In [1]:
```
#import libraries

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

In [3]:
```
#mporting Dataset
df = pd.read_csv(r"C:\Users\Akankasha\OneDrive\Desktop\ds\Hours and Scores.csv")
```

In [4]:
```
#Checking and visualising data
df.head()
```

Out[4]:

	Unnamed: 0	Hours	Scores
0	2.5,21	2.5	21
1	5.1,47	5.1	47
2	3.2,27	3.2	27
3	8.5,75	8.5	75
4	3.5,30	3.5	30

In [5]:
```
#Checking for Null values
df.isnull()
```

Out[5]:

	Unnamed: 0	Hours	Scores
0	False	False	False
1	False	False	False
2	False	False	False
3	False	False	False
4	False	False	False
5	False	False	False
6	False	False	False
7	False	False	False
8	False	False	False
9	False	False	False
10	False	False	False
11	False	False	False
12	False	False	False
13	False	False	False
14	False	False	False
15	False	False	False
16	False	False	False
17	False	False	False
18	False	False	False
19	False	False	False
20	False	False	False
21	False	False	False
22	False	False	False
23	False	False	False
24	False	False	False

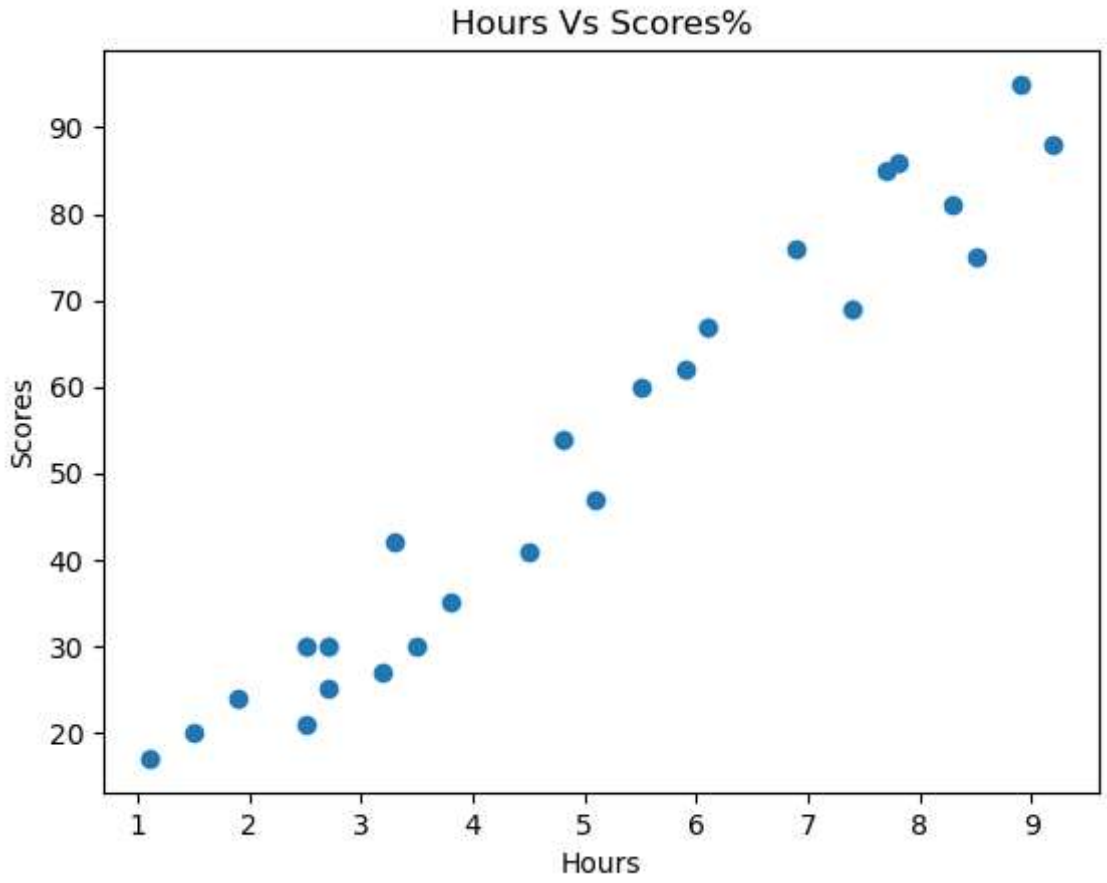
In [6]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25 entries, 0 to 24
Data columns (total 3 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Unnamed: 0  25 non-null     object
 1   Hours       25 non-null     float64
 2   Scores      25 non-null     int64
dtypes: float64(1), int64(1), object(1)
memory usage: 732.0+ bytes
```

In [7]:
```
#Making a list of columns
columns = list(df.columns)
```

In [8]:
```
X = df["Hours"].values.reshape(-1,1)
Y = df["Scores"].values.reshape(-1,1)
```

In [10]:
```
#Visualising Data
plt.scatter( X , Y , color = "#1f77b4" )
plt.xlabel("Hours")
plt.ylabel("Scores")
plt.title("Hours Vs Scores%")

plt.show()
```



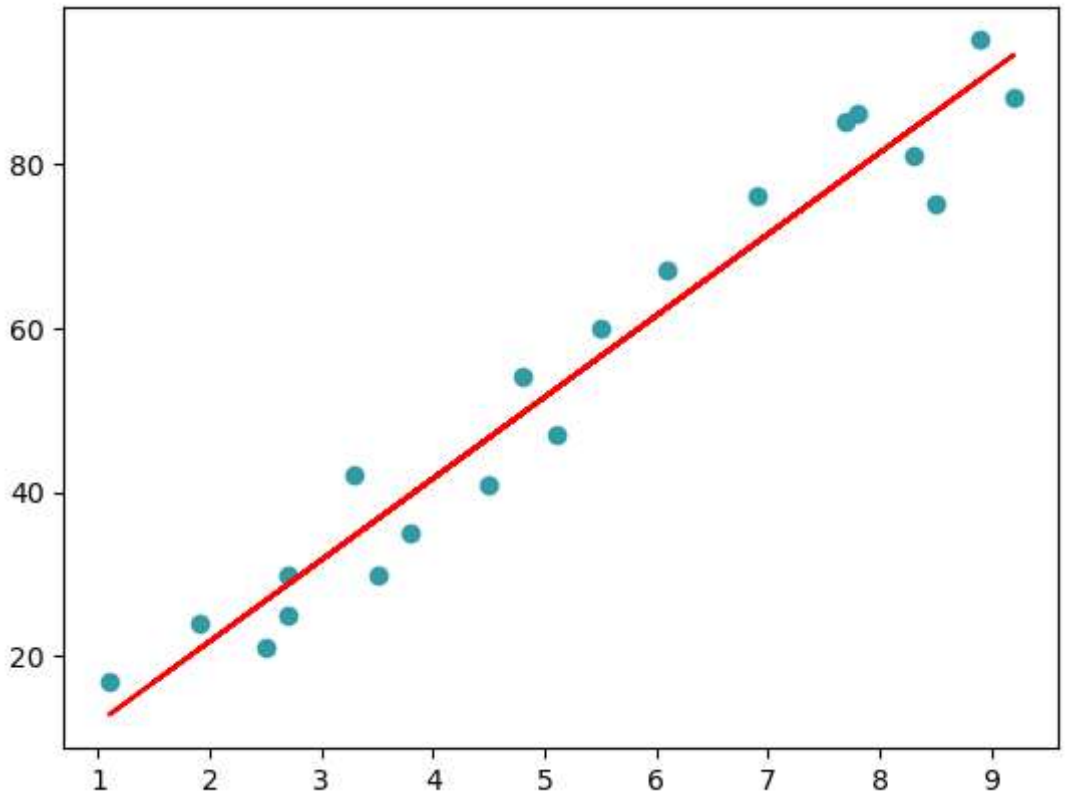
In [11]:
```
#Splitting the data
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=0.20, random_st
```

In [12]:
```
#Let's load the modules for linear regression:
from sklearn.linear_model import LinearRegression
lr = LinearRegression()
lr
```

Out[12]:
```
▾ LinearRegression

LinearRegression()
```

In [15]:
```
#train the model
lr.fit(x_train,y_train)
line = lr.coef_ * X + lr.intercept_
```

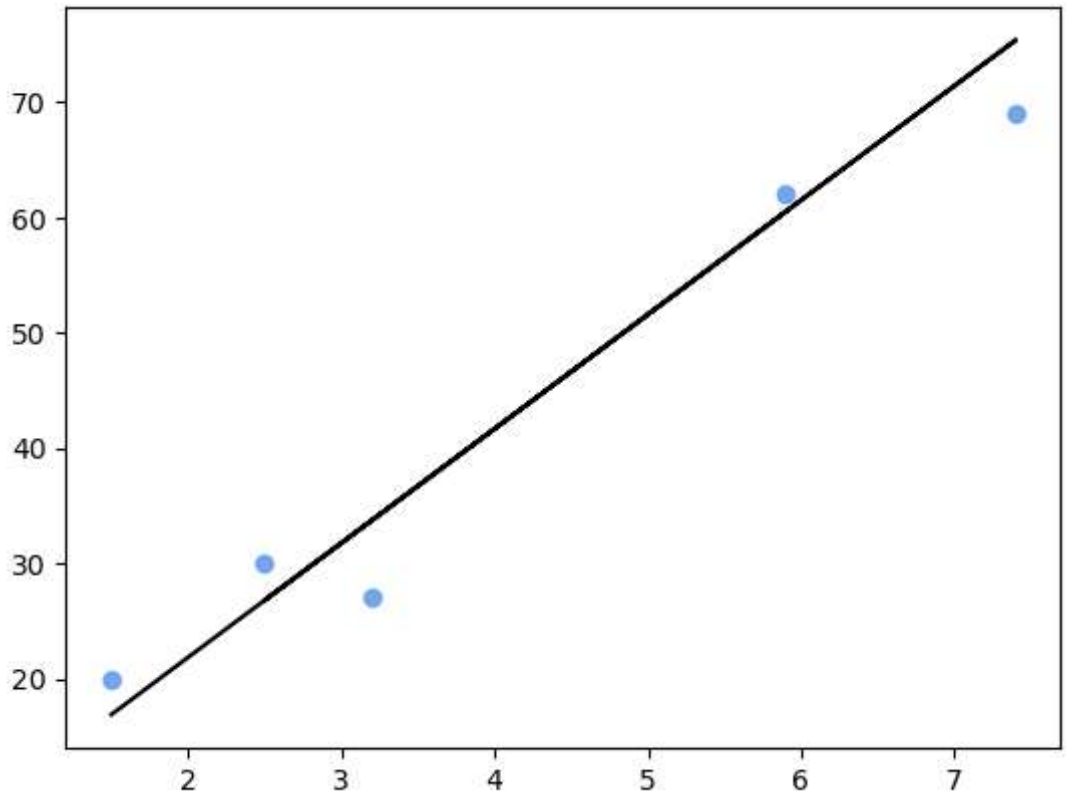
In [16]:
```
#Visualising Train Data
plt.scatter(x_train , y_train , color = "#329ba8")
plt.plot(X , line , color = "r")
plt.show()
```



In [17]:
```
#Making Predictions
Y_pred = lr.predict(x_test)
Y_pred
```

Out[17]:
```
array([[16.88414476],
       [33.73226078],
       [75.357018  ],
       [26.79480124],
       [60.49103328]])
```

In [18]:
```
#Visualising Test Data
plt.scatter(x_test,y_test , color = "#75a6eb")
plt.plot(x_test,Y_pred , color = "black")
plt.show()
```



In [19]:
```
df_predict = pd.DataFrame({"Hours": x_test.reshape(1,-1)[0] , "Actual Score" : y_te
df_predict
```

Out[19]:

	Hours	Actual Score	Predicted Score
0	1.5	20	16.884145
1	3.2	27	33.732261
2	7.4	69	75.357018
3	2.5	30	26.794801
4	5.9	62	60.491033

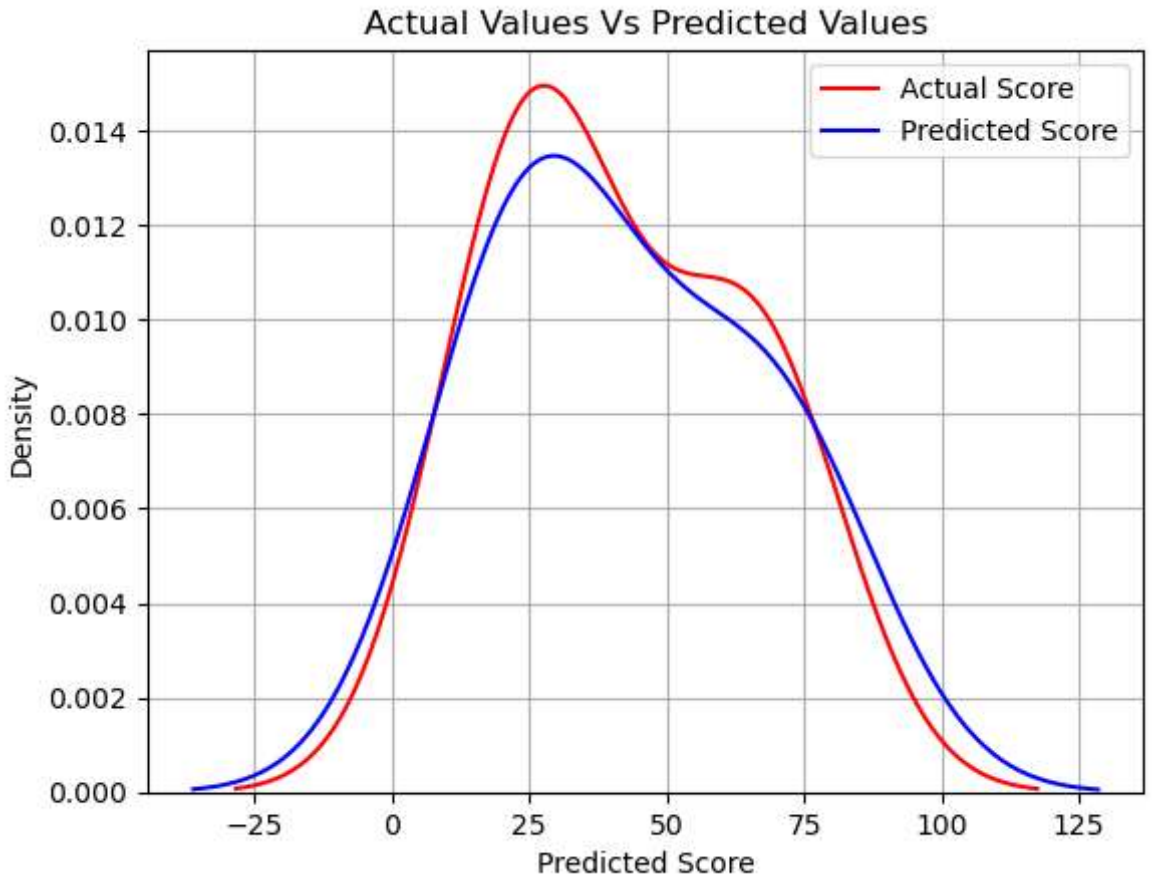
In [20]:
```
#Visualising the accuracy of the model
df_sorted = df_predict.sort_values(by = "Hours")
df_sorted
```

Out[20]:

	Hours	Actual Score	Predicted Score
0	1.5	20	16.884145
3	2.5	30	26.794801
1	3.2	27	33.732261
4	5.9	62	60.491033
2	7.4	69	75.357018

In [21]:
```
title = "Actual Values Vs Predicted Values"
ax1 = sns.distplot(df_sorted["Actual Score"], hist = False , color = "red" , label
sns.distplot(df_sorted["Predicted Score"] , hist = False , color = "blue" , label =
plt.legend()
plt.grid()
plt.title(title)
plt.show()
```

```
C:\Users\Akankasha\AppData\Local\Temp\ipykernel_2572\1582711396.py:2: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `kdeplot` (an axes-level function for kernel density plot
s).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  ax1 = sns.distplot(df_sorted["Actual Score"], hist = False , color = "red" , lab
el = "Actual Score")
C:\Users\Akankasha\AppData\Local\Temp\ipykernel_2572\1582711396.py:3: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `kdeplot` (an axes-level function for kernel density plot
s).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df_sorted["Predicted Score"] , hist = False , color = "blue" , labe
l = "Predicted Score" , ax = ax1)
```



In [22]:
```python
from sklearn.metrics import r2_score
from sklearn import metrics

mean_absolute_error=metrics.mean_absolute_error(y_test,Y_pred)
print('Mean absolute error:',mean_absolute_error)

corr=r2_score(y_train,lr.predict(x_train))
print('correlation:',corr)
```

```
acc=r2_score(y_test,Y_pred)
print('Accuracy:',acc)
```

```
Mean absolute error: 4.183859899002975
correlation: 0.9515510725211552
Accuracy: 0.9454906892105356
```

In [23]:
```
#Making Predictions
hrs = 9.25
pred = lr.predict([[9.25]])
print("The predicted score if a student studies for 9.25 hrs/ day is",pred[0])
```

```
The predicted score if a student studies for 9.25 hrs/ day is [93.69173249]
```

In [ ]:
```
SSS
```