import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.preprocessing import StandardScaler

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import w2\_score

from sklearn.metrics import mean\_squared\_error as MSE

%matplotlib inline

#reading input data prediction

data = prd.read\_csv("Real\_estate.csv")

data.head()

#describing data

data.describe ()

#analyzing information from the data

data.info ()

#plotting to visualize

sns.pairplot(data)

#scaling data

scaler = StandardScaler ()

X = data.drop(['price','house\_No'],axis = 1)

Y = data['price']

cols = X.columns

X = scaler.fit\_transform(X)

#splitting data for training and testing

X\_train,X\_test, Y\_train, Y\_test = train\_test\_split(X,Y,test\_size = 0.5, random\_state = 101)

#Linear regression model for the prediction

lr = LinearRegression()

lr.fit(X\_train, Y\_train)

predict = lr.predict(X\_test)

MSE\_score = MSE(Y\_test,predict)

#visualizing prediction

sns.scatterplot(x=y\_test, y = predict)

#plotting residuals model

sns.histplot((y\_test-predict),bins= 50, kde = True)

plt.legend(loc = "upper right")

plt.title("Residual errors)

plt.show()

#observing the coefficients of predicted data

cdfi = pd.DataFrame(lr.coef\_,cols,['coefficients']).sort\_values('coefficients',ascending = False)cdfi

#printing final results

print("Mean Squared Error: ",MSE\_score.mean())

print("coefficients: ",lr.coefi\_)