Author- Ankita chowdhury

The spark Foundation- Data Science and business Analytics Task

Prediction Using Supervised ML

**TASK- To predict the percentage of students based on the number of study hours by using Linear Regression method with given variables

** STEP 1- Import the required libraries

In [3]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error
```

** STEP 2- Import the csv file from your drive

In [10]:
```
data = pd.read_csv(r"C:\Users\Admin\Desktop\TSF\PROJECT 1- LINEAR REGRESSION\data.csv")
```

**STEP 3- Check if the data set has been uploaded for the task by using following commands

In [11]:
```
data.head(10)
```

Out[11]:

	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

In [12]:
```
data.shape
```

Out[12]:
```
(25, 2)
```

**STEP 4- Check if there's any null values to avoid any errors

In [13]:
```
data.isnull == True
```

Out[13]:
```
False
```

**STEP 5- Familiarize yourself with the data set

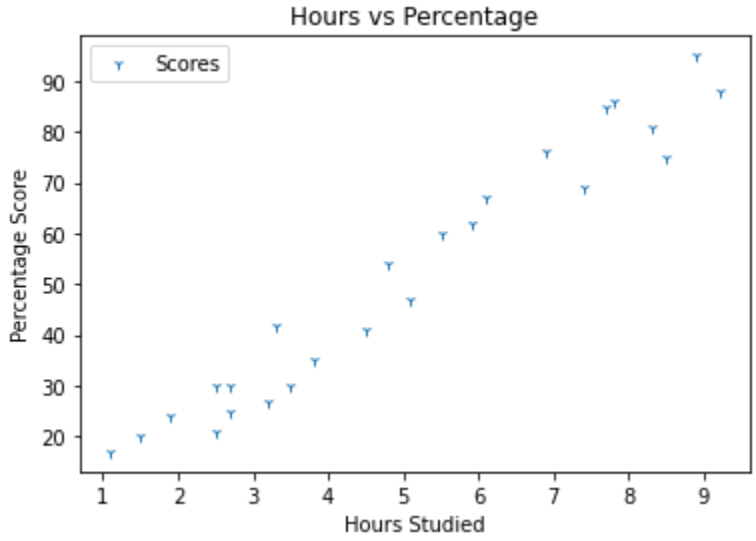
In [14]:
```
data.describe()
```

Out[14]:

	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

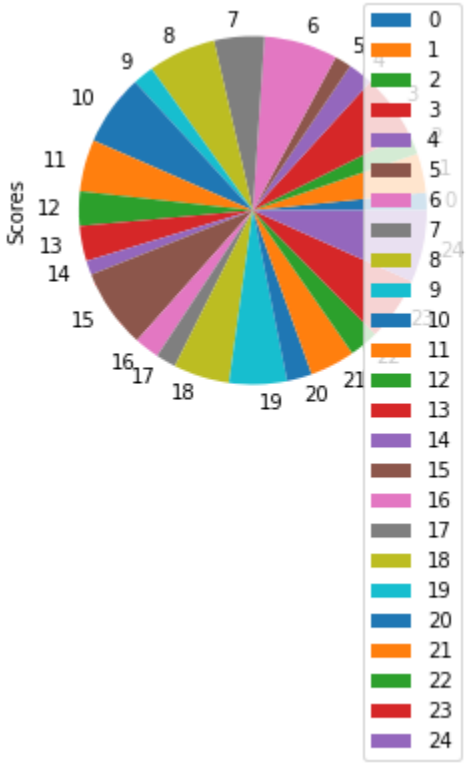
**STEP 6- Plot graphs for deatiled analysis by using the following commands

In [15]:
```
data.plot(x='Hours',y='Scores',style='1')
plt.title('Hours vs Percentage')
plt.xlabel('Hours Studied')
plt.ylabel('Percentage Score')
plt.show()
```



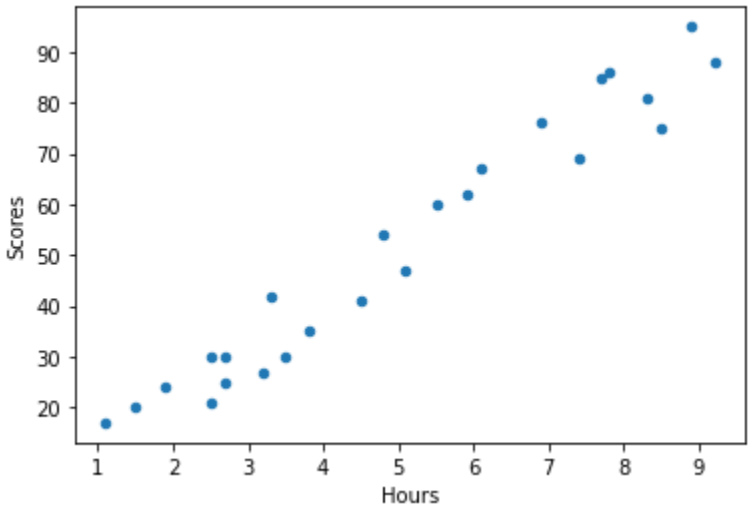
In [16]:
```
data.plot.pie(x='Hours',y='Scores')
```

Out[16]:
```
<AxesSubplot:ylabel='Scores'>
```



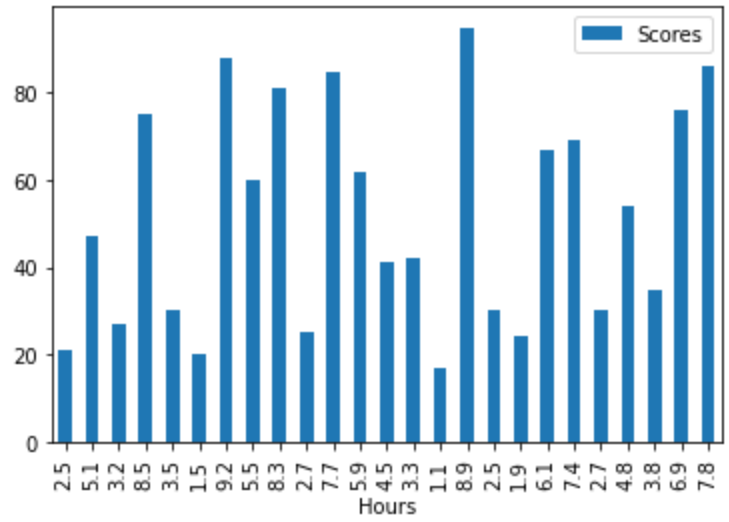
In [17]:
```
data.plot.scatter(x='Hours',y='Scores')
```

Out[17]:
```
<AxesSubplot:xlabel='Hours', ylabel='Scores'>
```



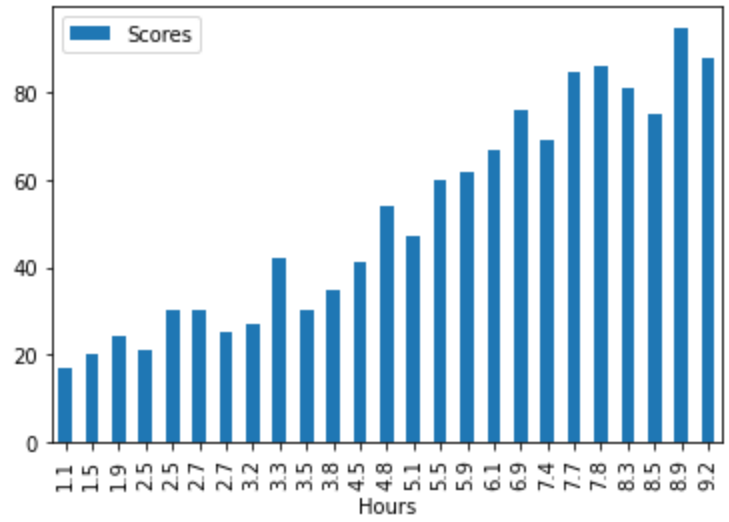
In [18]:
```
data.plot.bar(x='Hours',y='Scores')
```

Out[18]:
```
<AxesSubplot:xlabel='Hours'>
```



In [19]:
```
data.sort_values(["Hours"], axis=0, ascending=[True], inplace=True)
data.head(10)
data.plot.bar(x='Hours', y='Scores')
```

Out[19]:
```
<AxesSubplot:xlabel='Hours'>
```



THE DETAILED ANALYSIS BY PLOTTING DIFFERENT GRAPHS WE CAN CONCLUDE THAT THE SCORES OF THE STUDENTS INCREASED WITH THE NUMBER OF STUDY HOURS.

**STEP 7- Prepare data for the model

In [23]:
```
X = data.iloc[:, :-1].values
y = data.iloc[:, 1].values
# print(X)
```

**STEP 8- Divide the data for training and testing model

In [24]:
```
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)
```

**STEP 9- Training the algorithm

In [25]:
```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()


# from sklearn.ensemble import RandomForestRegressor
# regressor = RandomForestRegressor(n_estimators = 1000, random_state = 42)


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

```
Training complete.
```

In [26]:
```
# Plotting the regression line
# line = regressor.coef_*X+regressor.intercept_

# # Plotting for the test data
# plt.scatter(X, y)
# plt.plot(X, line);
# plt.show()
```

**STEP 10- Testing the model

In [33]:
```
print(X_test)
print("Predection of Score")
y_pred = regressor.predict(X_test)
print(y_pred)
```

```
[[2.7]
 [1.9]
 [7.7]
 [6.1]
 [4.5]]
Predection of Score
[28.6177145  20.88803334 76.92822173 61.46885942 46.0094971 ]
```

**STEP 11- Checking the accuracy of the model

In [28]:
```
df = pd.DataFrame({'Actual': y_test, 'Predicted': y_pred})

df
```

Out[28]:

	Actual	Predicted
0	30	28.617714
1	24	20.888033
2	85	76.928222
3	67	61.468859
4	41	46.009497

**STEP 12- Predicting the custom input

In [29]:
```
hours = [[9.25]]
pred = regressor.predict(hours)
print(pred)
```

```
[91.90447898]
```

**STEP 13- Evaluating the model

In [30]:
```
from sklearn import metrics
print('Mean Absolute Error:',
      metrics.mean_absolute_error(y_test, y_pred))
```

```
Mean Absolute Error: 4.621333622532769
```

ACCORDING TO THE GIVEN VARIABLES AND USING THE LINEAR REGRESSION METHOD WE CAN CONCLUDE THAT- If the student study for 9.25 hrs/day, than the predicted score of the student will be [91.90447898]