# GRIP : The Sparks Foundation

# Data Science and Buisness Analytics Intern

# Author : Sweta Patel

# Task 1 : Prediction Using Supervised ML

Aim = To predict the percentage of student based on the number of study hours. This is simple linear regression task as it involves just two variables.

In [10]:
```
# Importing all libraries.
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [16]:
```
# Reading data from URL
url="http://bit.ly/w-data"
df=pd.read_csv(url)
```

In [20]:
```
#Exploring Data
print(df.shape)
df.head()
```

```
(25, 2)
```

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 [22]:
```
df.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 |
| **max** | 9.200000 | 95.000000 |

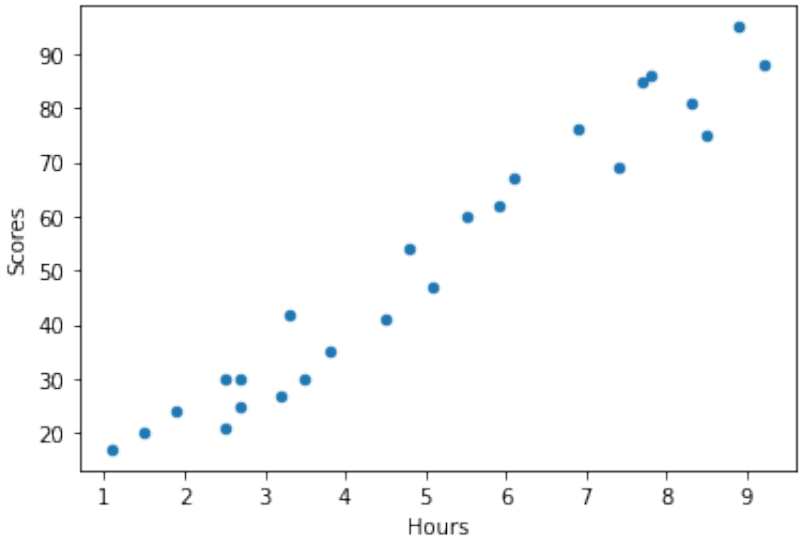
In [23]:

```
df.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 [24]:

```
df.plot(kind='scatter',x='Hours',y='Scores');
plt.show()
```




In [27]:

```
#Preparing the data
X = df.iloc[:, :-1].values
y = df.iloc[:, 1].values
```

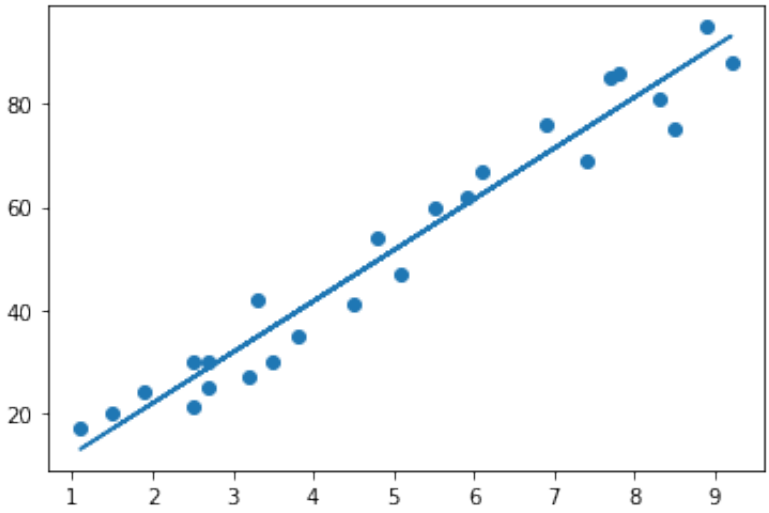
In [28]:
```
#train_test_split()method
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y,
                                test_size=0.2, random_state=0)
```

## Linear Regression

In [29]:
```
#We have split our data
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)
```

Out[29]:
```
LinearRegression()
```

In [30]:
```
#Plotting Regeression and Data Set
line = regressor.coef_*X+regressor.intercept_
plt.scatter(X, y)
plt.plot(X, line);
plt.show()
```




## Predictions

In [32]:
```
print(X_test)
y_pred = regressor.predict(X_test)
```

```
[[1.5]
 [3.2]
 [7.4]
 [2.5]
 [5.9]]
```

In [33]:
```
df = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred})
df
```

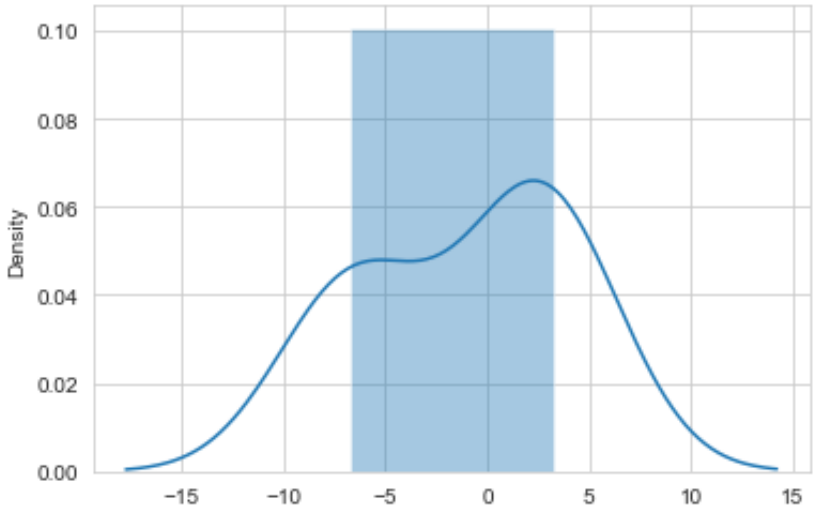
Out[33]:

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

In [35]:

```
sns.set_style('whitegrid')
sns.distplot(np.array(y_test-y_pred))
plt.show()
```

```
/Users/apple/opt/anaconda3/lib/python3.8/site-packages/seaborn/distributions.py:2557: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```




If a student studies for 9.25 hours/day what would be the predicted score?

In [41]:

```
h = 9.25
df = regressor.predict([[hours]])
print("No of Hours = {}".format(hours))
print("Predicted Score = {}".format(df[0]))
```

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

Modal Evaluation

In [44]:

```
from sklearn import metrics
print('Mean Absolute Error:',
      metrics.mean_absolute_error(y_test, y_pred))
```

```
Mean Absolute Error: 4.183859899002975
```

In [45]:

```
from sklearn.metrics import r2_score
print('R2 Score:', r2_score(y_test,y_pred))
```

```
R2 Score: 0.9454906892105356
```