The Sparks Foundation- GRIP -Data Science and Business Analytics- JUNE2022

Task 1- Prediction using Supervised ML

Author- Kalyani Gawande

Objective-

1. Predict the percentage of an student based on no. of study hours
2. What will be predicted score if a student studies for 9.25 hrs/day ?

Here there are two variables, where the feature is the no. of hours studied and the target value is percentage score. This can be solved using Simple Linear Regression.

importing libraries

In [1]:

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

Reading and Exploring the Dataset

In [2]:

```
data=pd.read_csv('Task_1_dataset.csv')
print(data.shape)
```

```
(25, 2)
```

In [3]:

```
print("For this, we print first 10 data-points of the dataset: ")
data.head(10)
```

```
For this, we print first 10 data-points of the dataset:
```

Out[3]:

	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

Describing the data

In [4]:

```
data.describe()
```

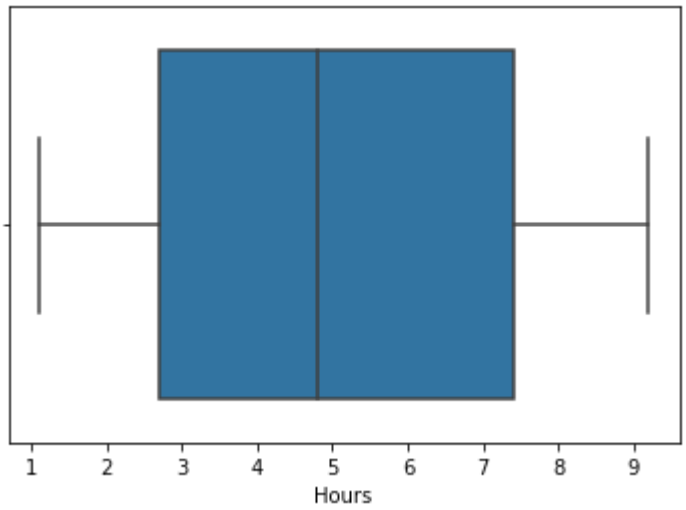
Out[4]:

	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

Visualizing Data

In [5]:

```
sns.boxplot(x=data["Hours"])
plt.show()
```



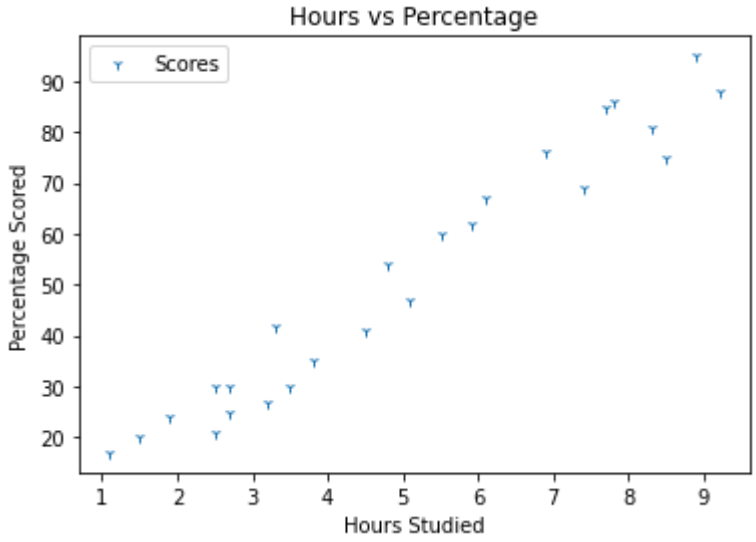
In [6]:

```
print("We successfully and correctlt imported dataset.!")
```

```
We successfully and correctlt imported dataset.!
```

In [7]:

```
data.plot(x='Hours',y='Scores',style='1')
plt.title('Hours vs Percentage')
plt.xlabel('Hours Studied')
plt.ylabel('Percentage Scored')
plt.show()
```

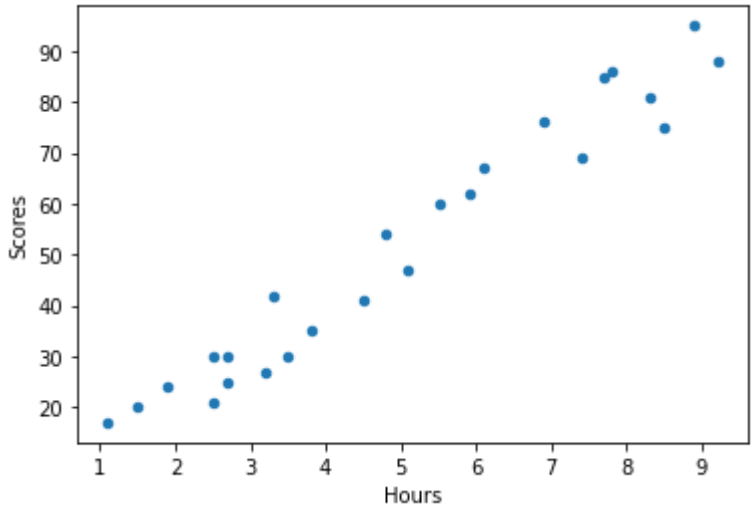


In [8]:

```
data.plot.scatter(x='Hours',y='Scores')
```

Out[8]:

```
<AxesSubplot:xlabel='Hours', ylabel='Scores'>
```



In [9]:

```
data.corr(method='pearson')
```

Out[9]:

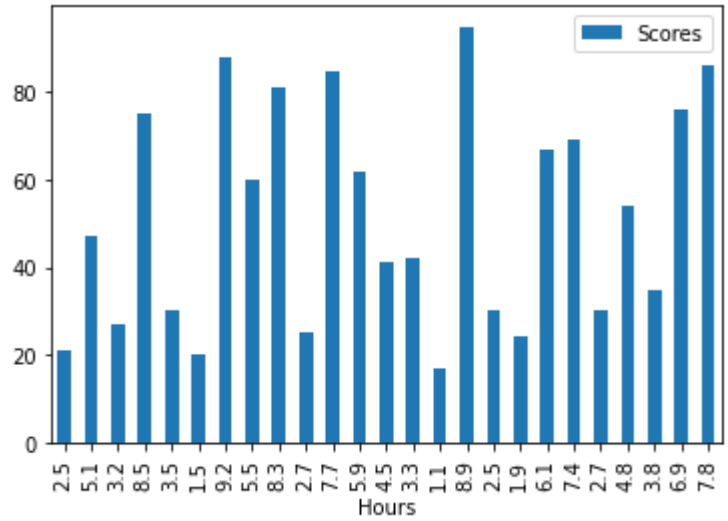
	Hours	Scores
Hours	1.000000	0.976191
Scores	0.976191	1.000000

In [10]:

```
data.plot.bar(x='Hours',y='Scores')
```

Out[10]:

```
<AxesSubplot:xlabel='Hours'>
```



Linear Regression

In [11]:

```
X=data.iloc[:, :-1].values
y=data.iloc[:, 1].values
```

In [12]:

```
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)
```

In [13]:

```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()

regressor.fit(X_train,y_train)
print("Training Completed.")
```

```
Training Completed.
```

In [14]:

```
print(y_test)
print("Prediction of score:")
y_pred=regressor.predict(X_test)
print(y_pred)
```

```
[20 27 69 30 62]
Prediction of score:
[16.88414476 33.73226078 75.357018   26.79480124 60.49103328]
```

In [15]:

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

Out[15]:

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

What will be the predicted score if a student studies for 9.25hrs/day ?

In [16]:

```
hours=[[9.25]]
pred=regressor.predict(hours)
print("If the student studies for {}hrs/day then it is predicted that the student will scor
```

```
If the student studies for [[9.25]]hrs/day then it is predicted that the stu
dent will score [93.69173249] % .
```

Model Evaluation

In [17]:

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

```
Mean Absolute Error: 4.183859899002975
```