Lab 5

```
## Fitting a simple linear regression model using "Study hours" as predictor and
"Exam score" as response variable.
import numpy as np

import matplotlib.pyplot as plt

from scipy import stats as st
```

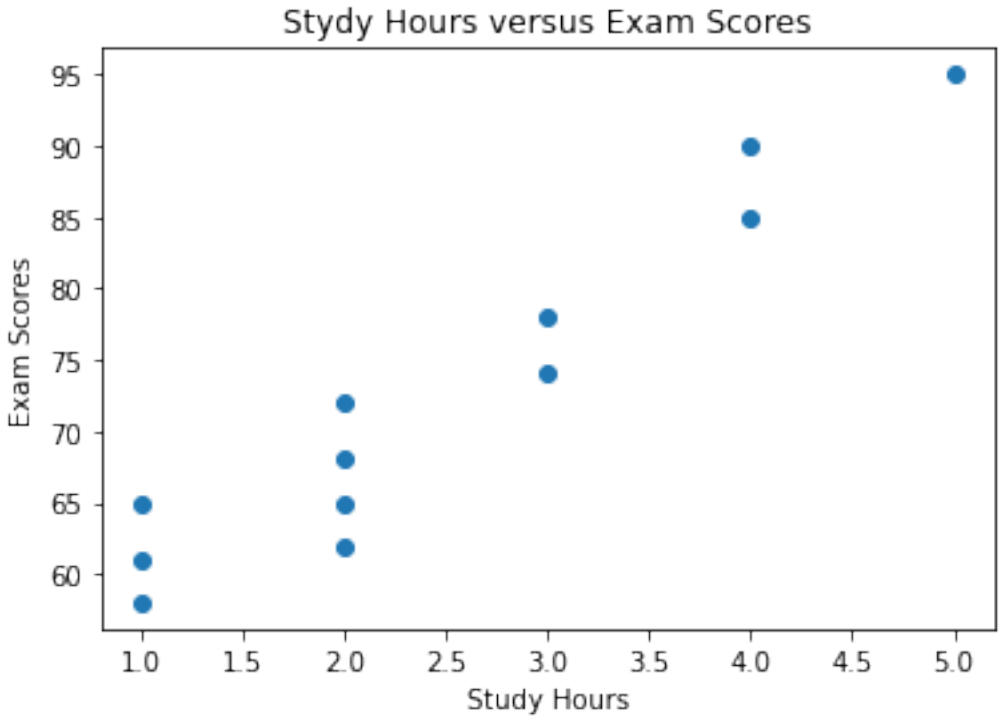
In [18]:

```
x = [1,1,2,2,1,2,2,3,3,4,4,5] #List of Study hours

y = [58,61,62,65,65,68,72,74,78,85,90,95]  #List of  Exam scores
```

In [19]:

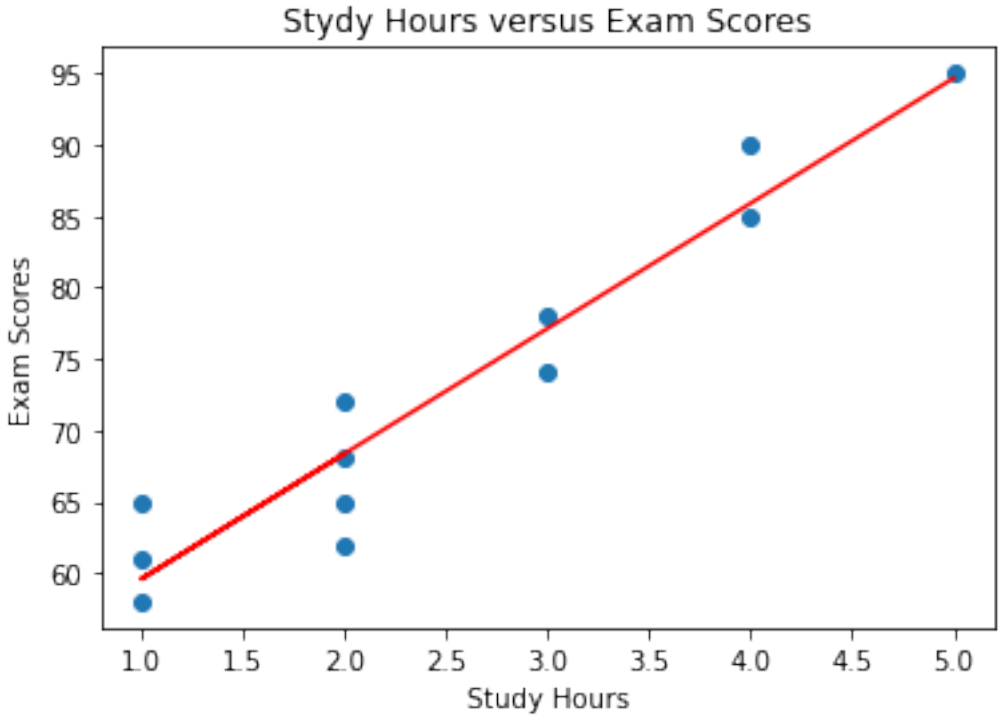
```
plt.scatter(x,y) # Scatter graph
plt.title("Stydy Hours versus Exam Scores")
plt.xlabel("Study Hours")
plt.ylabel("Exam Scores")
plt.show()
```



In [27]:

```
# Fitting the model
slope, intercept, r, p, std_err = st.linregress(x,y)
def createModel(x):                    # defining a python function for creating model
    respondedVar = slope*x+intercept
    return respondedVar
model = map(createModel,x) # Creating a model object as a map object
```

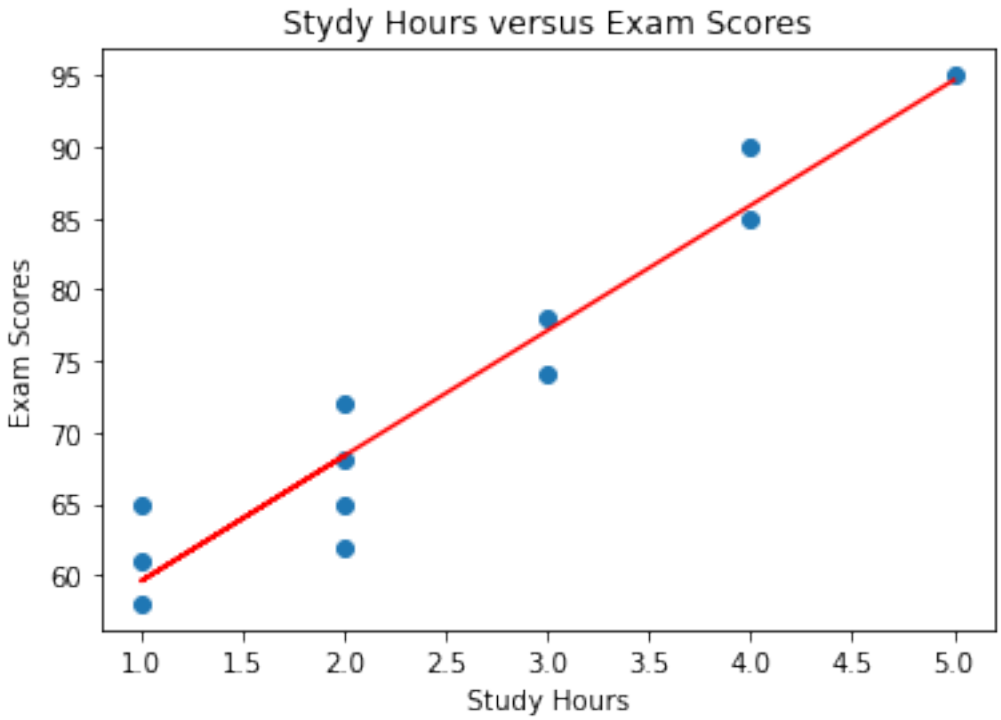
```
listModel = list(model) # Creating a list of model object
plt.plot(x,listModel,'r')
plt.scatter(x,y) # Scatter graph
plt.title("Stydy Hours versus Exam Scores")
plt.xlabel("Study Hours")
plt.ylabel("Exam Scores")
plt.show()
```



In [30]:

```
slope, intercept, r, p, std_err = st.linregress(x,y)
def createModel(x):                    # defining a python function for creating model
variables
    return slope*x+intercept

myModel = list(map(createModel,x)) # Creating a list of model object using python
map() function
plt.plot(x,myModel,'r')
plt.scatter(x,y) # Scatter graph
plt.title("Stydy Hours versus Exam Scores")
plt.xlabel("Study Hours")
plt.ylabel("Exam Scores")
plt.show()
```



In [32]:

```
print(r) # Relationship strenth, thst is correlation beteween study hours and exam scores
```

```
0.9590729006011031
```

Here, correlation between study hours and exam scores (r) is 0.959, which is greater than 0.5. So, r>0.5 is a strong correlation . So, the professor can confidently use this relationship