

In []:

```
In [7]: import pandas as pd
import numpy as np
import matplotlib as plt
import seaborn as sb
import sklearn as skl
```

```
In [8]: import os
os.getcwd()
```

Out[8]: 'C:\\\\Users\\nadir albab'

```
In [9]: os.chdir("C:\\\\Users\\nadir albab\\Desktop")
```

```
In [10]: os.getcwd()
```

Out[10]: 'C:\\\\Users\\nadir albab\\Desktop'

```
In [11]: df = pd.read_csv('student_scores - student_scores.csv')
```

```
In [12]: df.head(25)
```

Out[12]:

	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
10	7.7	85
11	5.9	62
12	4.5	41
13	3.3	42
14	1.1	17
15	8.9	95
16	2.5	30
17	1.9	24
18	6.1	67
19	7.4	69
20	2.7	30
21	4.8	54
22	3.8	35
23	6.9	76
24	7.8	86

```
In [13]: df.shape
```

Out[13]: (25, 2)

```
In [14]: df.info()
```

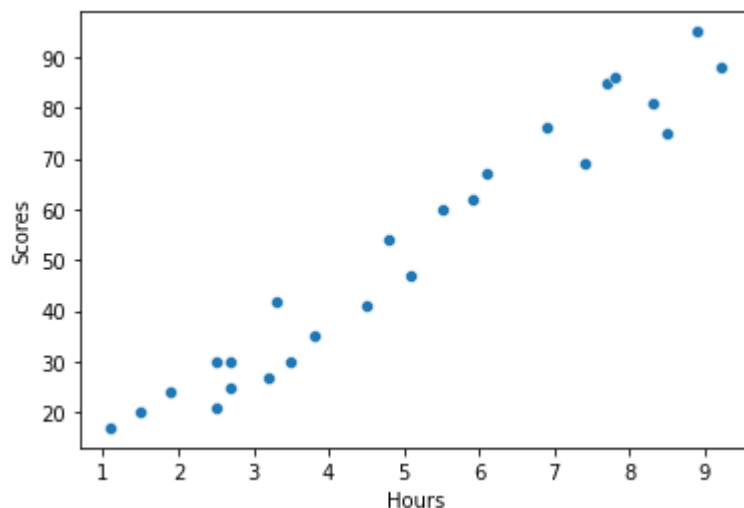
```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 25 entries, 0 to 24  
Data columns (total 2 columns):  
#   Column  Non-Null Count  Dtype    
---  ---      -  
0   Hours    25 non-null    float64  
1   Scores   25 non-null    int64     
dtypes: float64(1), int64(1)  
memory usage: 528.0 bytes
```

```
In [15]: df.describe()
```

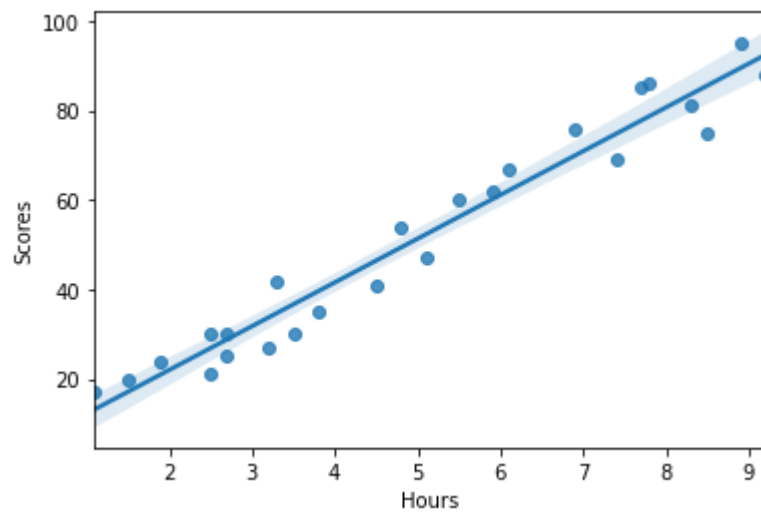
Out[15]:

	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 [16]: sb.scatterplot(x=df['Hours'], y=df['Scores']);
```



```
In [17]: sb.regplot(x=df['Hours'], y=df['Scores']);
```



```
In [18]: X = df[['Hours']]  
y = df['Scores']
```

```
In [19]: X
```

Out[19]:

Hours	
0	2.5
1	5.1
2	3.2
3	8.5
4	3.5
5	1.5
6	9.2
7	5.5
8	8.3
9	2.7
10	7.7
11	5.9
12	4.5
13	3.3
14	1.1
15	8.9
16	2.5
17	1.9
18	6.1
19	7.4
20	2.7
21	4.8
22	3.8
23	6.9
24	7.8

In [20]:

y

Out[20]:

0	21
1	47
2	27
3	75
4	30
5	20
6	88
7	60
8	81
9	25
10	85
11	62
12	41
13	42
14	17
15	95
16	30
17	24
18	67
19	69
20	30
21	54
22	35
23	76
24	86

Name: Scores, dtype: int64

In [21]:

```
from sklearn.model_selection import train_test_split
train_X, val_X, train_y, val_y = train_test_split(X, y, random_state = 0)
```

In [22]:

```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
```

In [23]:

```
regressor.fit(train_X, train_y)
```

Out[23]:

LinearRegression()

In [24]:

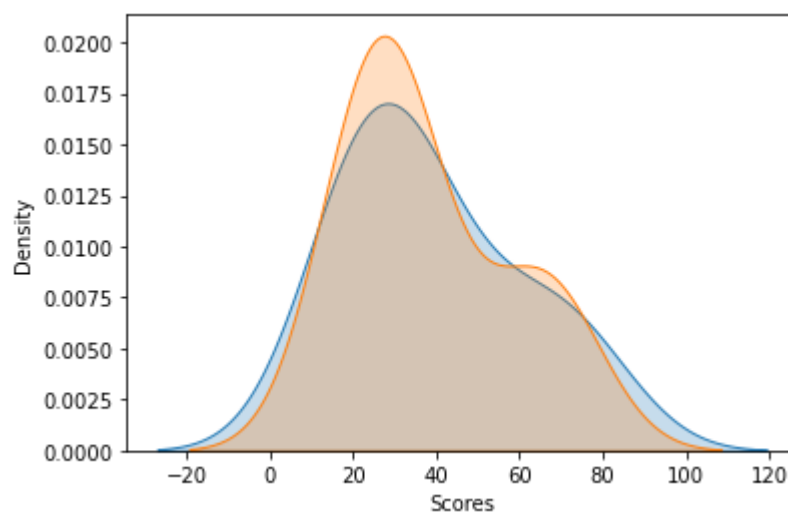
```
pred_y = regressor.predict(val_X)
```

```
In [25]: pd.DataFrame({'Actual': val_y, 'Predicted': pred_y,})
```

Out[25]:

	Actual	Predicted
5	20	16.844722
2	27	33.745575
19	69	75.500624
16	30	26.786400
11	62	60.588106
22	35	39.710582
17	24	20.821393

```
In [26]: sb.kdeplot(pred_y,label="Predicted", shade=True);
sb.kdeplot(data=val_y, label="Actual", shade=True);
```



```
In [27]: print('Train accuracy: ', regressor.score(train_X, train_y),
              '\nTest accuracy : ', regressor.score(val_X, val_y) )
```

Train accuracy: 0.9484509249326872
Test accuracy : 0.9367661043365056

```
In [28]: h = [[9.25]]
s = regressor.predict(h)
print('A student who will study ', h[0][0] , ' hours is estimated to score ', s[0])
```

A student who will study 9.25 hours is estimated to score 93.89272889341652

```
In [29]: from sklearn import metrics  
print('Mean Absolute Error:',  
      metrics.mean_absolute_error(val_y, pred_y))
```

Mean Absolute Error: 4.130879918502482

In []:

In []:

In []:

In []: