```
import numpy as np
import pandas as pd
import sklearn
from sklearn.datasets import load_boston
df = load_boston()
df.keys()
print(df.data)
boston = pd.DataFrame(df.data,columns=df.feature_names)
boston.head()
boston ['MEDV']=df.target
boston.head()
boston.isnull()
boston.isnull().sum()
from sklearn.model_selection import train_test_split
X=boston.drop('MEDV',axis=1)
Y=boston['MEDV']
X_train, X_test,Y_train,Y_test=train_test_split(X,Y,test_size=0.15,random_state=5)
print(X_train.shape)
print(X_test.shape)
print(Y_train.shape)
print(Y_test.shape)
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
lin_model=LinearRegression()
lin_model.fit(X_train,Y_train)
y_train_predict=lin_model.predict(X_train)
rmse=(np.sqrt(mean_squared_error(Y_train, y_train_predict)))
print('The model performance for training set')
print(rmse)
print("/n")
y_test_predict=lin_model.predict(X_test)
rmse=(np.sqrt(mean_squared_error(Y_test, y_test_predict)))
print ("The model performance for testing set")
print(rmse)
```

```
      /usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecation.py:87: FutureW

          The Boston housing prices dataset has an ethical problem. You can refer to
          the documentation of this function for further details.

          The scikit-learn maintainers therefore strongly discourage the use of this
          dataset unless the purpose of the code is to study and educate about
          ethical issues in data science and machine learning.

          In this special case, you can fetch the dataset from the original
          source::

              import pandas as pd
              import numpy as np


              data_url = "http://lib.stat.cmu.edu/datasets/boston"
              raw_df = pd.read_csv(data_url, sep="\s+", skiprows=22, header=None)
              data = np.hstack([raw_df.values[::2, :], raw_df.values[1::2, :2]])
              target = raw_df.values[1::2, 2]
```

```
    Alternative datasets include the California housing dataset (i.e.
    :func:`~sklearn.datasets.fetch_california_housing`) and the Ames housing
    dataset. You can load the datasets as follows::

        from sklearn.datasets import fetch_california_housing
        housing = fetch_california_housing()

    for the California housing dataset and::

        from sklearn.datasets import fetch_openml
        housing = fetch_openml(name="house_prices", as_frame=True)

    for the Ames housing dataset.

  warnings.warn(msg, category=FutureWarning)
[[6.3200e-03 1.8000e+01 2.3100e+00 ... 1.5300e+01 3.9690e+02 4.9800e+00]
 [2.7310e-02 0.0000e+00 7.0700e+00 ... 1.7800e+01 3.9690e+02 9.1400e+00]
 [2.7290e-02 0.0000e+00 7.0700e+00 ... 1.7800e+01 3.9283e+02 4.0300e+00]
 ...
 [6.0760e-02 0.0000e+00 1.1930e+01 ... 2.1000e+01 3.9690e+02 5.6400e+00]
 [1.0959e-01 0.0000e+00 1.1930e+01 ... 2.1000e+01 3.9345e+02 6.4800e+00]
 [4.7410e-02 0.0000e+00 1.1930e+01 ... 2.1000e+01 3.9690e+02 7.8800e+00]]
(430, 13)
(76, 13)
(430,)
(76,)
The model performance for training set
4.710901797319796
/n
The model performance for testing set
4.687543527902972
```