In [32]:
```
import pandas as pd
```

In [161…
```
df=pd.read_csv("./BostonHousing.csv")
```

In [162…
```
df
```

Out[162…

	crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	b	l
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	5
...	...	...	...	...	...	...	...	...	...	...	...	...	
501	0.06263	0.0	11.93	0	0.573	6.593	69.1	2.4786	1	273	21.0	391.99	9
502	0.04527	0.0	11.93	0	0.573	6.120	76.7	2.2875	1	273	21.0	396.90	9
503	0.06076	0.0	11.93	0	0.573	6.976	91.0	2.1675	1	273	21.0	396.90	5
504	0.10959	0.0	11.93	0	0.573	6.794	89.3	2.3889	1	273	21.0	393.45	6
505	0.04741	0.0	11.93	0	0.573	6.030	80.8	2.5050	1	273	21.0	396.90	7

506 rows × 14 columns

In [163…
```
x=df.drop("medv",axis=1)
```

In [164…
```
y=df["medv"]
```

In [165…
```
x.shape
```

Out[165…
```
(506, 13)
```

In [166…
```
from sklearn.model_selection import train_test_split
```

In [167…
```
x_train,x_test,y_train,y_test=train_test_split(x,y,random_state=0, test_size
```

In [168…
```
x_train
```

Out[168…

	crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	b
245	0.19133	22.0	5.86	0	0.431	5.605	70.2	7.9549	7	330	19.1	389.13
59	0.10328	25.0	5.13	0	0.453	5.927	47.2	6.9320	8	284	19.7	396.90
276	0.10469	40.0	6.41	1	0.447	7.267	49.0	4.7872	4	254	17.6	389.25
395	8.71675	0.0	18.10	0	0.693	6.471	98.8	1.7257	24	666	20.2	391.98
416	10.83420	0.0	18.10	0	0.679	6.782	90.8	1.8195	24	666	20.2	21.57
...	...	...	...	...	...	...	...	...	...	...	...	...
323	0.28392	0.0	7.38	0	0.493	5.708	74.3	4.7211	5	287	19.6	391.13
192	0.08664	45.0	3.44	0	0.437	7.178	26.3	6.4798	5	398	15.2	390.49
117	0.15098	0.0	10.01	0	0.547	6.021	82.6	2.7474	6	432	17.8	394.51
47	0.22927	0.0	6.91	0	0.448	6.030	85.5	5.6894	3	233	17.9	392.74
172	0.13914	0.0	4.05	0	0.510	5.572	88.5	2.5961	5	296	16.6	396.90

379 rows × 13 columns

In [210…

```
x_train.shape
```

Out[210…

```
(379, 13)
```

In [211…

```
x_test.shape
```

Out[211…

```
(127, 13)
```

In [212…

```
from sklearn.linear_model import LinearRegression
```

In [213…

```
regressor=LinearRegression()
```

In [214…

```
regressor.fit(x_train,y_train)
```

Out[214…

```
▾ LinearRegression i ?

LinearRegression()
```

In [215…

```
LinearRegression()
```

Out[215…

```
▾ LinearRegression i ?

LinearRegression()
```

In [216…

```
regressor.coef_
```

Out[216…
```
array([-1.17735289e-01,  4.40174969e-02, -5.76814314e-03,  2.39341594e+00,
       -1.55894211e+01,  3.76896770e+00, -7.03517828e-03, -1.43495641e+00,
        2.40081086e-01, -1.12972810e-02, -9.85546732e-01,  8.44443453e-03,
       -4.99116797e-01])
```

In [217…
```
regressor.intercept_
```

Out[217…
```
36.93325545711923
```

In [218…
```
y_pred=regressor.predict(x_test)
```

In [219…
```
y_pred.shape
```

Out[219…
```
(127,)
```

In [220…
```
result=pd.DataFrame({'Actual':y_test, 'Producted':y_pred})
```

In [221…
```
result
```

Out[221…

	Actual	Producted
329	22.6	24.952333
371	50.0	23.616997
219	23.0	29.205886
403	8.3	11.960705
78	21.2	21.333620
...	...	...
49	19.4	17.538048
498	21.2	21.502223
309	20.3	23.632813
124	18.8	20.282598
306	33.4	35.179734

127 rows × 2 columns

In [222…
```
residual_error=abs(y_test-y_pred)
```

In [223…
```
residual_error;
```

In [224…
```
sum(residual_error)/len(residual_error)
```

Out[224…
```
3.66833014813572
```

In [225…
```
from sklearn.metrics import mean_absolute_error
mean_absolute_error(y_pred,y_test)
```

```
Out[225…  3.668330148135719
```

```
In [226…  from sklearn.metrics import mean_absolute_percentage_error
```

```
In [227…  mean_absolute_percentage_error(y_test,y_pred)
```

```
Out[227…  0.1754993780061571
```

```
In [228…  regressor.score(x_test,y_test)
```

```
Out[228…  0.6354638433202128
```

```
In [229…  from sklearn.metrics import r2_score
          r2_score(y_test,y_pred)
```

```
Out[229…  0.6354638433202128
```

```
In [230…  import matplotlib.pyplot as plt
          %matplotlib inline
          import seaborn as sns
```

```
In [231…  model = LinearRegression(n_jobs=-1)
          model.fit(x_train, y_train)
```

```
Out[231…  ▾   LinearRegression  i ?

          LinearRegression(n_jobs=-1)
```

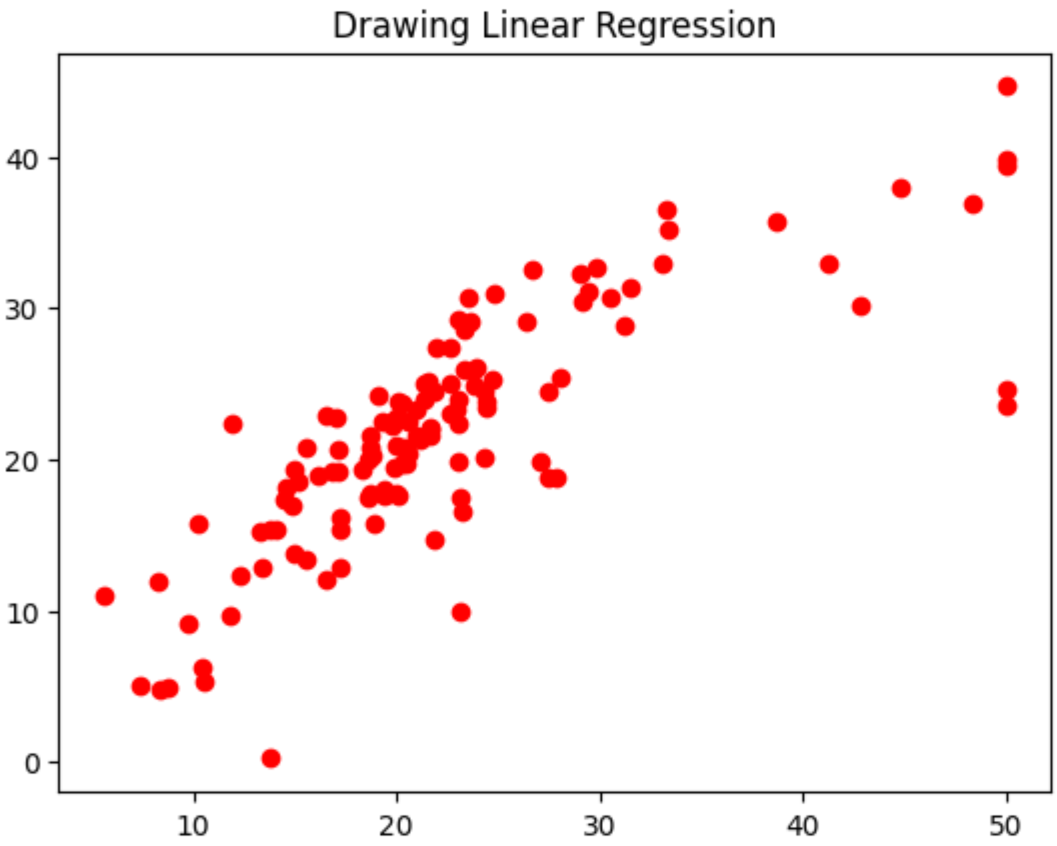
```
In [232…  y_test
```

```
Out[232…  329    22.6
          371    50.0
          219    23.0
          403     8.3
          78     21.2
                 ...
          49     19.4
          498    21.2
          309    20.3
          124    18.8
          306    33.4
          Name: medv, Length: 127, dtype: float64
```

```
In [233…  from sklearn.preprocessing import StandardScaler
```

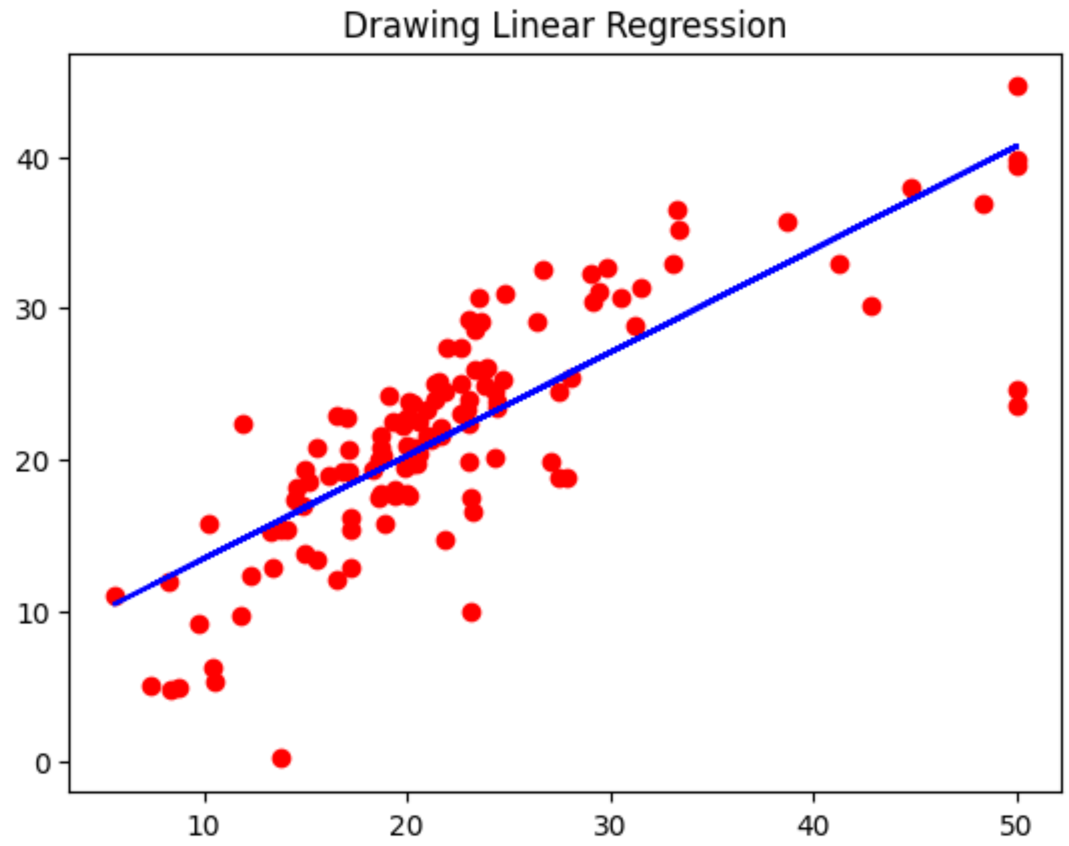
```
In [234…  y_pred = model.predict(x_test)
          plt.scatter(y_test, y_pred, color='red')
          z=np.polyfit(y_test,y_pred,1)
          p=np.poly1d(z)
          plt.title('Drawing Linear Regression')
```

```
Out[234…  Text(0.5, 1.0, 'Drawing Linear Regression')
```



In [235…

```
plt.scatter(y_test, y_pred, color='red')
plt.title('Drawing Linear Regression')
plt.plot(y_test,p(y_test), color='Blue')
plt.show()
```


Drawing Linear Regression
0
10
20
30
40
10
20
30
40
50