EX NO: 8 DATE :	MULTIPLE LINEAR REGRESSION

AIM:

To write a python program to implement multiple linear regression.

CODE:

```
import pandas as pd
import numpy as np
import sklearn as sk
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import confusion_matrix
```

```
data = pd.read_csv("../input/advertising-
data/Advertising.csv",index_col=False)
data.head()
```

	Unnamed: 0	TV	Radio	Newspaper	Sales
0	1	230.1	37.8	69.2	22.1
1	2	44.5	39.3	45.1	10.4
2	3	17.2	45.9	69.3	9.3
3	4	151.5	41.3	58.5	18.5
4	5	180.8	10.8	58.4	12.9

```
x = data.iloc[:,0:-1]
y = data.iloc[:,-1]
```

```
train_x,test_x,train_y,test_y = train_test_split(x,y,test_size=0.2)
```

```
model = LinearRegression()
model.fit(train_x,train_y)
```

```
LinearRegression()
```

```
print("Intercept: ",model.intercept_)
print("Coefficients :")
list(zip(x,model.coef_))
```

```
Intercept:  3.3315105240830647
Coefficients :

[("TV", 0.04477428406990247),
 ("Radio", 0.18540600918250505),
 ("Newspaper", -0.0040379945017015445)]
```

```
y_pred = model.predict(test_x)
```

```
print("Prediction for test set : {} ",format(y_pred))
```

```
Prediction for test set : {}  [11.71654425 18.32738979 12.11202414
12.25612314  6.88525856 20.5921916
 16.23214688  9.97604149 19.0517318  12.30049651  9.59220955 15.11127306
  9.85967607 14.73372645 16.40525643 12.69273555 23.54923464 15.67284995
 21.04075073 10.79170368 20.72679435 18.1973761  12.80400592 12.67386262
 12.42830877 18.26238969 19.73207535 16.80955259 11.78184574 21.06806566
 10.66979993 13.866794   21.15389281 12.52912335  8.70109868 10.2257276
 10.68872118 15.01946175 23.1266277  17.66345253]
```

```
mlr_diff = pd.DataFrame({"Actual value": test_y,"Predicted value": y_pred})
mlr_diff.head()
```

	Actual value	Predicted value
88	12.9	11.716544
150	16.1	18.327390
116	12.2	12.112024
177	11.7	12.256123
127	8.8	6.885259

```
from sklearn import metrics
```

```
meanAbErr = metrics.mean_absolute_error(test_y,y_pred)
meanSqErr = metrics.mean_squared_error(test_y,y_pred)
rootMeanSqErr = np.sqrt(metrics.mean_squared_error(test_y,y_pred))
```

```
print("R squared: {:,2f}",format(model.score(x,y)*100))
print("Mean Absolute Error:",meanAbErr)
print("Mean Square Error:",meanSqErr)
print("Root Mean Square Error:",rootMeanSqErr)
```

```
R squared: {:,2f} 89.63391600098502
Mean Absolute Error: 1.4977438835047738
Mean Square Error: 4.774886541440521
Root Mean Square Error: 2.185151377236946
```

RESULT:

Thus the python program to implement multiple linear regression is executed and verified.