THE SPARKS FOUNDATION

DataScience and Business Analysis

Task1 = Pridection Using Supervised Machine Learning

To Predict the percentage of marks that a student is expected to score based upon the number of hours they studied.

Importing Libraties

In [18]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

Importing DataSet

In [19]:
```
url = "http://bit.ly/w-data"
dataset = pd.read_csv(url)
print("Data imported successfully")
```

```
Data imported successfully
```

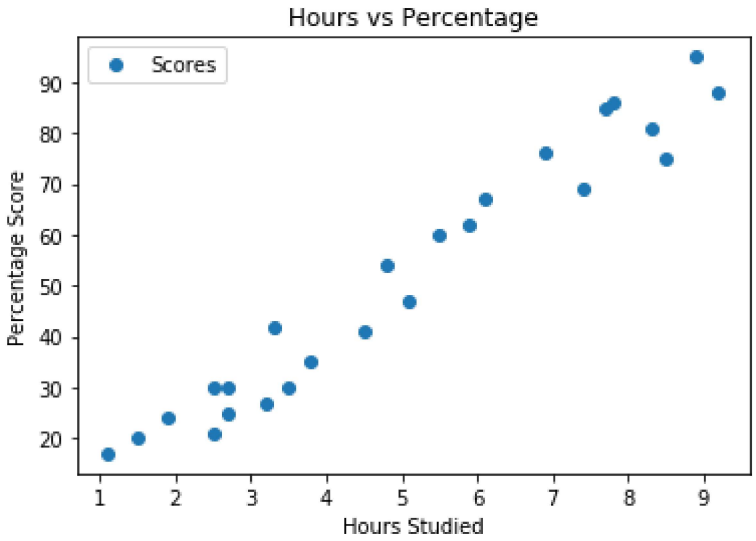
In [20]:
```
dataset.head(10)
```

Out[20]:

	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

Data Visualization

In [21]:
```
dataset.plot(x='Hours', y='Scores', style='o')
plt.title('Hours vs Percentage')
plt.xlabel('Hours Studied')
plt.ylabel('Percentage Score')
plt.show()
```



Preparing The Data

In [22]:
```
X=dataset.iloc[:, :1].values
Y=dataset.iloc[:, 1].values
```

Now we have our attributes and labels, now split the data into training test sets. no we will do this by using Scikit-learn's buit-in train_test_split() method.

In [23]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size = 0.
```

Training The Algorithm

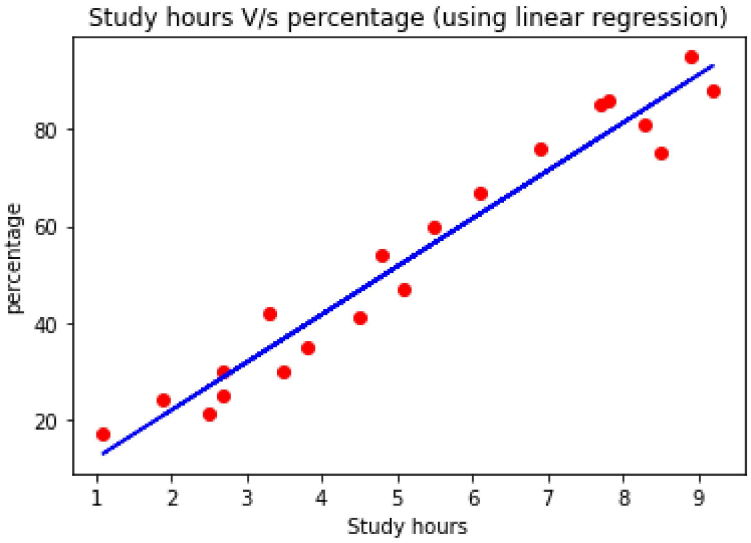
We have to split our data into training and testing sets, and now is finally the time to train our algorithm

In [24]:
```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)

y_pred = regressor.predict(X_test)
```

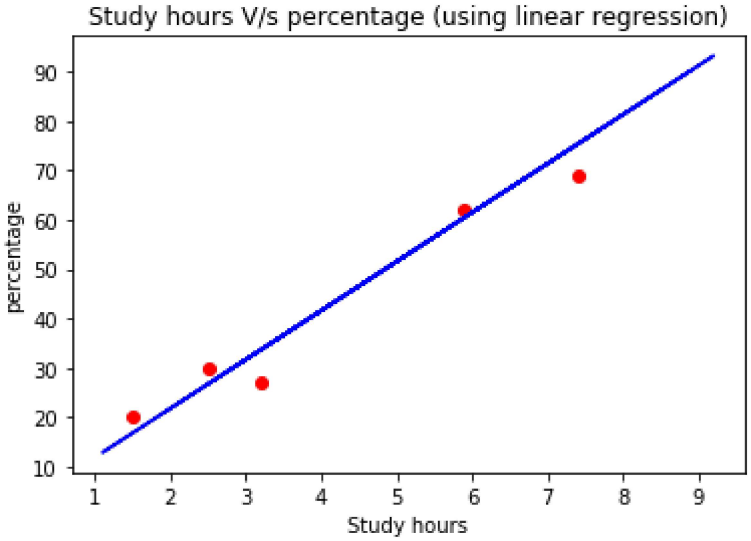
Visualising the Training set results

In [25]:
```
plt.scatter(X_train, y_train, color = 'red')
plt.plot(X_train, regressor.predict(X_train), color = 'blue')
plt.title("Study hours V/s percentage (using linear regression)")
plt.xlabel("Study hours")
plt.ylabel("percentage")
plt.show()
```



Visualising the Test set results

In [26]:
```
plt.scatter(X_test, y_test, color = 'red')
plt.plot(X_train, regressor.predict(X_train), color = 'blue')
plt.title("Study hours V/s percentage (using linear regression)")
plt.xlabel("Study hours")
plt.ylabel("percentage")
plt.show()
```



In [27]:

```
1 print('Thank YOU')
```

```
Thank YOU
```