In [1]:

```
import pandas as pd
import numpy as np
```

In [34]:

```
from matplotlib import style
import matplotlib.pyplot as plt
import os
```

In [35]:

```
data=pd.read_csv(r"C:\Users\Abrar\Desktop\New folder (4)\students_dataset.csv")
data.head()
```

Out[35]:

	study_hours_student	student_marks	health_condition_rate_0_to_100
0	7.00	80.00	56.0
1	6.59	77.00	34.0
2	7.23	78.68	45.0
3	5.67	71.82	67.0
4	8.67	84.19	35.0

In [19]:

```
pwd
```

Out[19]:

```
'C:\\Users\\Abrar\\Desktop\\New folder (4)'
```

In [27]:

```
data.tail
```

Out[27]:

```
<bound method NDFrame.tail of      study hours student  student_marks  health condition rate 0 to 100
0                    7.00          80.00                            56.0
1                    6.59          77.00                            34.0
2                    7.23          78.68                            45.0
3                    5.67          71.82                            67.0
4                    8.67          84.19                            35.0
..                    ...            ...                             ...
101                  8.69          83.57                            67.0
102                  8.75          85.95                            35.0
103                  6.46          76.02                            24.0
104                  7.14          77.65                            75.0
105                  6.38          77.01                            86.0

[106 rows x 3 columns]>
```

In [29]:

```
data.shape
```

Out[29]:

```
(106, 3)
```

In [30]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 106 entries, 0 to 105
Data columns (total 3 columns):
study hours student               104 non-null float64
student_marks                     106 non-null float64
health condition rate 0 to 100    104 non-null float64
dtypes: float64(3)
memory usage: 2.6 KB
```

In [31]:

```
data.describe()
```

Out[31]:

	study hours student	student_marks	health condition rate 0 to 100
count	104.000000	106.000000	104.000000
mean	6.881635	77.403302	59.153846
std	1.245280	4.821349	19.053715
min	5.010000	68.570000	22.000000
25%	5.685000	73.122500	44.000000
50%	6.815000	77.180000	56.000000
75%	7.950000	81.587500	76.000000
max	8.990000	86.650000	99.000000

In [37]:

```
plt.scatter(x=data.study_hours_student, y=data.student_marks)
plt. xlabel("student hours of study")
plt.ylabel("marks of student")
plt.title("plot show student marks depends on houres of study")
plt.show()
```

plt.scatter(x=data.student_marks,y=data.health_condition_rate_0_to_100)

data cleaning now¶

In [40]:

```
data.isnull()
```

Out[40]:

	study_hours_student	student_marks	health_condition_rate_0_to_100
0	False	False	False
1	False	False	False
2	False	False	False
3	False	False	False
4	False	False	False
...	...	...	...
101	False	False	False
102	False	False	False
103	False	False	False
104	False	False	False
105	False	False	False

106 rows × 3 columns

In [41]:

```
data.isnull().sum()
```

Out[41]:

```
study_hours_student               2
student_marks                     0
health_condition_rate_0_to_100    2
dtype: int64
```

In [42]:

```
data.mean()
```

Out[42]:

```
study_hours_student                6.881635
student_marks                     77.403302
health_condition_rate_0_to_100    59.153846
dtype: float64
```

In [43]:

```
datafram_without_null=data.fillna(data.mean())
```

In [45]:

```
datafram_without_null.isnull().sum()
```

Out[45]:

```
study_hours_student               0
student_marks                     0
health_condition_rate_0_to_100    0
dtype: int64
```

In [46]:

```
datafram_without_null.head()
```

Out[46]:

	study_hours_student	student_marks	health_condition_rate_0_to_100
0	7.00	80.00	56.0
1	6.59	77.00	34.0
2	7.23	78.68	45.0
3	5.67	71.82	67.0
4	8.67	84.19	35.0

now slit the dataet trainung and testing¶

In [108]:

```
x=datafram_without_null.drop("student_marks", axis = "columns")
y=datafram_without_null.drop("study_hours_student", axis = "columns")
z=datafram_without_null.drop("health_condition_rate_0_to_100", axis = "columns")
```

In [109]:

```
print("shape of x is ",x.shape)
print("shape of y is ",y.shape)
print("shape of z is ",z.shape)
```

```
shape of x is  (106, 2)
shape of y is  (106, 2)
shape of z is  (106, 2)
```

In [110]:

```
from sklearn.model_selection import train_test_split
```

In [118]:

```
train_test_split(x,y,z,test_size=0.2, random_state=51)
```

Out[118]:

```
[     study_hours_student  health_condition_rate_0_to_100
 78              7.950000                           42.0
 12              7.750000                           89.0
 83              5.340000                           75.0
 86              5.020000                           56.0
 70              7.900000                           89.0
 ..                   ...                            ...
 69              5.440000                           78.0
 73              6.881635                           68.0
 96              6.600000                           64.0
 101             8.690000                           67.0
 57              8.340000                           56.0

 [84 rows x 2 columns],
      study_hours_student  health_condition_rate_0_to_100
 44              8.260000                           67.0
 58              7.940000                           78.0
 102             8.750000                           35.0
 99              6.881635                           75.0
 87              7.980000                           66.0
 11              8.560000                           78.0
 47              5.370000                           56.0
 98              8.310000                           66.0
 4               8.670000                           35.0
 92              5.790000                           45.0
 43              7.860000                           45.0
 72              6.090000                           22.0
 76              8.070000                           66.0
 80              7.310000                           79.0
 19              5.460000                           64.0
 32              8.690000                           78.0
 59              6.660000                           89.0
 2               7.230000                           45.0
 79              8.260000                           76.0
 68              8.200000                           56.0
 0               7.000000                           56.0
 56              5.880000                           44.0,
      student_marks  health_condition_rate_0_to_100
 78           82.03                           42.0
 12           79.50                           89.0
 83           72.10                           75.0
 86           70.58                           56.0
 70           79.10                           89.0
 ..             ...                            ...
 69           72.08                           78.0
 73           75.39                           68.0
 96           75.55                           64.0
 101          83.57                           67.0
 57           84.00                           56.0

 [84 rows x 2 columns],
      student_marks  health_condition_rate_0_to_100
 44           81.70                           67.0
 58           82.93                           78.0
 102          85.95                           35.0
 99           76.83                           75.0
 87           81.08                           66.0
 11           83.88                           78.0
 47           71.80                           56.0
 98           82.69                           66.0
 4            84.19                           35.0
 92           74.44                           45.0
 43           81.25                           45.0
 72           76.48                           22.0
 76           82.30                           66.0
 80           79.26                           79.0
 19           71.10                           64.0
 32           85.48                           78.0
 59           76.63                           89.0
 2            78.68                           45.0
 79           82.99                           76.0
 68           82.10                           56.0
```

```
0           80.00                          56.0
56          73.34                          44.0,
     study_hours_student  student_marks
78              7.950000          82.03
12              7.750000          79.50
83              5.340000          72.10
86              5.020000          70.58
70              7.900000          79.10
..                   ...            ...
69              5.440000          72.08
73              6.881635          75.39
96              6.600000          75.55
101             8.690000          83.57
57              8.340000          84.00

[84 rows x 2 columns],
     study_hours_student  student_marks
44              8.260000          81.70
58              7.940000          82.93
102             8.750000          85.95
99              6.881635          76.83
87              7.980000          81.08
11              8.560000          83.88
47              5.370000          71.80
98              8.310000          82.69
4               8.670000          84.19
92              5.790000          74.44
43              7.860000          81.25
72              6.090000          76.48
76              8.070000          82.30
80              7.310000          79.26
19              5.460000          71.10
32              8.690000          85.48
59              6.660000          76.63
2               7.230000          78.68
79              8.260000          82.99
68              8.200000          82.10
0               7.000000          80.00
56              5.880000          73.34]
```

In [119]:

```
x_train,x_test,train_test_split(x,y,z,test_size=0.2, random_state=51)
```

Out[119]:

```
(    study_hours_student  health_condition_rate_0_to_100
 17             5.280000                           67.0
 10             6.590000                           57.0
 29             8.710000                           46.0
 54             6.560000                           46.0
 94             8.830000                           67.0
 ..                  ...                            ...
 48             5.110000                           68.0
 4              8.670000                           35.0
 56             5.880000                           44.0
 79             8.260000                           76.0
 73             6.881635                           68.0

 [84 rows x 2 columns],
     study_hours_student  health_condition_rate_0_to_100
 45                 5.07                      78.000000
 93                 5.40                      55.000000
 21                 8.09                      56.000000
 51                 7.31                      42.000000
 23                 7.01                      89.000000
 52                 6.04                      56.000000
 71                 7.69                      35.000000
 75                 8.88                      90.000000
 36                 5.65                      34.000000
 86                 5.02                      56.000000
 55                 5.09                      33.000000
 40                 7.72                      78.000000
 105                6.38                      86.000000
 13                 7.90                      80.000000
 38                 7.26                      34.000000
 11                 8.56                      78.000000
 47                 5.37                      56.000000
 31                 8.76                      56.000000
 81                 7.23                      90.000000
 89                 6.92                      59.153846
 42                 5.45                      66.000000
 3                  5.67                      67.000000,
 [     study_hours_student  health_condition_rate_0_to_100
  78             7.950000                           42.0
  12             7.750000                           89.0
  83             5.340000                           75.0
  86             5.020000                           56.0
  70             7.900000                           89.0
  ..                  ...                            ...
  69             5.440000                           78.0
  73             6.881635                           68.0
  96             6.600000                           64.0
  101            8.690000                           67.0
  57             8.340000                           56.0

  [84 rows x 2 columns],
       study_hours_student  health_condition_rate_0_to_100
```

```
44              8.260000                           67.0
58              7.940000                           78.0
102             8.750000                           35.0
99              6.881635                           75.0
87              7.980000                           66.0
11              8.560000                           78.0
47              5.370000                           56.0
98              8.310000                           66.0
4               8.670000                           35.0
92              5.790000                           45.0
43              7.860000                           45.0
72              6.090000                           22.0
76              8.070000                           66.0
80              7.310000                           79.0
19              5.460000                           64.0
32              8.690000                           78.0
59              6.660000                           89.0
2               7.230000                           45.0
79              8.260000                           76.0
68              8.200000                           56.0
0               7.000000                           56.0
56              5.880000                           44.0,
     student_marks  health_condition_rate_0_to_100
78           82.03                           42.0
12           79.50                           89.0
83           72.10                           75.0
86           70.58                           56.0
70           79.10                           89.0
..             ...                            ...
69           72.08                           78.0
73           75.39                           68.0
96           75.55                           64.0
101          83.57                           67.0
57           84.00                           56.0

[84 rows x 2 columns],
     student_marks  health_condition_rate_0_to_100
44           81.70                           67.0
58           82.93                           78.0
102          85.95                           35.0
99           76.83                           75.0
87           81.08                           66.0
11           83.88                           78.0
47           71.80                           56.0
98           82.69                           66.0
4            84.19                           35.0
92           74.44                           45.0
43           81.25                           45.0
72           76.48                           22.0
76           82.30                           66.0
80           79.26                           79.0
19           71.10                           64.0
32           85.48                           78.0
59           76.63                           89.0
2            78.68                           45.0
79           82.99                           76.0
68           82.10                           56.0
0            80.00                           56.0
56           73.34                           44.0,
     study_hours_student  student_marks
78              7.950000          82.03
12              7.750000          79.50
83              5.340000          72.10
86              5.020000          70.58
70              7.900000          79.10
..                   ...            ...
69              5.440000          72.08
73              6.881635          75.39
96              6.600000          75.55
101             8.690000          83.57
57              8.340000          84.00

[84 rows x 2 columns],
     study_hours_student  student_marks
44              8.260000          81.70
58              7.940000          82.93
102             8.750000          85.95
99              6.881635          76.83
87              7.980000          81.08
11              8.560000          83.88
47              5.370000          71.80
98              8.310000          82.69
4               8.670000          84.19
92              5.790000          74.44
43              7.860000          81.25
72              6.090000          76.48
76              8.070000          82.30
80              7.310000          79.26
19              5.460000          71.10
32              8.690000          85.48
59              6.660000          76.63
2               7.230000          78.68
79              8.260000          82.99
68              8.200000          82.10
0               7.000000          80.00
56              5.880000          73.34])
```

In [100]:

```
print("shape of x is ",x_train.shape)
print("shape of y is ",y_train.shape)
print("shape of z is ",z_train.shape)
print("shape of x is ",x_test.shape)
print("shape of y is ",y_test.shape)
print("shape of z is ",z_test.shape)
```

```
shape of x is  (84, 2)
shape of y is  (84, 2)
shape of z is  (84, 2)
shape of x is  (22, 2)
shape of y is  (22, 2)
shape of z is  (22, 2)
```

model selection and also traing¶

In [66]:

```
#y=m*x+c
```

In [79]:

```
from sklearn.linear_model import LinearRegression
```

In [120]:

```
lr=LinearRegression()
```

In [121]:

```
lr.fit(x_train,y_train)
```

Out[121]:

```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

In [122]:

```
lr.intercept_
```

Out[122]:

```
array([ 5.18357679e+01, -2.84217094e-14])
```

In [123]:

```
lr.coef_
```

Out[123]:

```
array([[ 3.79554985e+00, -6.58996116e-03],
       [-6.35106154e-16,  1.00000000e+00]])
```

In [129]:

```
m=3.79
c=50.18
y =m * 7 + c
y
```

Out[129]:

```
76.71000000000001
```

In [139]:

```
x_predicted=lr.predict(x_test)
x_predicted
```

Out[139]:

```
array([[70.56518866, 78.        ],
       [71.96928922, 55.        ],
       [82.17272836, 56.        ],
       [79.30445893, 42.        ],
       [77.8560658 , 89.        ],
       [74.39185116, 56.        ],
       [80.79289761, 35.        ],
       [84.94715407, 90.        ],
       [73.05656586, 34.        ],
       [70.52039031, 56.        ],
       [70.93764791, 33.        ],
       [80.62339577, 78.        ],
       [75.48463928, 86.        ],
       [81.29341482, 80.        ],
       [79.16740113, 34.        ],
       [83.81165765, 78.        ],
       [71.84883276, 56.        ],
       [84.71574676, 56.        ],
       [78.68449681, 90.        ],
       [77.71115131, 59.15384615],
       [72.08657714, 66.        ],
       [72.91500814, 67.        ]])
```

In [140]:

```
y_predicted=lr.predict(y_test)
y_predicted
```

Out[140]:

```
array([[314.23948927,  78.        ],
       [318.45229672,  55.        ],
       [364.144127  ,  56.        ],
       [346.05570266,  42.        ],
       [345.252553  ,  89.        ],
       [325.99885097,  56.        ],
       [359.08261288,  35.        ],
       [368.70246114,  90.        ],
       [323.75263371,  34.        ],
       [319.35663872,  56.        ],
       [317.49656641,  33.        ],
       [356.59782564,  78.        ],
       [343.56432545,  86.        ],
       [357.83717717,  80.        ],
       [353.01632308,  34.        ],
       [369.69247263,  78.        ],
       [323.98720955,  56.        ],
       [363.49888352,  56.        ],
       [345.58756253,  90.        ],
       [341.35004415,  59.15384615],
       [317.31705319,  66.        ],
       [323.99063097,  67.        ]])
```

In [141]:

```
z_predicted=lr.predict(z_test)
z_predicted
```

Out[141]:

```
array([[70.62271902, 69.27      ],
       [71.86819921, 70.34      ],
       [81.99888519, 82.38      ],
       [79.06992222, 77.59      ],
       [77.93211395, 77.46      ],
       [74.2842371 , 72.33      ],
       [80.48969349, 81.01      ],
       [84.98906622, 83.64      ],
       [72.80812433, 71.7       ],
       [70.42430868, 70.58      ],
       [70.69348985, 70.05      ],
       [80.60738217, 80.43      ],
       [75.54388303, 77.01      ],
       [81.28840645, 80.76      ],
       [78.86815099, 79.41      ],
       [83.77290868, 83.88      ],
       [71.74471137, 71.8       ],
       [84.54302388, 82.21      ],
       [78.76654183, 77.55      ],
       [77.59763163, 76.38      ],
       [72.05982189, 70.06      ],
       [72.88324453, 71.82      ]])
```

lets tune model and test afficiency and correctnesss of our model¶

In [142]:

```
lr.score(x_test,y_test)
```

```
C:\Users\Abrar\Anaconda3\anaconda\lib\site-packages\sklearn\base.py:420: FutureWarning: The default value of multioutput (not exposed in score method) wil
  "multioutput='uniform_average')."， FutureWarning)
```

Out[142]:

```
0.9959707534587428
```

you see best tuned model corectness 0.99¶

In [143]:

```
plt.scatter(x_train,y_train)
```

Out[143]:

```
<matplotlib.collections.PathCollection at 0x6f8bf99c8>
```

In [145]:

```
plt.scatter(x_test,y_test)
plt.plot(x_train, lr.predict(x_train), color="r")
```

Out[145]:

```
[<matplotlib.lines.Line2D at 0x6f91af6c8>,
 <matplotlib.lines.Line2D at 0x6f91d9e48>]
```

In [146]:

```
import joblib
```

In [148]:

```
joblib.dump(lr,"Student performance prediction base on hours of study.pkl")
```

Out[148]:

```
['Student performance prediction base on hours of study.pkl']
```

In [ ]: