In [1]:

```
# importing all the necessary libraries

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:

```
# Loading of Data

url = "http://bit.ly/w-data"
data = pd.read_csv("http://bit.ly/w-data")
data
```

Out[2]:

	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

Visualization of Data

In [5]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25 entries, 0 to 24
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   Hours   25 non-null     float64
 1   Scores  25 non-null     int64  
dtypes: float64(1), int64(1)
memory usage: 528.0 bytes
```

In [6]:

```
data.describe()
```

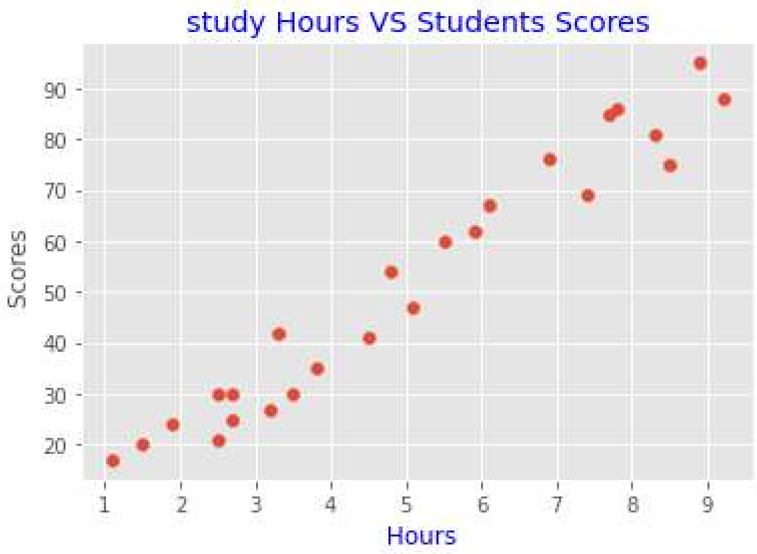
Out[6]:

	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
max	9.200000	95.000000

In [12]:

```
# scatter plot graph between hours of study and score

plt.scatter(x=data.Hours, y=data.Scores)
plt.xlabel(" Hours")
plt.ylabel("Scores ")
plt.title("study Hours VS Students Scores",color="blue")
plt.show()
```



In [9]:

```
# shape of data
data.shape
```

Out[9]:

```
(25, 2)
```

Prepare the Data for Machine Learning Algorithm

In [13]:

```
# checking for any null values
data.isnull().sum()
```

Out[13]:

```
Hours     0
Scores    0
dtype: int64
```

In [14]:

```
data.mean()
```

Out[14]:

```
Hours      5.012
Scores    51.480
dtype: float64
```

In [16]:

```
# fill null values with mean  (if any null values)
data2=data.fillna(data.mean())
data2.isnull().sum()
```

Out[16]:

```
Hours     0
Scores    0
dtype: int64
```

In [20]:

```
data2.head(5)
```

Out[20]:

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

In [21]:

```
data2.tail(5)
```

Out[21]:

	Hours	Scores
20	2.7	30
21	4.8	54
22	3.8	35
23	6.9	76
24	7.8	86

In [22]:

```
# split dataset for training

x=data2.drop("Scores",axis="columns")
y=data2.drop("Hours",axis="columns")
print("shape of x",x.shape)
print("shape of y",y.shape)
```

```
shape of x (25, 1)
shape of y (25, 1)
```

In [23]:

```
# importing sklearn model
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_sta
# defining test_size

print("shape of X Train ", X_train.shape)
print("shape of Y Train ", Y_train.shape)
print("shape of X Test ", X_test.shape)
print("shape of Y Test ", Y_test.shape)
```

```
shape of X Train  (20, 1)
shape of Y Train  (20, 1)
shape of X Test  (5, 1)
shape of Y Test  (5, 1)
```

Select a model and train it

In [24]:

```
# Linear Regression model
from sklearn.linear_model import LinearRegression
lr= LinearRegression()

lr.fit(X_train,Y_train)
```

Out[24]:

```
LinearRegression()
```

In [27]:

```
# coeffecient
lr.coef_
```

Out[27]:

```
array([[10.46110829]])
```

In [28]:

```
# intercept
lr.intercept_
```

Out[28]:

```
array([-1.53695733])
```

In [29]:

```
lr.predict([[9.25]])[0][0].round(2)
```

Out[29]:

```
95.23
```

In [30]:

```
y_pred = lr.predict(X_test)
y_pred
```

Out[30]:

```
array([[ 9.97026179],
       [32.98470004],
       [18.33914843],
       [87.38246316],
       [48.67636248]])
```

```
In [31]: # Score of my model

         lr.score(x,y)*100

Out[31]: 94.77059220582453
```

# THANK YOU