# # GRIP - Internship, November 2022 Group

TASK 1 - SUPERVISED MACHINE LEARNING

NAME: MAANDA MULAUDZI

In [2]:

```
import pandas as pd
```

In [3]:

```
dataset = pd.read_csv('https://raw.githubusercontent.com/AdiPersonalWorks/Random/master/stu
```

In [4]:

```
dataset.head()
```

Out[4]:

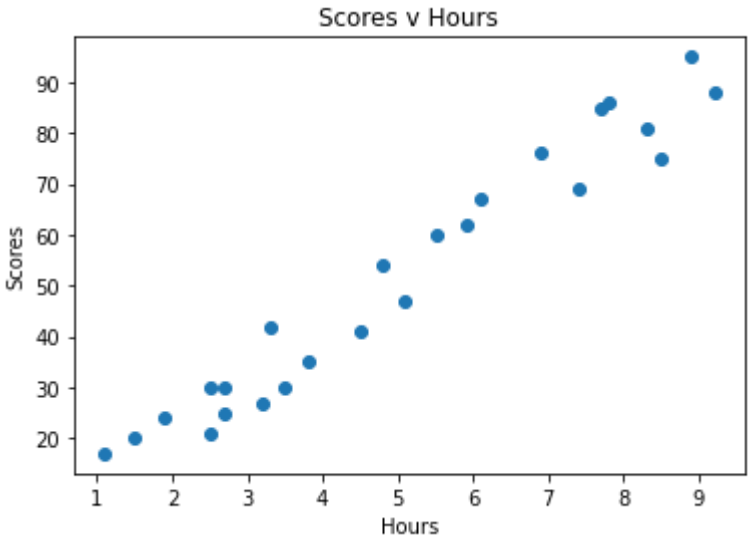
| | Hours | Scores |
|---|---|---|
| **0** | 2.5 | 21 |
| **1** | 5.1 | 47 |
| **2** | 3.2 | 27 |
| **3** | 8.5 | 75 |
| **4** | 3.5 | 30 |

In [5]:

```
import matplotlib.pyplot as plt
```

In [6]:

```
plt.scatter (x = dataset['Hours'], y = dataset['Scores'])
plt.xlabel ('Hours')
plt.ylabel ('Scores')
plt.title ('Scores v Hours')
plt.show()
```




In [7]:

```
X = pd.DataFrame(dataset['Hours'])
y = pd.DataFrame(dataset['Scores'])
```

In [8]:

```
print (X.head())
print (y.head())
```

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

In [9]:

```
from sklearn.model_selection import train_test_split
```

In [10]:

```
X_train, X_test, y_train, y_test = train_test_split(
                                        X, y, test_size = 0.2, random_state = 1
)
```

In [11]:

```
print (X_train.shape)
print (y_train.shape)
print (X_test.shape)
print (y_test.shape)
```

```
(20, 1)
(20, 1)
(5, 1)
(5, 1)
```

In [12]:

```
from sklearn.linear_model import LinearRegression
```

In [13]:

```
model = LinearRegression()
model.fit(X_train, y_train)
```

Out[13]:

```
LinearRegression()
```

In [14]:

```
print(model.intercept_)
```

```
[-1.53695733]
```

In [15]:

```
print(model.coef_)
```

```
[[10.46110829]]
```

In [16]:

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

Out[16]:

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

In [17]:

```
y_test
```

Out[17]:

| | Scores |
|---|---|
| **14** | 17 |
| **13** | 42 |
| **17** | 24 |
| **3** | 75 |
| **21** | 54 |

In [18]:

```
model.score(X_test, y_test)
```

Out[18]:

```
0.8421031525243527
```

In [ ]:

```
# The model is 84.21% accurate on predicting scores based on hours studied.
```

In [ ]:

```
# Model Evaluation
```

In [19]:

```
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error
import numpy as np
```

In [20]:

```
print('Mean Absolute Error:', mean_absolute_error(y_test, y_pred))
print('Mean Squared Error:', mean_squared_error(y_test, y_pred))
print('Root Mean Squared Error:', np.sqrt(mean_squared_error(y_test, y_pred)))
```

```
Mean Absolute Error: 7.882398086270432
Mean Squared Error: 68.88092074277635
Root Mean Squared Error: 8.299453038771674
```

```
MAE shows that our prediction is approximately 8 units on average from true value.
```