In [2]:

**import** pandas **as** pd *# Data Manuplation purpose (Data Verification, Data Cleaning, Data Validation)*   
**import** numpy **as** np *# Numerical Python - Calculation kind of things - Multidimensional Array - Rows and Columns*  
**import** matplotlib.pyplot **as** plt *# Visulalisation of Data - Ploting type of Data - Graphical way, pie, line, bar*

In [4]:

dataset**=**pd**.**read\_csv('student\_scores.csv') *# in the dataset we are importing student\_socres.csv file*  
dataset**.**shape *# It will Display entire Numbers of Rows and Columns in that particular Data Set. Display 25 Rows and 2 Columns*

Out[4]:

(25, 2)

In [8]:

dataset**.**head(7) *# head will display top 5 rows by default, if you need any particular number, need to mention*

Out[8]:

|  |  |  |
| --- | --- | --- |
|  | **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 |

In [9]:

dataset**.**describe() *# it will provide us all the stastical information regarding that particular dataset*

Out[9]:

|  |  |  |
| --- | --- | --- |
|  | **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 [13]:

dataset**.**plot(x**=**'Hours', y**=**'Scores', style**=**"\*") *# dataset i want make what and X and Y*  
plt**.**title('Student Mark Predition') *# This the Title for the Plot*  
plt**.**xlabel('Hours') *# X-axis Label is Hours*  
plt**.**ylabel('Percentage Marks') *# Y-axis Lable is Percentage Marks*  
plt**.**show()

In [14]:

X**=**dataset**.**iloc[:,:**-**1]**.**values *# It starts with zero from Left to Right*  
Y**=**dataset**.**iloc[:,:**-**1]**.**values *# It starts with Zero from Bottom to Top*

In [15]:

**from** sklearn.model\_selection **import** train\_test\_split *#we will selecting the model train\_test\_split*  
X\_train, X\_test, Y\_train, Y\_test**=**train\_test\_split(X, Y, test\_size**=**0.2, random\_state**=**0) *#splitting X and Y into 2 parts as Training 80% and Testing 20%, random\_state = 0 means randomly changing the values*

In [17]:

**from** sklearn.linear\_model **import** LinearRegression  
regressor**=**LinearRegression()  
regressor**.**fit(X\_train, Y\_train)

Out[17]:

LinearRegression()

In [18]:

print(regressor**.**intercept\_)

[-8.8817842e-16]

In [19]:

print(regressor**.**coef\_)

[[1.]]

In [32]:

y\_pred**=**regressor**.**predict(X\_test)  
df**=**pd**.**DataFrame(('Actual:' ,Y\_test,'Predicted:', y\_pred))  
df

Out[32]:

|  |  |
| --- | --- |
|  | **0** |
| **0** | Actual: |
| **1** | [[1.5], [3.2], [7.4], [2.5], [5.9]] |
| **2** | Predicted: |
| **3** | [[1.4999999999999996], [3.2], [7.4000000000000... |

In [ ]:

In [ ]: