39343  
1           1.3    46205  
2           1.5    37731  
3           2.0    43525
```

```
In [32]: df.iloc[1,0]
```

```
Out[32]: 1.3
```

```
In [33]: df.head()
```

```
Out[33]:   YearsExperience  Salary  
0           1.1    39343  
1           1.3    46205  
2           1.5    37731  
3           2.0    43525  
4           2.2    39891
```

```
In [34]: df.loc[1, "Salary"]
```

```
Out[34]: 46205
```

```
In [35]: #Assigning values in X & Y  
x=df.iloc[:, :-1].values  
y=df.iloc[:, :-1].values
```

```
In [37]: a[:2]
```

```
Out[37]: (1, 2)
```

```
In [39]: a[2:]
```

Out[39]: (3, 4, 5, 6, 7, 8, 9, 10)

In [40]: a[1:6:2]

Out[40]: (2, 4, 6)

In [41]: 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 [42]: print(y)


```
[[ 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 [43]: #splitting testdata into x_train,y_train'
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=.3, random_stat
```

```
In [44]: 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 [45]: print(x_test)
```

```
[[ 9.5]
 [ 4.1]
 [ 8.7]
 [ 7.1]
 [ 4.9]
 [10.5]
 [ 6. ]
 [ 4. ]
 [ 3.2]
 [ 5.1]
 [ 3.7]]
```

```
In [46]: print(y_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 [47]: print(y_test)
```

```
[[ 9.5]
 [ 4.1]
 [ 8.7]
 [ 7.1]
 [ 4.9]
 [10.5]
 [ 6. ]
 [ 4. ]
 [ 3.2]
 [ 5.1]
 [ 3.7]]
```

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

```
In [ ]: #Assigning coefficient (slope) to m
m = lr.coef_
```

```
In [ ]: print("Coefficient :",a)
```