Implementation-of-Simple-Linear-Regression-Model-for-Predicting-the-Marks-Scored

AIM:

To write a program to predict the marks scored by a student using the simple linear regression model.

Equipments Required:

1. Hardware – PCs
2. Anaconda – Python 3.7 Installation / Jupyter notebook

Algorithm

1.
2.
3.
4.

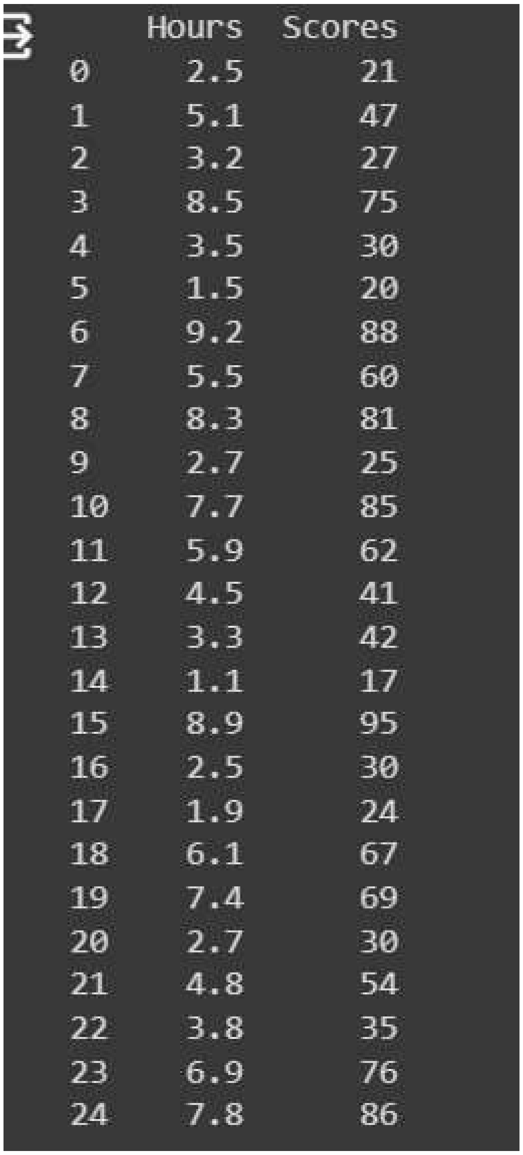
Program:

```
/*
Program to implement the simple linear regression model for predicting the marks scored.
Developed by: Dharshni V M
RegisterNumber: 212223240029

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import mean_absolute_error,mean_squared_error
df=pd.read_csv('student_scores.csv')
print(df)
df.head(2)
df.tail(4)
print(df.head())
print(df.tail())
x = df.iloc[:,:-1].values
print(x)
y = df.iloc[:,1].values
print(y)
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=1/3,random_state=0)
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(x_train,y_train)
y_pred = regressor.predict(x_test)
print(y_pred)
print(y_test)

plt.scatter(x_train,y_train,color='violet')
plt.plot(x_train,regressor.predict(x_train),color='black')
plt.title("Hours vs Scores(Training set)")
plt.xlabel("Hours")
plt.ylabel("Scores")
plt.show()

plt.scatter(x_test,y_test,color='black')
plt.plot(x_train,regressor.predict(x_train),color='green')
plt.title("Hours vs Scores(Testing set)")
plt.xlabel("Hours")
plt.ylabel("Scores")
plt.show()
mse=mean_absolute_error(y_test,y_pred)
print('MSE = ',mse)
mae=mean_absolute_error(y_test,y_pred)
print('MAE = ',mae)
rmse=np.sqrt(mse)
```

```
print("RMSE= ",rmse)
/*
```

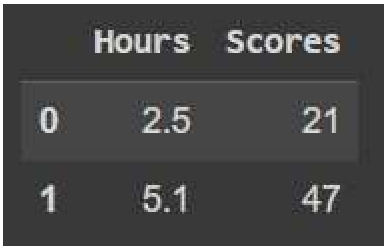
Output:

DATA SET



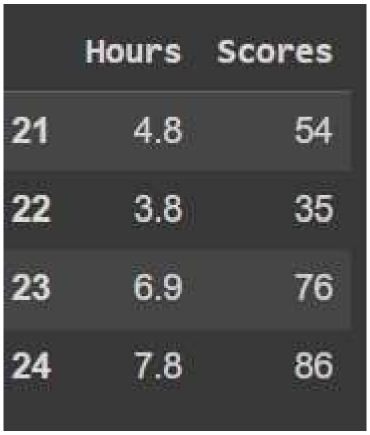
```
    Hours  Scores
0     2.5      21
1     5.1      47
2     3.2      27
3     8.5      75
4     3.5      30
5     1.5      20
6     9.2      88
7     5.5      60
8     8.3      81
9     2.7      25
10    7.7      85
11    5.9      62
12    4.5      41
13    3.3      42
14    1.1      17
15    8.9      95
16    2.5      30
17    1.9      24
18    6.1      67
19    7.4      69
20    2.7      30
21    4.8      54
22    3.8      35
23    6.9      76
24    7.8      86
```

HEAD VALUES



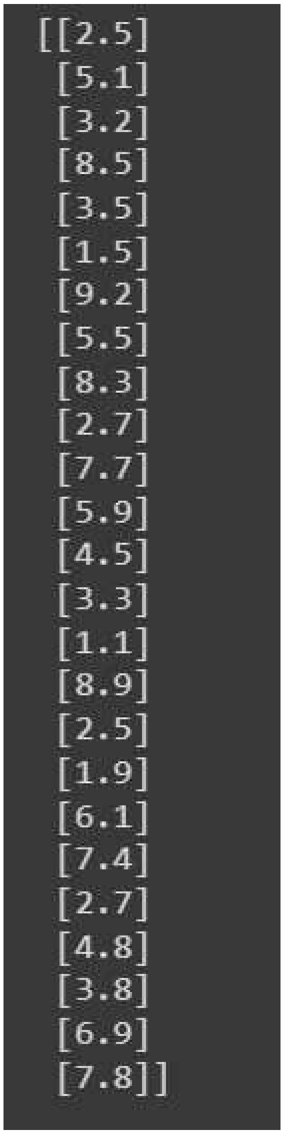
	Hours	Scores
0	2.5	21
1	5.1	47

TAIL VALUES



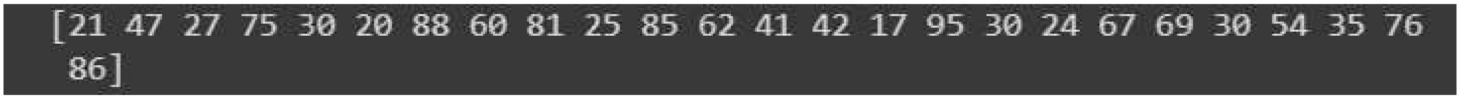
	Hours	Scores
21	4.8	54
22	3.8	35
23	6.9	76
24	7.8	86

X VALUES



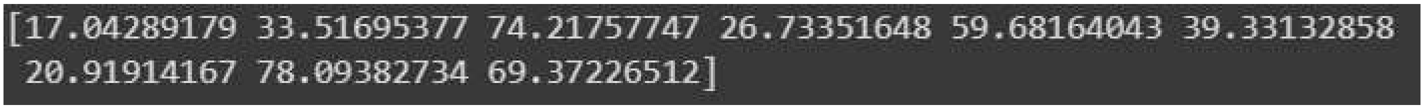
```
[[2.5]
 [5.1]
 [3.2]
 [8.5]
 [3.5]
 [1.5]
 [9.2]
 [5.5]
 [8.3]
 [2.7]
 [7.7]
 [5.9]
 [4.5]
 [3.3]
 [1.1]
 [8.9]
 [2.5]
 [1.9]
 [6.1]
 [7.4]
 [2.7]
 [4.8]
 [3.8]
 [6.9]
 [7.8]]
```

Y VALUES

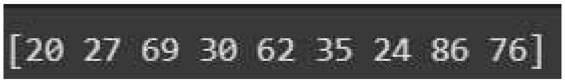


```
[21 47 27 75 30 20 88 60 81 25 85 62 41 42 17 95 30 24 67 69 30 54 35 76
 86]
```

PREDICTION VALUES

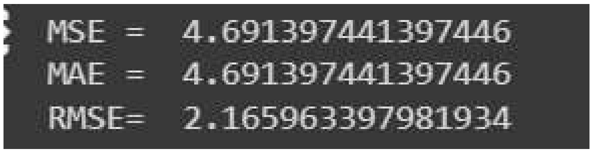


```
[17.04289179 33.51695377 74.21757747 26.73351648 59.68164043 39.33132858
 20.91914167 78.09382734 69.37226512]
```



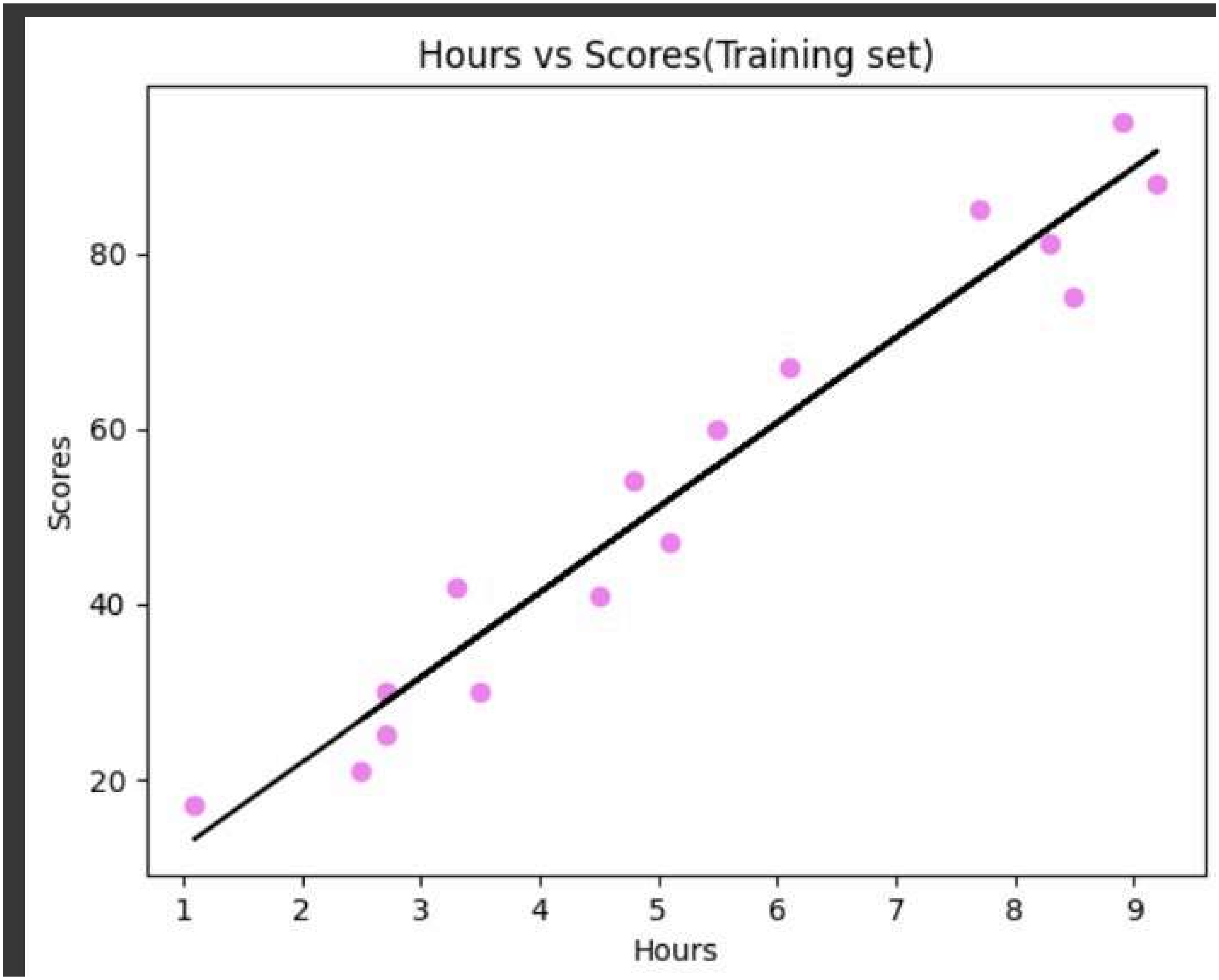
```
[20 27 69 30 62 35 24 86 76]
```

MSE,MAE and RMSE

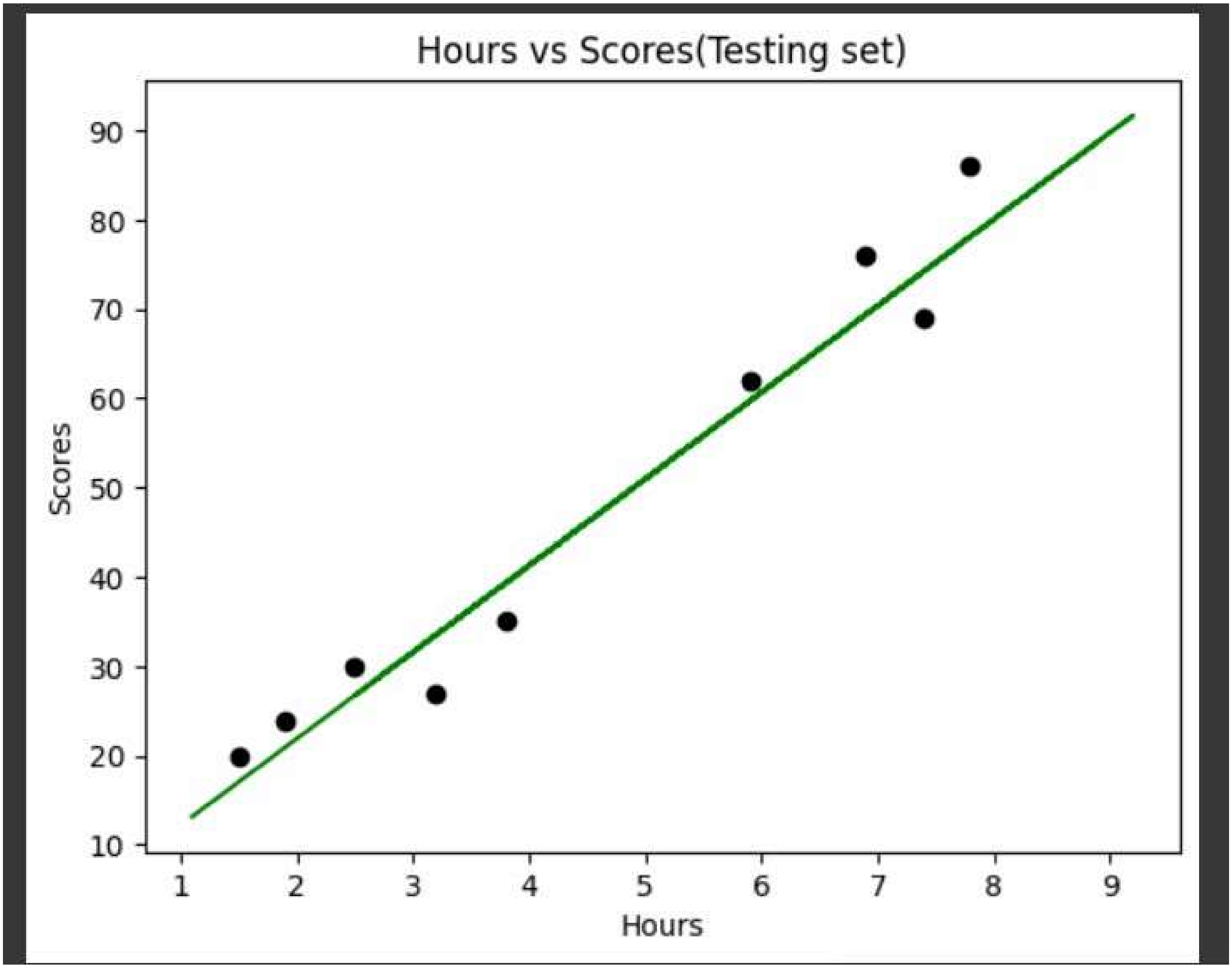


```
MSE =  4.691397441397446
MAE =  4.691397441397446
RMSE=  2.165963397981934
```

TRAINING SET



TESTING TEST



Result:

Thus the program to implement the simple linear regression model for predicting the marks scored is written and verified using python programming.