SPARKS FOUNDATION

GRIP JUNE 2022

NAME:DHARANI GUNTUPALLI

DATA SCIENCE AND BUSINESS ANAYLTICS INTERN

TASK 1:PREDCITON USING UNSUPERVISED ML

SIMPLE LINEAR REGRESSION

Prediction of percentage of marks based upon the number of hours. This is a simple linear regression task as it involves two feature.

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib as mpl
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

In [2]:
```
url=r'https://raw.githubusercontent.com/AdiPersonalWorks/Random/master/student_scores%20-%20student_scores.csv'
s=pd.read_csv(url)
```

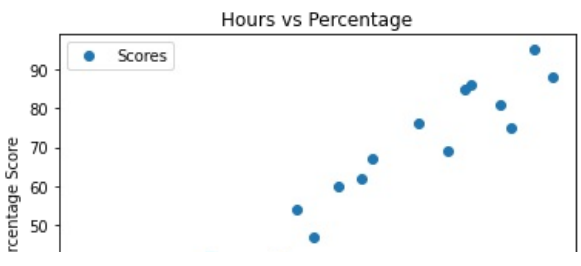
In [3]:
```
s.head(10)
```

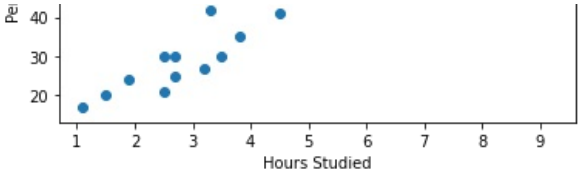
Out[3]:

	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

PLOTTING THE DISTRIBUTION OF SCORES

In [4]:
```
s.plot(x='Hours',y='Scores',style='o')
plt.title('Hours vs Percentage')
plt.xlabel('Hours Studied')
plt.ylabel('Percentage Score')
plt.show()
```





In [5]:
```
X=s.iloc[:,:-1].values
```

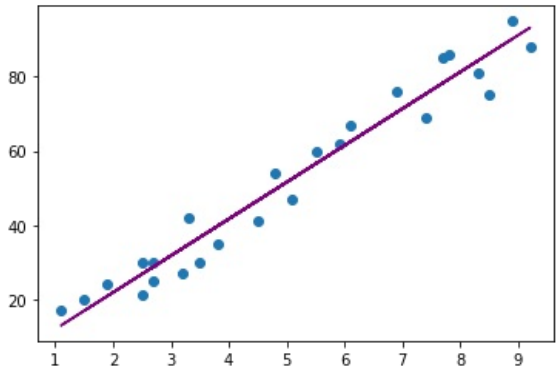
In [6]:
```
y=s.iloc[:,1].values
```

In [7]:
```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=0)
regressor=LinearRegression()
regressor.fit(X_train.reshape(-1,1),y_train)
print('Training complete.')
```

```
Training complete.
```

PLOTTING THE REGRESSION LINE FOR THE TEST DATA

In [8]:
```
line=regressor.coef_*X+regressor.intercept_
plt.scatter(X,y)
plt.plot(X,line,color='purple');
plt.show()
```



TESTING THE DATA AND MODEL PREDICTION

In [9]:
```
print(X_test)
y_pred=regressor.predict(X_test)
```

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

COMPARING ACTUAL VS PREDICTED

In [10]:
```
s=pd.DataFrame({'Actual':y_test,'Predicted':y_pred})
s
```

Out[10]:

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

4	62	60.491033

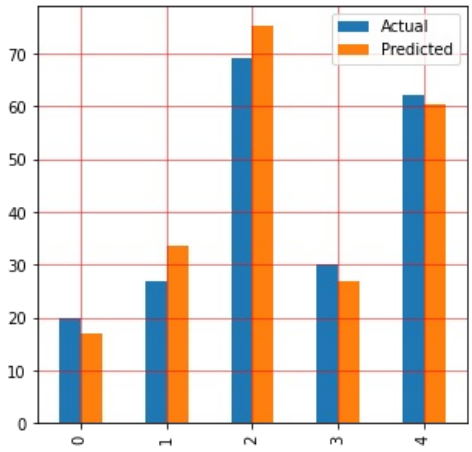
ESTIMATING TRAINING AND TEST SCORE

In [11]:
```
print('Training Score:',regressor.score(X_train,y_train))
print('Test Score:',regressor.score(X_test,y_test))
```

```
Training Score: 0.9515510725211552
Test Score: 0.9454906892105356
```

PLOTTING THE BAR GRAPH TO DEPICT THE DIFFERENCE BETWEEN ACTUAL AND PREDICTED VALUE

In [12]:
```
s.plot(kind='bar',figsize=(5,5))
plt.grid(which='major',linewidth='0.5',color='red')
plt.grid(which='minor',linewidth='0.5',color='blue')
plt.show()
```



TESTING THE MODEL WITH OUR OWN DATA

In [13]:
```
hours=9.25
test=np.array([hours])
test=test.reshape(-1,1)
own_pred=regressor.predict(test)
print('Number of Hours={}'.format(hours))
print('Predicted Score={}'.format(own_pred[0]))
```

```
Number of Hours=9.25
Predicted Score=93.69173248737538
```

In [14]:
```
from sklearn import metrics
print('Mean Absolute Error:',metrics.mean_absolute_error(y_test,y_pred))
```

```
Mean Absolute Error: 4.183859899002975
```

In [ ]:
```

```