

Author: Harshavardhan Madhuranthakam

## GRIP @ The Sparks Foundation

In this regression task I have predicted the percentage of marks that a student is expected to score based upon the number of hours they studied. This is a simple linear regression problem as it has just one predictor.

### Step:1 Importing Useful Python Library

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

### Step:2 Importing Dataset

```
In [2]: data=pd.read_csv("data.csv")
print('Importing Data Successfully')
```

Importing Data Successfully

```
In [3]: print('First ten data') data.head(10)
```

First ten data

	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

Preparing Data for Machine learning

```
In [25]: 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 [4]: #Data cleaning
data.isnull().sum()
```

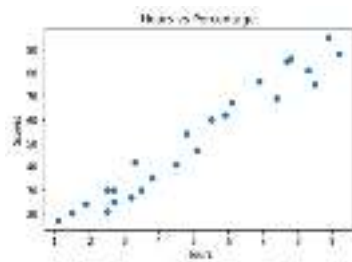
```
Out[4]: Hours    0
Scores    0
dtype: int64
```

### Step:3 Data Visualisation

```
In [5]: x=np.array(data[['Hours']])
```

```
In [6]: y=np.array(data[['Scores']])
```

```
In [7]: plt.scatter(x,y)
plt.title("Hours vs Percentage") plt.xlabel("Hours")
plt.ylabel("Scores") plt.show()
```



```
In [8]: print('We can see that Scores increases as the no. of hours studied is increase')
print('hence we can conclude that there exist a positive linear relation between the number of hours studied and percentage of score.')
```

We can see that Scores increases as the no. of hours studied is increase  
hence we can conclude that there exist a positive linear relation between the number of hours studied and percentage of score.

### Step:4 Train-Test-Split

```
In [9]: x=data[['Hours']]
y=data[['Scores']]

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)
```

### Step:5 Training Algorithm

```
In [10]: from sklearn.linear_model import LinearRegression regressor=LinearRegression()

regressor.fit(x_train, y_train) print('Training
```