1.Importing important library

In [23]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

2. Importing dataset

In [2]:
```
data1 = pd.read_csv('delivery_time.csv',sep=',')
data1.head()
```

Out[2]:

	Delivery Time	Sorting Time
0	21.00	10
1	13.50	4
2	19.75	6
3	24.00	9
4	29.00	10

3. Understanding dataset

In [3]:
```
data1.shape
```

Out[3]:
```
(21, 2)
```

In [4]:
```
data1.dtypes
```

Out[4]:
```
Delivery Time    float64
Sorting Time       int64
dtype: object
```

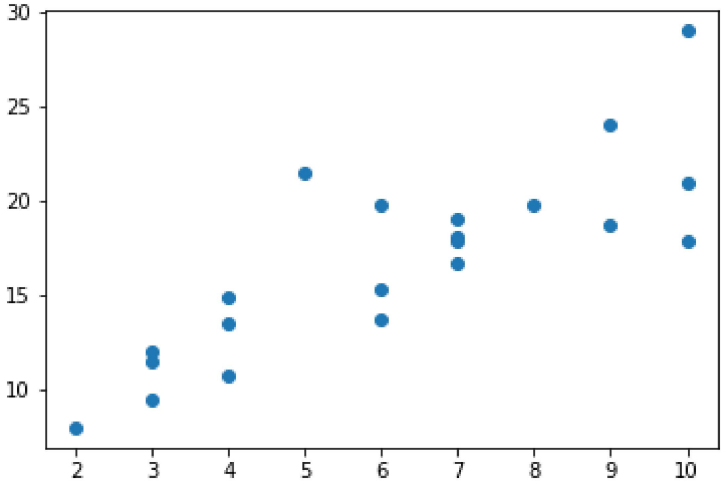
In [5]:
```
data1.isnull().sum()
```

Out[5]:
```
Delivery Time    0
Sorting Time     0
dtype: int64
```

4. checking assumptions

In [6]:
```
plt.scatter(x='Sorting Time',y='Delivery Time',data=data1)
```

Out[6]:
```
<matplotlib.collections.PathCollection at 0x1df6dd6ac70>
```



In [7]:
```
data1.corr()
```

Out[7]:

	Delivery Time	Sorting Time
Delivery Time	1.000000	0.825997
Sorting Time	0.825997	1.000000

In [8]:
```
data1.std()
```

Out[8]:
```
Delivery Time    5.074901
Sorting Time     2.542028
dtype: float64
```

In [9]:
```
## data follows homoscadacity and has no autoregression
```

5. Model Building

In [10]:
```
x = data1[['Sorting Time']]
y = data1[['Delivery Time']]
```

In [11]:
```
from sklearn.model_selection import train_test_split
```

In [12]:
```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.20)
```

In [13]:
```
print(x_train.shape)
print(y_train.shape)
```

```
(16, 1)
(16, 1)
```

In [14]:
```
print(x_test.shape)
print(y_test.shape)
```

```
(5, 1)
(5, 1)
```

6. Model Training

In [15]:
```
from sklearn.linear_model import LinearRegression
```

In [16]:
```
linear_model = LinearRegression()
linear_model.fit(x_train,y_train)
```

Out[16]:
```
LinearRegression()
```

In [17]:
```
print('slope: ',linear_model.coef_)
print('intercept: ',linear_model.intercept_)
```

```
slope:  [[1.663506]]
intercept:  [6.69605812]
```

7. Model Testing

In [18]:
```
y_predict = linear_model.predict(x_test)
y_predict
```

Out[18]:
```
array([[23.33111813],
       [18.34060013],
       [11.68657612],
       [18.34060013],
       [20.00410613]])
```

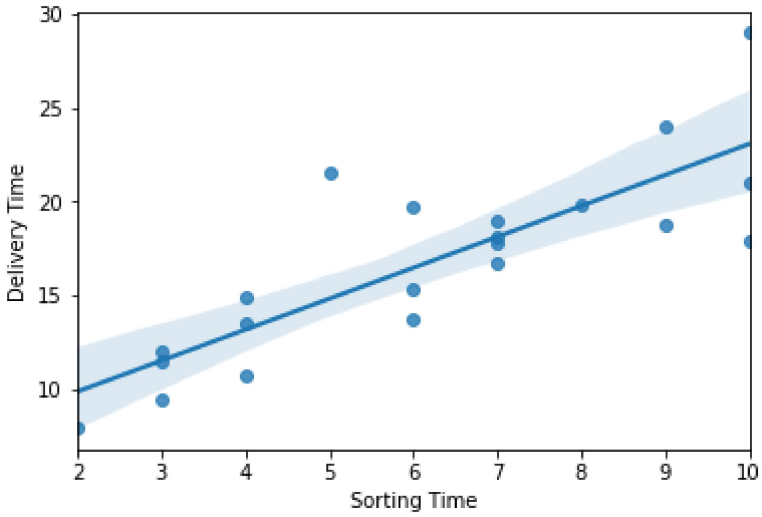
In [19]:
```
y_test
```

Out[19]:

	Delivery Time
0	21.00
17	18.11
7	9.50
6	19.00
10	19.83

In [24]:
```
sns.regplot(x='Sorting Time',y='Delivery Time', data=data1)
```

Out[24]:
```
<AxesSubplot:xlabel='Sorting Time', ylabel='Delivery Time'>
```



8. Model Evaluation

In [20]:
```
from sklearn.metrics import mean_squared_error,r2_score
```

In [21]:
```
mean_squared_error(y_test,y_predict)
```

Out[21]: 2.1467048852283583

In [22]:
```
r2_score(y_test,y_predict)
```

Out[22]: 0.8726751034973811

In [ ]:
```

```