Importing Modules and Loading Dataset

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
df = pd.read_csv('HousingData.csv')
```

```
df.head()
```

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1	296	15.3	3
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2	242	17.8	3
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2	242	17.8	3
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3	222	18.7	3
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3	222	18.7	3

```
df.isnull().sum()
```

```
CRIM       20
ZN         20
INDUS      20
CHAS       20
NOX         0
RM          0
AGE        20
DIS         0
RAD         0
TAX         0
PTRATIO     0
B           0
LSTAT      20
MEDV        0
dtype: int64
```

```
df.dropna(inplace=True)
```

```
df.isnull().sum()
```

```
CRIM       0
ZN         0
INDUS      0
CHAS       0
NOX        0
RM         0
```

✓ 0s completed at 8:42 PM

```
DIS        0
RAD        0
TAX        0
PTRATIO    0
B          0
LSTAT      0
MEDV       0
dtype: int64
```

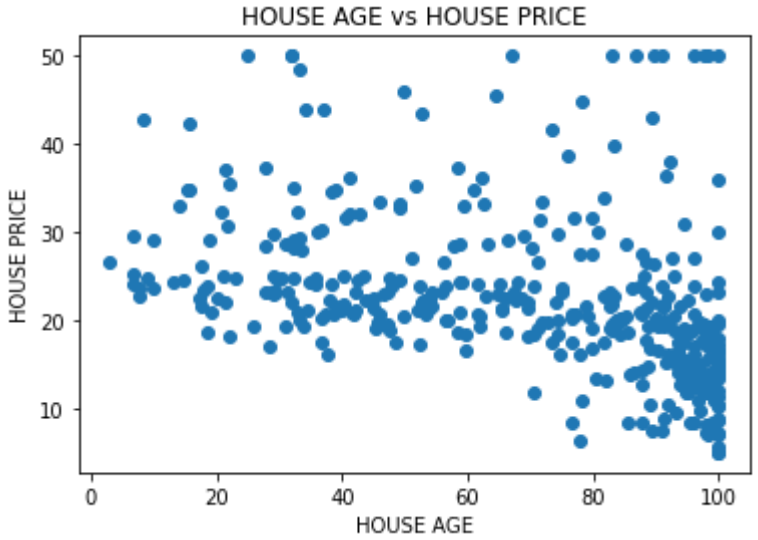
```
df = df.rename(columns={'CRIM':'CRIME', 'DIS':'DISTANCE', 'B':'BLACKS', 'MEDV':'
```

Visualising

HOUSE AGE VS HOUSE PRICE

```
plt.scatter(df['AGE'], df['PRICE'])
plt.xlabel('HOUSE AGE')
plt.ylabel('HOUSE PRICE')
plt.title('HOUSE AGE vs HOUSE PRICE')
```

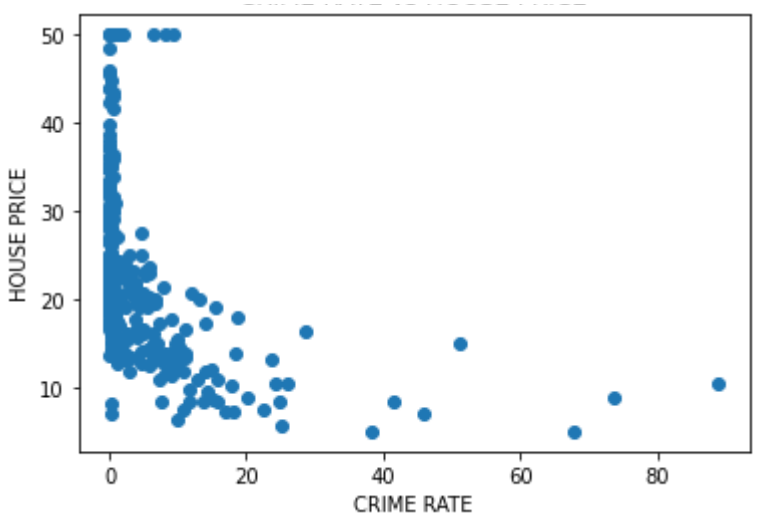
```
Text(0.5, 1.0, 'HOUSE AGE vs HOUSE PRICE')
```



CRIME RATE VS HOUSE PRICE

```
plt.scatter(df['CRIME'], df['PRICE'])
plt.xlabel('CRIME RATE')
plt.ylabel('HOUSE PRICE')
plt.title('CRIME RATE vs HOUSE PRICE')
```

```
Text(0.5, 1.0, 'CRIME RATE vs HOUSE PRICE')
```




```
df.columns
```

```
Index(['CRIME', 'ZN', 'INDUS', 'CHAS', 'NOX', 'RM', 'AGE', 'DISTANCE', 'RAI
       'TAX', 'PTRATIO', 'BLACKS', 'LSTAT', 'PRICE'],
      dtype='object')
```

Observations

- We can see that the older the house is the lower the price gets.
- We can see that the lower the crime rate is the higher thr price is.

Splitting Dataset

```
X = df[['CRIME', 'ZN', 'INDUS', 'CHAS', 'NOX', 'RM', 'AGE', 'DISTANCE', 'RAD',
       'TAX', 'PTRATIO', 'BLACKS', 'LSTAT']]
y = df['PRICE']
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_
```

Creating the Linear Regression Model and Fitting the Data

```
from sklearn.linear_model import LinearRegression
model = LinearRegression()
X.isnull().sum()
model.fit(X_train, y_train)
```

```
LinearRegression()
```

```
print(model.intercept_)
```

```
26.643103405316122
```

```
coeff_df = pd.DataFrame(model.coef_,X.columns,columns=['Coefficient'])
```

```
coeff_df
```

	Coefficient
CRIME	-0.071668
ZN	0.040781
INDUS	0.065085
CHAS	2.711994
NOX	-19.794461
RM	5.253264
AGE	-0.024198
DISTANCE	-1.470089
RAD	0.241940
TAX	-0.013832
PTRATIO	-0.896696
BLACKS	0.010394
LSTAT	-0.319934

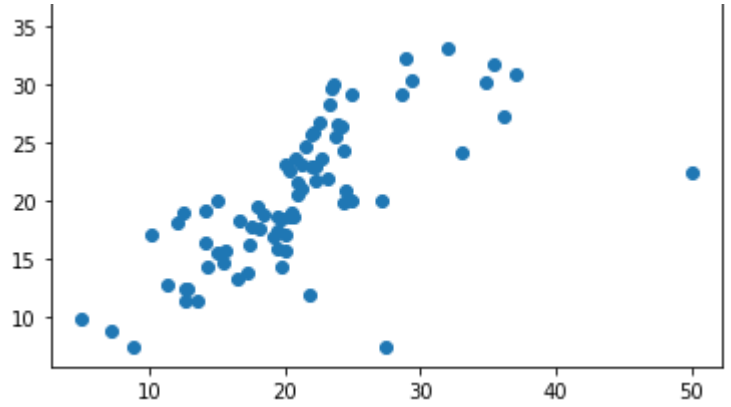
Predicting the Data

```
predictions = model.predict(X_test)
```

Plotting the Graph

```
plt.scatter(y_test, predictions)
```

```
<matplotlib.collections.PathCollection at 0x7ff7be4de5d0>
```




```
from sklearn import metrics


print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, predictions))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, predictions))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, pre
print('R^2:',metrics.r2_score(y_test, predictions))
```

```
Mean Absolute Error: 3.4843562556302334
Mean Squared Error: 28.849872777166873
Root Mean Squared Error: 5.3712077577735595
R^2: 0.5806563123015832
```