In [20]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

In [21]:
```
path = "http://bit.ly/w-data"
dataset = pd.read_csv(path)
dataset.head()
```

Out[21]:

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

In [22]:
```
dataset.describe()
```

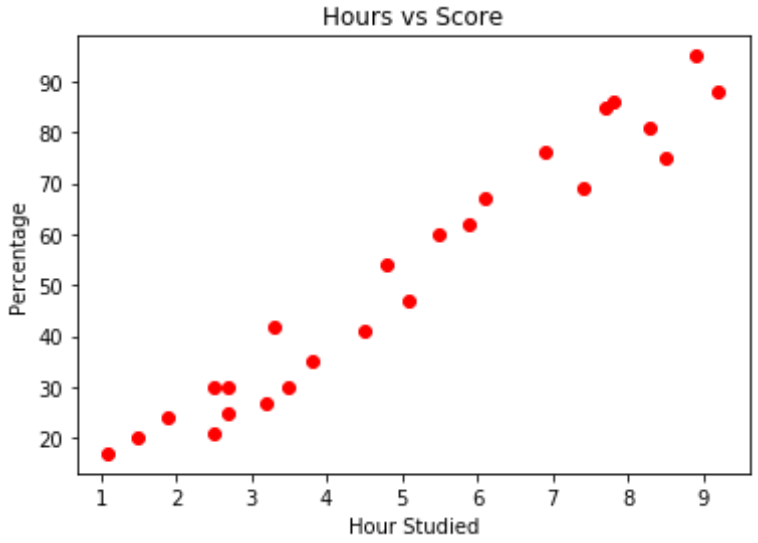
Out[22]:

	Hours	Scores
count	25.000000	25.000000
mean	5.012000	51.480000
std	2.525094	25.286887
min	1.100000	17.000000
25%	2.700000	30.000000
50%	4.800000	47.000000
75%	7.400000	75.000000

	Hours	Scores
max	9.200000	95.000000

In [23]:
```
x = dataset.iloc[:, :-1].values
y = dataset.iloc[:, 1].values
```

In [24]:
```
plt.scatter(x,y,color = 'red')
plt.title("Hours vs Score")
plt.xlabel("Hour Studied")
plt.ylabel('Percentage')
plt.show()
```



In [25]:
```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2
, random_state=0)
```

In [26]:
```
from sklearn.linear_model import LinearRegression
regressor_train = LinearRegression()
regressor_train.fit(x_train, y_train)
```

Out[26]:
```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normaliz
e=False)
```

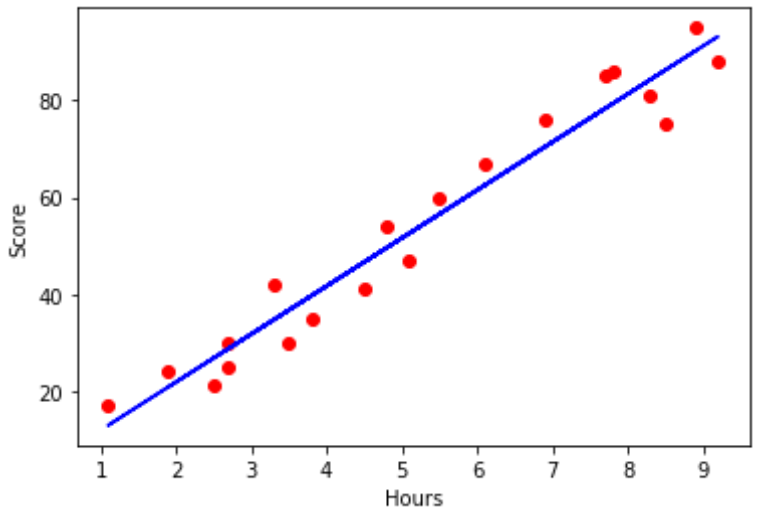
In [27]:
```
y_pred = regressor_train.predict(x_test)
df = pd.DataFrame({'Actual Score': y_test, 'Predicted Score': y_pred})

df
```

Out[27]:

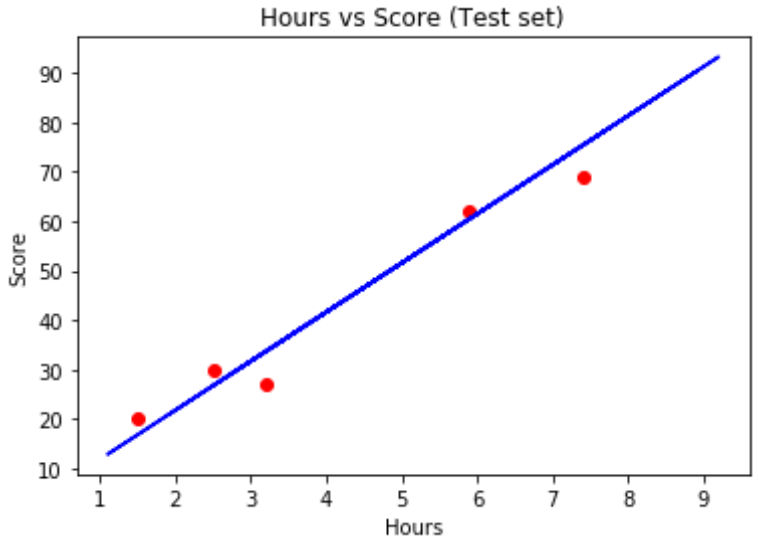
	Actual Score	Predicted Score
0	20	16.884145
1	27	33.732261
2	69	75.357018
3	30	26.794801
4	62	60.491033

In [28]:
```
plt.scatter(x_train, y_train, color='red')
plt.plot(x_train, regressor_train.predict(x_train), color='blue')
plt.xlabel("Hours")
plt.ylabel("Score")
plt.show()
```



In [29]:
```
plt.scatter(x_test, y_test, color = 'red')
plt.plot(x_train, regressor_train.predict(x_train), color = 'blue')
plt.title('Hours vs Score (Test set)')
plt.xlabel('Hours')
plt.ylabel('Score')
```

Out[29]:
```
Text(0, 0.5, 'Score')
```



In [31]:
```
no_of_hours = 9.25
predictor = regressor_train.predict(np.array([9.25]).reshape(1, 1))
print("No of Hours = {}".format(no_of_hours))
print("Predicted Score = {}".format(predictor[0]))
plt.show()
```

```
No of Hours = 9.25
Predicted Score = 93.69173248737538
```

In [32]:
```
from sklearn import metrics
print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, y_pre
d))
```

```
Mean Absolute Error: 4.183859899002975
```

In [ ]:

In [ ]:

In [ ]: