In [2]:

```
# model optimization
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
from sklearn.linear_model import LinearRegression
```

In [3]:

```
# reading the dataset
data = pd.read_csv("C:\\Users\\hm\\Desktop\\LINEAR PROGRAMMING CENTRALS\\studer
```

In [4]:

```
data
```

Out[4]:

	Study Hours	Exam Scores
0	3.7	87.9
1	9.5	143.6
2	7.3	123.7
3	6.0	99.9
4	1.6	64.5
...	...	...
95	4.9	95.3
96	5.2	101.9
97	4.3	94.5
98	0.3	53.9
99	1.1	64.9

100 rows × 2 columns

In [5]:

```
x = np.array(data["Study Hours"]).reshape(-1,1)
y = np.array(data["Exam Scores"])
```

In [6]:

```
x
```

```
Out[6]: array([[3.7],
               [9.5],
               [7.3],
               [6. ],
               [1.6],
               [1.6],
               [0.6],
               [8.7],
               [6. ],
               [7.1],
               [0.2],
               [9.7],
               [8.3],
               [2.1],
               [1.8],
               [1.8],
               [3. ],
               [5.2],
               [4.3],
               [2.9],
               [6.1],
               [1.4],
               [2.9],
               [3.7],
               [4.6],
               [7.9],
               [2. ],
               [5.1],
               [5.9],
               [0.5],
               [6.1],
               [1.7],
               [0.7],
               [9.5],
               [9.7],
               [8.1],
               [3. ],
               [1. ],
               [6.8],
               [4.4],
               [1.2],
               [5. ],
               [0.3],
               [9.1],
               [2.6],
               [6.6],
               [3.1],
               [5.2],
               [5.5],
               [1.8],
               [9.7],
               [7.8],
               [9.4],
               [8.9],
               [6. ],
               [9.2],
               [0.9],
       [2. ],
       [0.5],
       [3.3],
       [3.9],
       [2.7],
       [8.3],
       [3.6],
       [2.8],
       [5.4],
       [1.4],
       [8. ],
       [0.7],
       [9.9],
       [7.7],
       [2. ],
       [0.1],
       [8.2],
       [7.1],
       [7.3],
       [7.7],
       [0.7],
       [3.6],
       [1.2],
       [8.6],
       [6.2],
       [3.3],
       [0.6],
       [3.1],
       [3.3],
       [7.3],
       [6.4],
       [8.9],
       [4.7],
       [1.2],
       [7.1],
       [7.6],
       [5.6],
       [7.7],
       [4.9],
       [5.2],
       [4.3],
       [0.3],
       [1.1]])
```

```
In [7]: y
```

```
Out[7]: array([ 87.9, 143.6, 123.7,  99.9,  64.5,  67.4,  63.2, 134. , 106.1,
               118.3,  56.6, 148.6, 130.6,  73.8,  68.7,  73.2,  76.9, 100.8,
                91.2,  71.8, 112.7,  65.3,  79.2,  85.5,  88.5, 126.4,  68.3,
                97.4, 108.4,  56.7, 120.2,  67.9,  57.8, 144.5, 137. , 130.7,
                80.8,  72.1, 117.5,  95.5,  62. ,  93.7,  59.2, 144.7,  79.8,
               111.7,  88.2,  95. , 107.6,  79.4, 142. , 124.7, 144.4, 137. ,
               102. , 142.5,  53.5,  72. ,  49.9,  90.3,  85. ,  75.5, 136.9,
                79.5,  79.2, 110.8,  56.1, 131.1,  58.8, 152.6, 121. ,  63.3,
                53.2, 133. , 121.9, 124.6, 123.7,  58.6,  87.3,  58. , 145.6,
               114.7,  77.1,  59.6,  76.2,  86.5, 128.8, 109.7, 143.5,  99.3,
                66.1, 130.8, 124.9, 102.4, 122.6,  95.3, 101.9,  94.5,  53.9,
                64.9])
```

```
In [8]: data.isnull().sum()
```

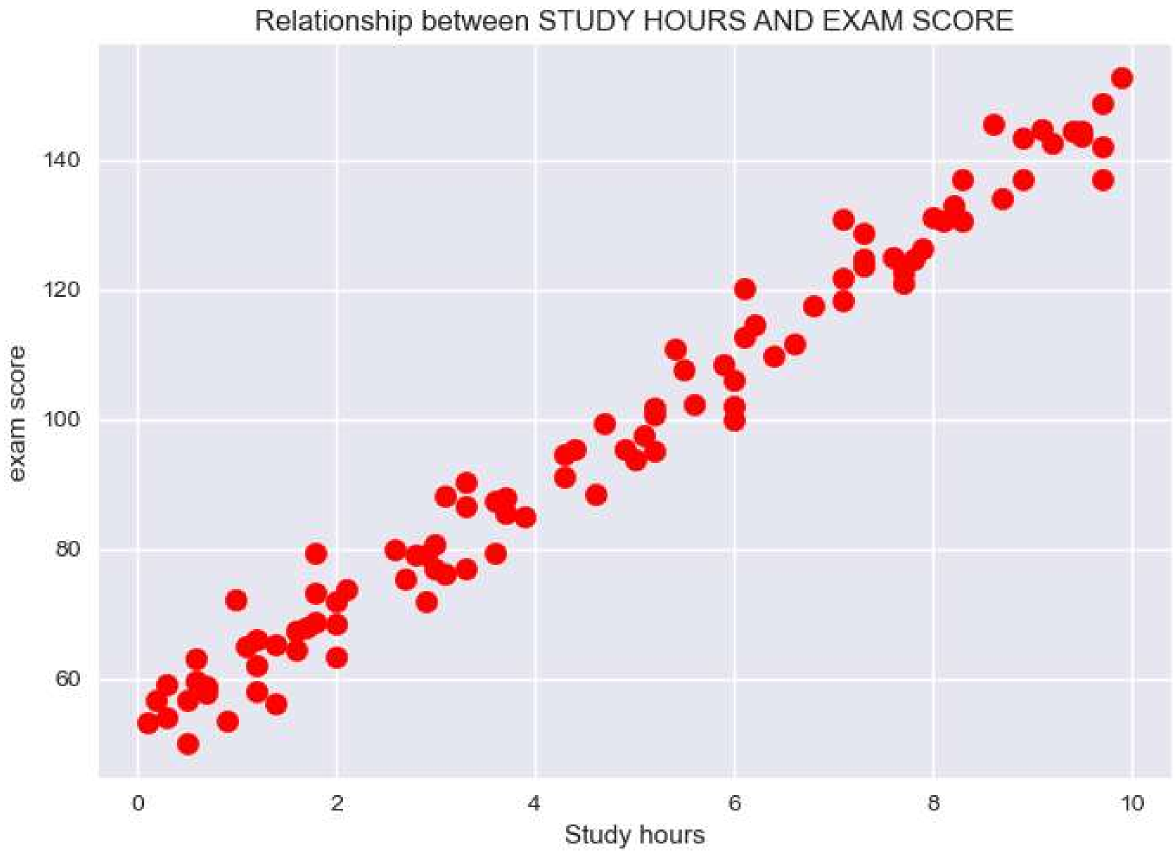
```
Out[8]: Study Hours    0
        Exam Scores    0
        dtype: int64
```

```
In [9]: # visualizing the relationship between  study hours and exam score
        import matplotlib.pyplot as plt
        from matplotlib import style
```

In [10]:
```
# ploting the scatter graph
style.use("seaborn")
plt.scatter(x, y, label= "datapoints", color = "red", s = 80)
plt.title("Relationship between STUDY HOURS AND EXAM SCORE")
plt.xlabel("Study hours")
plt.ylabel("exam score")
```

```
C:\Users\hm\AppData\Local\Temp\ipykernel_9728\1976094932.py:2: MatplotlibDepr
ecationWarning: The seaborn styles shipped by Matplotlib are deprecated since
3.6, as they no longer correspond to the styles shipped by seaborn. However,
they will remain available as 'seaborn-v0_8-<style>'. Alternatively, directly
use the seaborn API instead.
  style.use("seaborn")
```

Out[10]:
```
Text(0, 0.5, 'exam score')
```



In [11]:
```
# spliting the dataset into train and test
x_train, x_test, y_train,y_test =  train_test_split(x,y, test_size = 0.2, rand
```

In [12]:
```
# data preprocessing
scaler = StandardScaler()
x_train_scaled = scaler.fit_transform(x_train)
x_test_scaled = scaler.transform(x_test)
```

```
In [13]: # building the linear regreesion model
         model =LinearRegression()
         model.fit(x_train,y_train)

Out[13]: ▾ LinearRegression
         LinearRegression()
```

```
In [14]: y_pred = model.predict(x_test_scaled)
```

```
In [15]: y_pred

Out[15]: array([36.94787956, 64.87197905, 60.83475984, 57.13397558, 43.67657823,
                49.73240704, 44.68588303, 63.86267425, 35.60213983, 47.3773625 ,
                49.3959721 , 55.45180091, 62.51693451, 66.89058865, 38.96648916,
                40.3122289 , 60.83475984, 37.2843145 , 62.85336944, 40.64866383])
```

EVALUATING THE MODEL

```
In [16]: MAE =mean_absolute_error(y_test, y_pred)
         MSE = mean_squared_error(y_test, y_pred)
         R2 =r2_score(y_test, y_pred)
```

```
In [17]: MAE

Out[17]: 47.715961961709375
```

```
In [18]: MSE

Out[18]: 2682.9281679636674
```

```
In [19]: R2

Out[19]: -1.8659723784262328
```

```
In [20]: model.intercept_

Out[20]: 50.691246598122966
```

```
In [21]: model.coef_

Out[21]: array([9.79642549])
```

In [22]:
```
# improving the model
y_pred_improved= model.predict(x_test_scaled)
MAE_improved = mean_absolute_error(y_test,y_pred_improved)
MSE_improved = mean_squared_error(y_test,y_pred_improved)
R2_improved = r2_score(y_test, y_pred_improved)
```

In [23]:
```
MAE_improved
```

Out[23]:
```
47.715961961709375
```

In [24]:
```
MSE_improved
```

Out[24]:
```
2682.9281679636674
```

In [ ]: