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

```
df = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/housing.csv')
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 506 entries, 0 to 505
Data columns (total 14 columns):
 #   Column   Non-Null Count  Dtype  
---  ------   --------------  -----  
 0   CRIM     506 non-null    float64
 1   ZN       506 non-null    float64
 2   INDUS    506 non-null    float64
 3   CHAS     506 non-null    int64  
 4   NOX      506 non-null    float64
 5   RM       506 non-null    float64
 6   AGE      506 non-null    float64
 7   DIS      506 non-null    float64
 8   RAD      506 non-null    int64  
 9   TAX      506 non-null    int64  
 10  PTRATIO  506 non-null    float64
 11  B        506 non-null    float64
 12  LSTAT    506 non-null    float64
 13  MEDV     506 non-null    float64
dtypes: float64(11), int64(3)
memory usage: 55.5 KB
```

```
df = df.rename(columns = {'MEDV':'price'})
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 506 entries, 0 to 505
Data columns (total 14 columns):
 #   Column   Non-Null Count  Dtype  
---  ------   --------------  -----  
 0   CRIM     506 non-null    float64
 1   ZN       506 non-null    float64
 2   INDUS    506 non-null    float64
 3   CHAS     506 non-null    int64  
 4   NOX      506 non-null    float64
 5   RM       506 non-null    float64
 6   AGE      506 non-null    float64
 7   DIS      506 non-null    float64
 8   RAD      506 non-null    int64  
 9   TAX      506 non-null    int64  
 10  PTRATIO  506 non-null    float64
 11  B        506 non-null    float64
 12  LSTAT    506 non-null    float64
 13  price    506 non-null    float64
```

```
dtypes: float64(11), int64(3)
memory usage: 55.5 KB
```

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

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

```
x = df.drop(['price'], axis=1)
```

```
y = df['price']
```

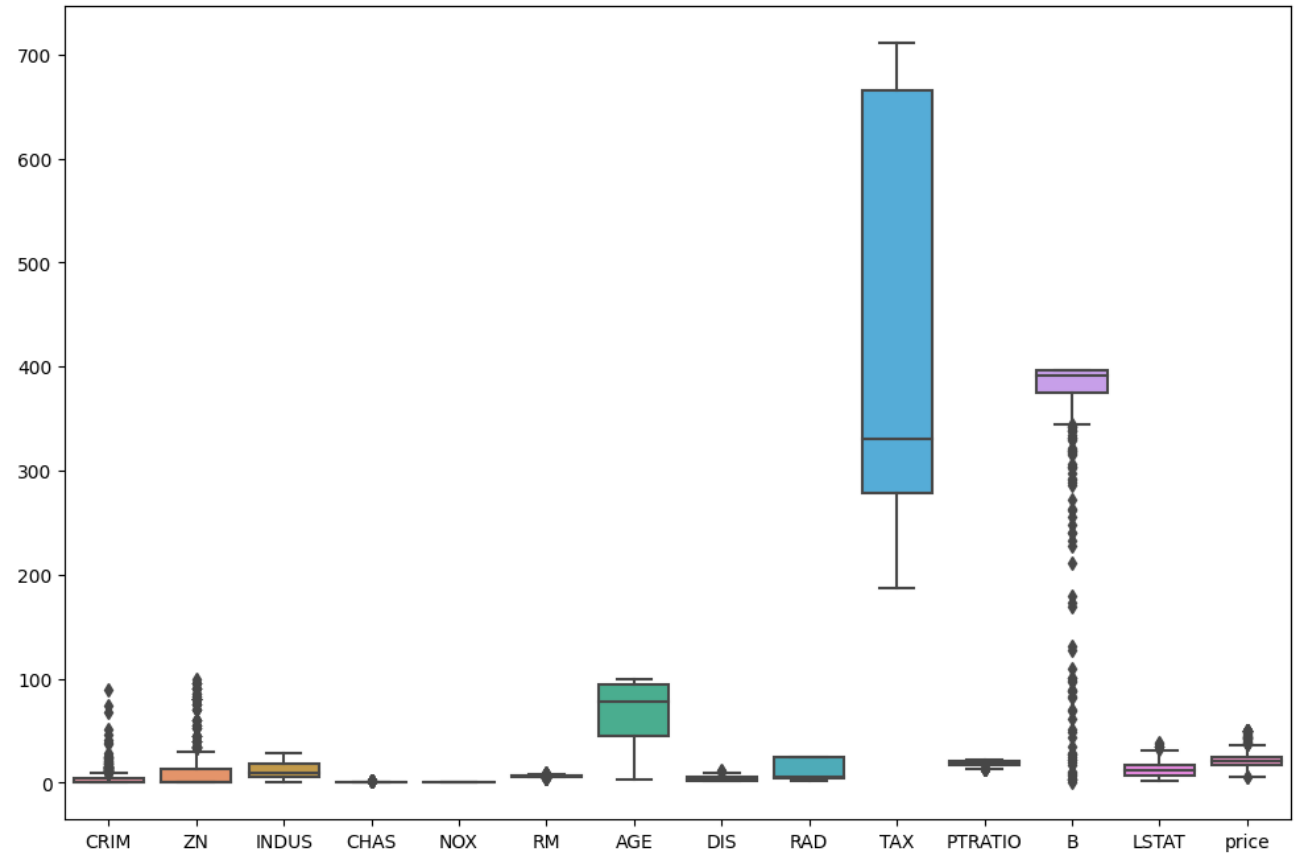
```
y
```

```
0      24.0
1      21.6
2      34.7
3      33.4
4      36.2
       ...
501    22.4
502    20.6
503    23.9
504    22.0
505    11.9
Name: price, Length: 506, dtype: float64
```

EXPLORATARY DATA ANALYSIS (EDA)

```
plt.figure(figsize=(12,8))
sns.boxplot(data = df)
```

```
<Axes: >
```




```
# splitting of to x_train, x_test, y_train, y_test
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_sta

# MODEL SELECTION
from sklearn.linear_model import LinearRegression

model=LinearRegression()

# MODEL TRAINING
model.fit(x_train, y_train)
```

```
▾ LinearRegression
LinearRegression()
```

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

```
y_pred
```

```
array([12.07495986, 26.9894969 , 17.58803353, 18.15584511, 36.92091659,
       25.43267386, 31.09256932, 19.72549907, 19.66103377, 22.96358632,
       28.38841214, 28.48925986, 18.99690357, 32.41097504, 21.52350275,
       15.25945122, 21.23364112, 11.6220597 , 11.37109662, 13.63515584,
        5.62431971, 17.35323315, 20.80951594, 22.51311312, 16.39055556,
       20.32352451, 17.88994185, 14.23445109, 21.1187098 , 17.50765806,
       14.54295525, 23.63289896, 34.32419647, 22.23027161, 16.82396516,
       20.16274383, 30.67665825, 35.61882904, 23.50372003, 24.66451121,
       36.91269871, 32.33290254, 19.11785719, 32.19546605, 33.42795148,
       25.52705821, 40.63477427, 18.21762788, 19.34587461, 23.80167377,
       33.42122982, 26.1451108 , 18.10363121, 28.19906437, 13.37486655,
       23.34019279, 24.44952678, 33.54973856, 16.71263275, 36.56402224,
       15.69684554, 18.55447039, 32.14543203, 15.49568061, 39.02363234,
       27.38174402, 31.96333419, 10.09436162, 19.13214621, 21.73038157,
       23.14682001, 22.82615401, 22.51245566, 28.21477189, 17.13262484,
       23.08039019, 16.65978367, 25.17892617, 13.68806399, 19.8195139 ,
       22.31237842, 20.24637447, 28.35989119, 19.12635952, 30.49206633,
       22.25649076, 29.98229473, 19.27750127, 23.73890345, 38.32216452,
       31.24781499, 41.92137782, 18.61466511, 37.47526878, 19.66151941,
       23.44504636, 26.55358092, 22.38454399,  9.59394823, 20.39499251,
        9.22793989, 27.36219976])
```

```
y_test
```

```
8      16.5
289    24.8
68     17.4
211    19.3
226    37.6
       ...
368    50.0
144    11.8
336    19.5
437     8.7
216    23.3
Name: price, Length: 102, dtype: float64
```

```
len(y_pred)
```

```
102
```

```
len(y_test)
```

```
102
```

```
# MODEL EVALUATION WITH EVALUATION METRICS LIKE MAE, MSE, RSQUARE

from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score


error1 = mean_absolute_error(y_test, y_pred)


error1
```

```
     3.3677909837965787
```

```
error2 = mean_squared_error(y_test, y_pred)


error2
```

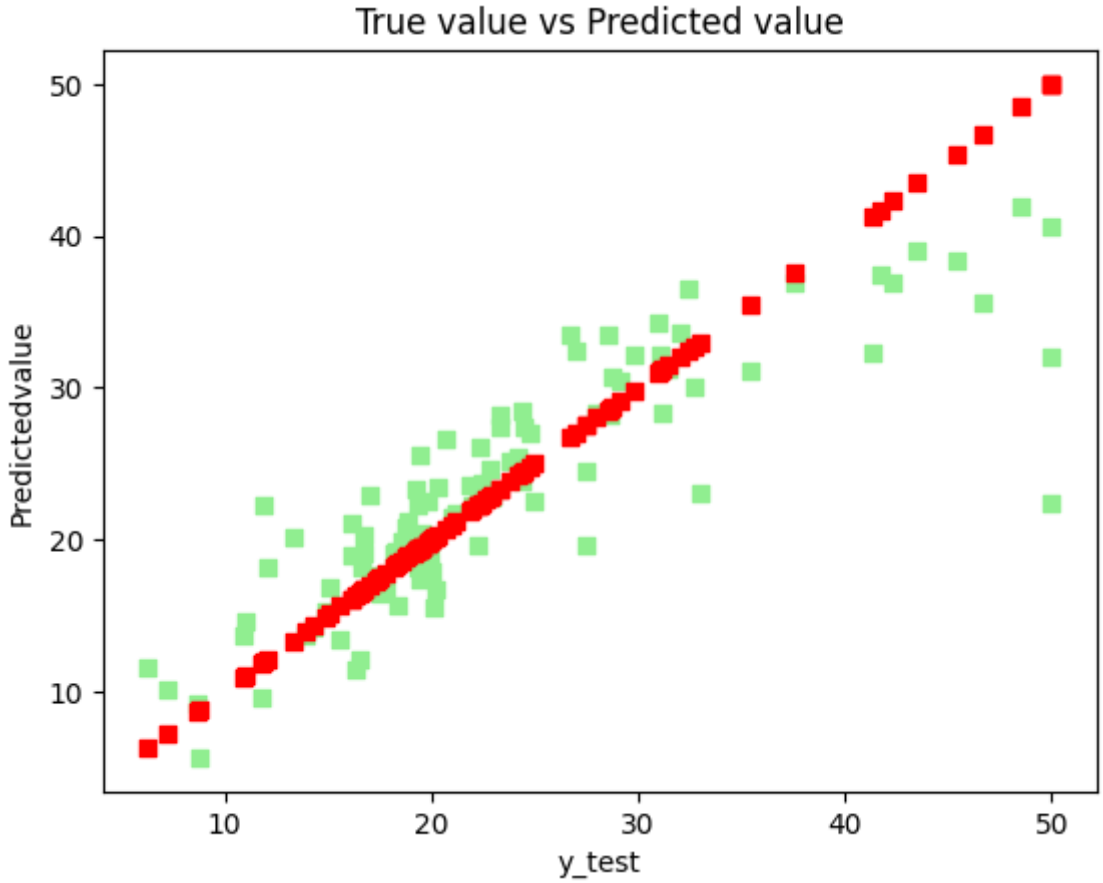
```
     25.419587126821785
```

```
error3 = r2_score(y_test, y_pred)


error3
```

```
     0.7263451459702517
```

```
# VISUALIZATION BY MATPLOTLIB AND SEABORN

plt.scatter(y_test,y_pred,c='lightgreen',marker='s',label='Test data')
plt.scatter(y_test,y_test,c="red",marker='s',label='Test data')
plt.xlabel('y_test')
plt.ylabel('Predictedvalue')
plt.title("True value vs Predicted value")
```

```
Text(0.5, 1.0, 'True value vs Predicted value')
```



✓ 0s completed at 11:23 AM