In [1]:
```
#importing the data
import pandas as pd
```

In [2]:
```
data=pd.read_csv('C:\\Users\\DELL\\Downloads\\delivery_time.csv')
data
```

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 |
| **5** | 15.35 | 6 |
| **6** | 19.00 | 7 |
| **7** | 9.50 | 3 |
| **8** | 17.90 | 10 |
| **9** | 18.75 | 9 |
| **10** | 19.83 | 8 |
| **11** | 10.75 | 4 |
| **12** | 16.68 | 7 |
| **13** | 11.50 | 3 |
| **14** | 12.03 | 3 |
| **15** | 14.88 | 4 |
| **16** | 13.75 | 6 |
| **17** | 18.11 | 7 |
| **18** | 8.00 | 2 |
| **19** | 17.83 | 7 |

| | Delivery Time | Sorting Time |
|---|---|---|
| **20** | 21.50 | 5 |

In [3]:
```
#rename the columns
df=pd.DataFrame(data)
df
```

Out[3]:

| | Delivery Time | Sorting Time |
|---|---|---|
| **0** | 21.00 | 10 |
| **1** | 13.50 | 4 |
| **2** | 19.75 | 6 |
| **3** | 24.00 | 9 |
| **4** | 29.00 | 10 |
| **5** | 15.35 | 6 |
| **6** | 19.00 | 7 |
| **7** | 9.50 | 3 |
| **8** | 17.90 | 10 |
| **9** | 18.75 | 9 |
| **10** | 19.83 | 8 |
| **11** | 10.75 | 4 |
| **12** | 16.68 | 7 |
| **13** | 11.50 | 3 |
| **14** | 12.03 | 3 |
| **15** | 14.88 | 4 |
| **16** | 13.75 | 6 |
| **17** | 18.11 | 7 |
| **18** | 8.00 | 2 |

| | Delivery Time | Sorting Time |
|---|---|---|
| **19** | 17.83 | 7 |
| **20** | 21.50 | 5 |

In [4]:
```
df.rename(columns = {'Delivery Time' : 'deliverytime', 'Sorting Time' : 'sortingtime'}, inplace = True)
df.rename
```

Out[4]:
```
<bound method DataFrame.rename of     deliverytime  sortingtime
0           21.00           10
1           13.50            4
2           19.75            6
3           24.00            9
4           29.00           10
5           15.35            6
6           19.00            7
7            9.50            3
8           17.90           10
9           18.75            9
10          19.83            8
11          10.75            4
12          16.68            7
13          11.50            3
14          12.03            3
15          14.88            4
16          13.75            6
17          18.11            7
18           8.00            2
19          17.83            7
20          21.50            5>
```

In [5]:
```
data.head()
```

Out[5]:

| | deliverytime | sortingtime |
|---|---|---|
| **0** | 21.00 | 10 |
| **1** | 13.50 | 4 |
| **2** | 19.75 | 6 |
| **3** | 24.00 | 9 |

| | deliverytime | sortingtime |
|---|---|---|
| **4** | 29.00 | 10 |

In [6]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21 entries, 0 to 20
Data columns (total 2 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   deliverytime  21 non-null     float64
 1   sortingtime   21 non-null     int64
dtypes: float64(1), int64(1)
memory usage: 464.0 bytes
```

In [7]:
```
#correlation
data.corr()
```

Out[7]:

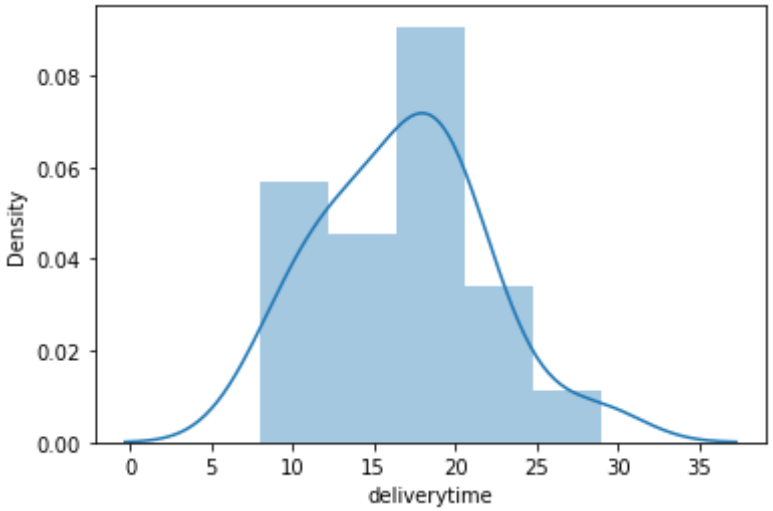
| | deliverytime | sortingtime |
|---|---|---|
| **deliverytime** | 1.000000 | 0.825997 |
| **sortingtime** | 0.825997 | 1.000000 |

In [8]:
```
#histogram
import seaborn as sn
```

In [10]:
```
sn.distplot(data['deliverytime'])
```

```
C:\Users\DELL\anaconda3\lib\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated fu
nction and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level functi
on with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

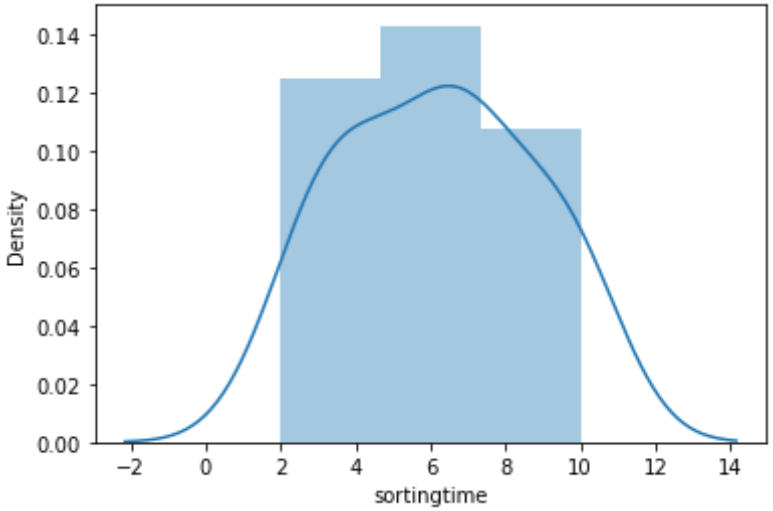
Out[10]: <AxesSubplot:xlabel='deliverytime', ylabel='Density'>




In [11]:
```
sn.distplot(data['sortingtime'])
```

```
C:\Users\DELL\anaconda3\lib\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated fu
nction and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level functi
on with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[11]: <AxesSubplot:xlabel='sortingtime', ylabel='Density'>



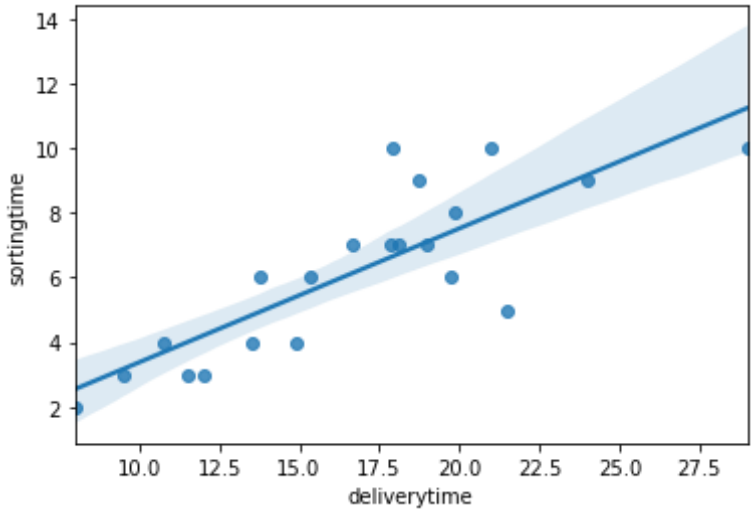
In [12]:
```
#building a model
```

In [13]:
```
#scatterplot for fitting together both
import statsmodels.formula.api as smf
```

In [14]:
```
model=smf.ols('deliverytime~sortingtime',data=data).fit()
```

In [15]:
```
sn.regplot(x="deliverytime", y="sortingtime",data=data)
```

Out[15]:
```
<AxesSubplot:xlabel='deliverytime', ylabel='sortingtime'>
```




In [16]:
```
#coefficients
model.params
```

Out[16]:
```
Intercept      6.582734
sortingtime    1.649020
dtype: float64
```

In [17]:
```
#t and p values
print(model.tvalues,'\n',model.pvalues)
```

```
Intercept      3.823349
sortingtime    6.387447
dtype: float64
```

```
Intercept      0.001147
sortingtime    0.000004
dtype: float64
```

In [19]:
```
#r squared and r squared adjecent values
print(model.rsquared,model.rsquared_adj)
```

```
0.6822714748417231 0.6655489208860244
```

In [20]:
```
model.summary()
```

Out[20]:

OLS Regression Results

| | | | |
|---|---|---|---|
| **Dep. Variable:** | deliverytime | **R-squared:** | 0.682 |
| **Model:** | OLS | **Adj. R-squared:** | 0.666 |
| **Method:** | Least Squares | **F-statistic:** | 40.80 |
| **Date:** | Fri, 16 Apr 2021 | **Prob (F-statistic):** | 3.98e-06 |
| **Time:** | 16:30:31 | **Log-Likelihood:** | -51.357 |
| **No. Observations:** | 21 | **AIC:** | 106.7 |
| **Df Residuals:** | 19 | **BIC:** | 108.8 |
| **Df Model:** | 1 | | |
| **Covariance Type:** | nonrobust | | |

| | coef | std err | t | P>\|t\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| **Intercept** | 6.5827 | 1.722 | 3.823 | 0.001 | 2.979 | 10.186 |
| **sortingtime** | 1.6490 | 0.258 | 6.387 | 0.000 | 1.109 | 2.189 |

| | | | |
|---|---|---|---|
| **Omnibus:** | 3.649 | **Durbin-Watson:** | 1.248 |
| **Prob(Omnibus):** | 0.161 | **Jarque-Bera (JB):** | 2.086 |
| **Skew:** | 0.750 | **Prob(JB):** | 0.352 |
| **Kurtosis:** | 3.367 | **Cond. No.** | 18.3 |

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.