Experiment-6

6) Consider the Student dataset and train Simple Linear Regression to predict student scores based on studay Hour and also evaluate the model using different matrices

In [1]:
```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error,r2_score
```

In [2]:
```
##Simple Linear Regression Model to predict student scores based on study hours
```

In [3]:
```
df = pd.read_csv('C:/Users/Guru Kiran/All CSV files/score.csv')
df.head(5)
```

Out[3]:

	Hours	Scores
0	2.5	21.0
1	5.1	47.0
2	3.2	27.0
3	8.5	75.0
4	3.5	30.0

In [4]:
```
df.isnull().sum()
```

Out[4]:
```
Hours     0
Scores    2
dtype: int64
```

In [5]:
```
df.dropna(inplace=True,axis=0)
df.head(5)
```

Out[5]:

	Hours	Scores
0	2.5	21.0
1	5.1	47.0
2	3.2	27.0
3	8.5	75.0
4	3.5	30.0

In [6]:
```
df.isnull().sum()
```

Out[6]:
```
Hours     0
Scores    0
dtype: int64
```

In [7]:
```
Y = df['Scores']
X = df.drop('Scores',axis=1)
```

In [8]:
```
X.head()
```

Out[8]:

	Hours
0	2.5
1	5.1
2	3.2
3	8.5
4	3.5

In [9]:
```
# Split the data into training and testing sets
x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=0.25)
```

In [10]:
```
## Build the model
```

In [11]:
```
lr = LinearRegression()
model = lr.fit(x_train,y_train)
y_pred = model.predict(x_test)
```

In [12]:
```
df = pd.DataFrame({'y_test':y_test,'y_pred':y_pred})
df
```

Out[12]:

	y_test	y_pred
1	47.0	51.968412
7	60.0	55.736910
2	27.0	34.068046
15	95.0	87.769144
16	30.0	27.473174
10	85.0	76.463650

In [13]:
```
print("The MSE is : ", mean_squared_error(y_test,y_pred))
print("The R square is : ", r2_score(y_test, y_pred))
```

```
The MSE is :  37.39261959263761
The R square is :  0.943420716823514
```