**PRACTICAL NO : 04**

**DATA ANALYTICS 1**

**CODE :**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

data=pd.read\_csv(r"C:\Users\LP Lab\Documents\DSBDA datasets\housing.csv")

print(data)

x=np.array([95,85,80,70,60])

y=np.array([85,95,70,65,70])

model=np.polyfit(x,y,1)

model

predict=np.poly1d(model)

predict(65)

y\_pred=predict(x)

y\_pred

from sklearn.metrics import r2\_score

r2\_score(y,y\_pred)

y\_line=model[1]+model[0]\*x

plt.plot(x,y\_line,c='r')

plt.scatter(x,y\_pred)

plt.scatter(x, y,c='r')

from sklearn.datasets import fetch\_california\_housing

housing=fetch\_california\_housing()

data=pd.DataFrame(housing.data)

data.columns=housing.feature\_names

data.head()

data['PRICE']=housing.target

data.isnull().sum()

x=data.drop(['PRICE'],axis=1)

y=data['PRICE']

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

xtrain,xtest,ytrain,ytest=train\_test\_split(x,y,test\_size=0.2,random\_state=0)

from sklearn.linear\_model import LinearRegression

lm=LinearRegression()

model=lm.fit(xtrain, ytrain)

ytrain\_pred=lm.predict(xtrain)

ytest\_pred=lm.predict(xtest)

df=pd.DataFrame(ytrain\_pred,ytrain)

df=pd.DataFrame(ytest\_pred,ytest)

from sklearn.metrics import mean\_squared\_error,r2\_score

mse=mean\_squared\_error(ytest, ytest\_pred)

print(mse)

mse=mean\_squared\_error(ytrain\_pred,ytrain)

print(mse)

mse=mean\_squared\_error(ytest,ytest\_pred)

print(mse)

plt.scatter(ytrain,ytrain\_pred,c='blue',marker='o',label='Training data')

plt.scatter(ytest,ytest\_pred,c='lightgreen',marker='s',label='Test data')

plt.xlabel('True values')

plt.ylabel('Predicted')

plt.title('True value vs Predicted value')

plt.legend(loc='upper left')

#plt.hlines(y=0,xmin=0,xmax=50)

plt.plot()

plt.show()

**OUTPUT :**

