In [1]:

```
import warnings
warnings.filterwarnings("ignore")
```

In [2]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

In [3]:

```
data = pd.read_csv("D:/Dixant/CDAC/Machine Learning/LAB GRADED/HCLTECH.csv")
```

In [4]:

```
data
```

Out[4]:

	Date	Symbol	Series	Prev Close	Open	High	Low	Last	Close	VWAP	
0	2000-01-11	HCLTECH	EQ	580.00	1550.0	1725.00	1492.00	1560.00	1554.45	1582.72	
1	2000-01-12	HCLTECH	EQ	1554.45	1560.0	1678.85	1560.00	1678.85	1678.85	1657.05	
2	2000-01-13	HCLTECH	EQ	1678.85	1790.0	1813.20	1781.00	1813.20	1813.20	1804.69	
3	2000-01-14	HCLTECH	EQ	1813.20	1958.3	1958.30	1835.00	1958.30	1958.30	1939.90	
4	2000-01-17	HCLTECH	EQ	1958.30	2115.0	2115.00	1801.65	1801.65	1801.65	1990.55	
...	...	...	...	...	...	...	...	...	...	...	
5193	2020-11-23	HCLTECH	EQ	819.25	825.0	842.00	816.25	838.50	839.20	832.35	
5194	2020-11-24	HCLTECH	EQ	839.20	843.9	857.40	835.35	841.00	840.50	847.95	8
5195	2020-11-25	HCLTECH	EQ	840.50	840.5	846.00	822.50	825.00	824.70	829.08	5
5196	2020-11-26	HCLTECH	EQ	824.70	824.1	845.00	819.60	841.20	842.05	834.43	8
5197	2020-11-27	HCLTECH	EQ	842.05	842.0	847.80	814.35	823.15	822.10	827.29	1

5198 rows × 15 columns

In [5]:

```
df=pd.DataFrame(data['Prev Close'])
```

In [6]:

```
df
```

Out[6]:

	Prev Close
0	580.00
1	1554.45
2	1678.85
3	1813.20
4	1958.30
...	...
5193	819.25
5194	839.20
5195	840.50
5196	824.70
5197	842.05

5198 rows × 1 columns

In [7]:

```
data.Timestamp = pd.to_datetime(data.Date,format='%Y-%m-%d')
df.index = data.Timestamp
```

In [8]:

```
#df = df.resample('M').mean()
```

In [9]:

```
df
```

Out[9]:

Date	Prev Close
2000-01-11	580.00
2000-01-12	1554.45
2000-01-13	1678.85
2000-01-14	1813.20
2000-01-17	1958.30
...	...
2020-11-23	819.25
2020-11-24	839.20
2020-11-25	840.50
2020-11-26	824.70
2020-11-27	842.05

5198 rows × 1 columns

In [10]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 5198 entries, 2000-01-11 to 2020-11-27
Data columns (total 1 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   Prev Close  5198 non-null   float64
dtypes: float64(1)
memory usage: 81.2 KB
```

In [11]:

```
df.describe()
```

Out[11]:

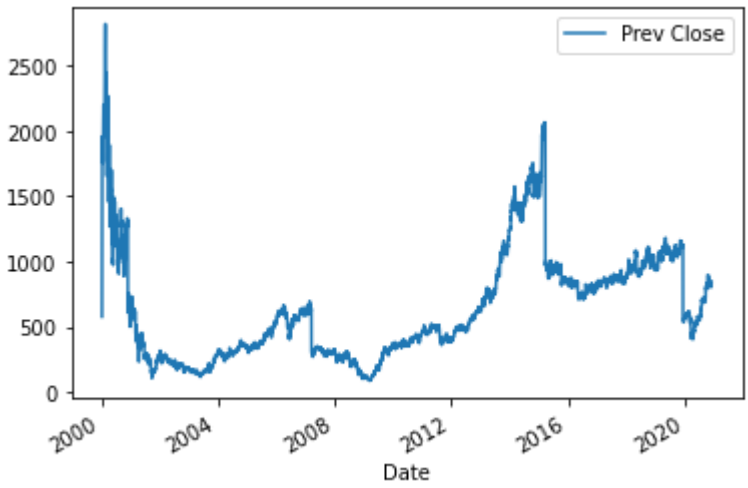
	Prev Close
count	5198.000000
mean	636.039727
std	413.306168
min	89.700000
25%	311.550000
50%	517.125000
75%	877.225000
max	2819.150000

In [12]:

```
df.plot()
```

Out[12]:

```
<AxesSubplot:xlabel='Date'>
```



In [13]:

```
from statsmodels.tsa.stattools import adfuller
```

In [14]:

```
test_result=adfuller(df['Prev Close'])
test_result
```

Out[14]:

```
(-4.023837952855129,
 0.0012907058197707806,
 33,
 5164,
 {'1%': -3.431616954601018, '5%': -2.86209986084596, '10%': -2.5670680139345
8},
 48833.8741194518)
```

In [15]:

```
df['Seasonal_Difference']=df['Prev Close']-df['Prev Close'].shift(1)
## Again test dickey fuller test
test_result=adfuller(df['Seasonal_Difference'].dropna())
test_result
```

Out[15]:

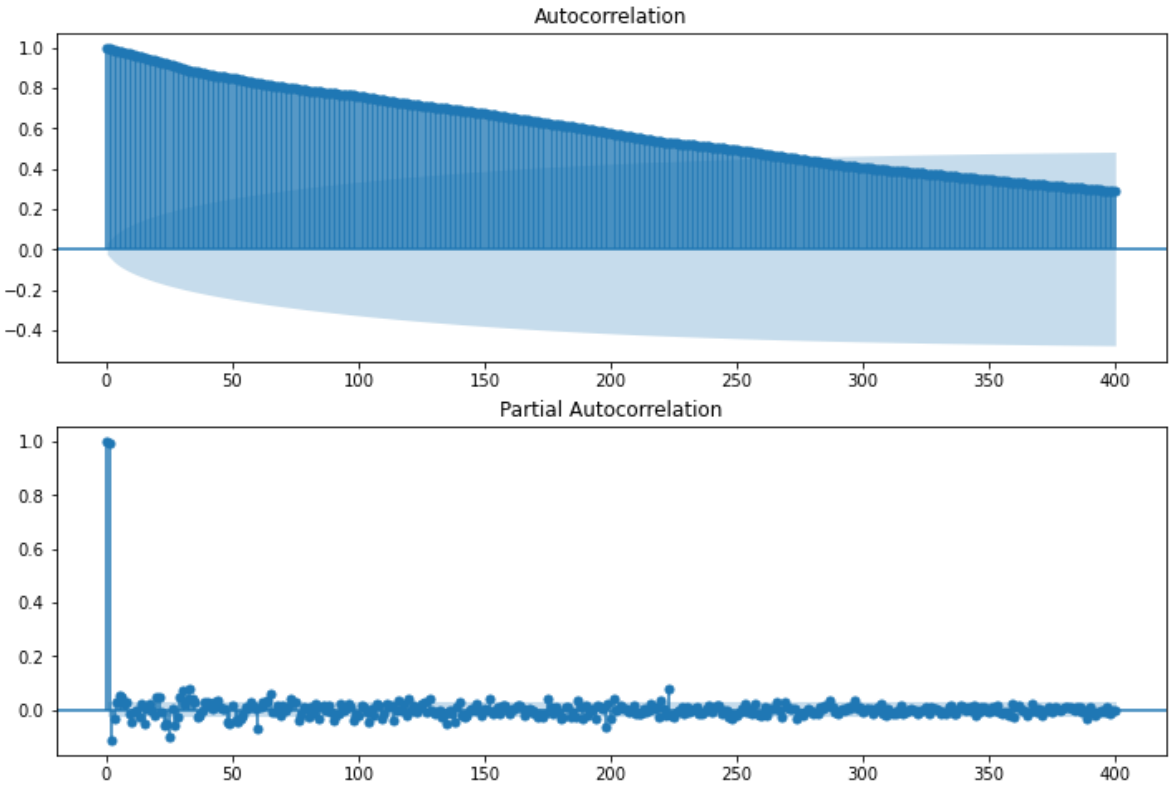
```
(-15.763872864896225,
 1.1828348703235796e-28,
 33,
 5163,
 {'1%': -3.4316172001143523,
  '5%': -2.8620999693139497,
  '10%': -2.567068071676066},
 48832.71871212932)
```

In [16]:

```
import statsmodels.api as sm
```

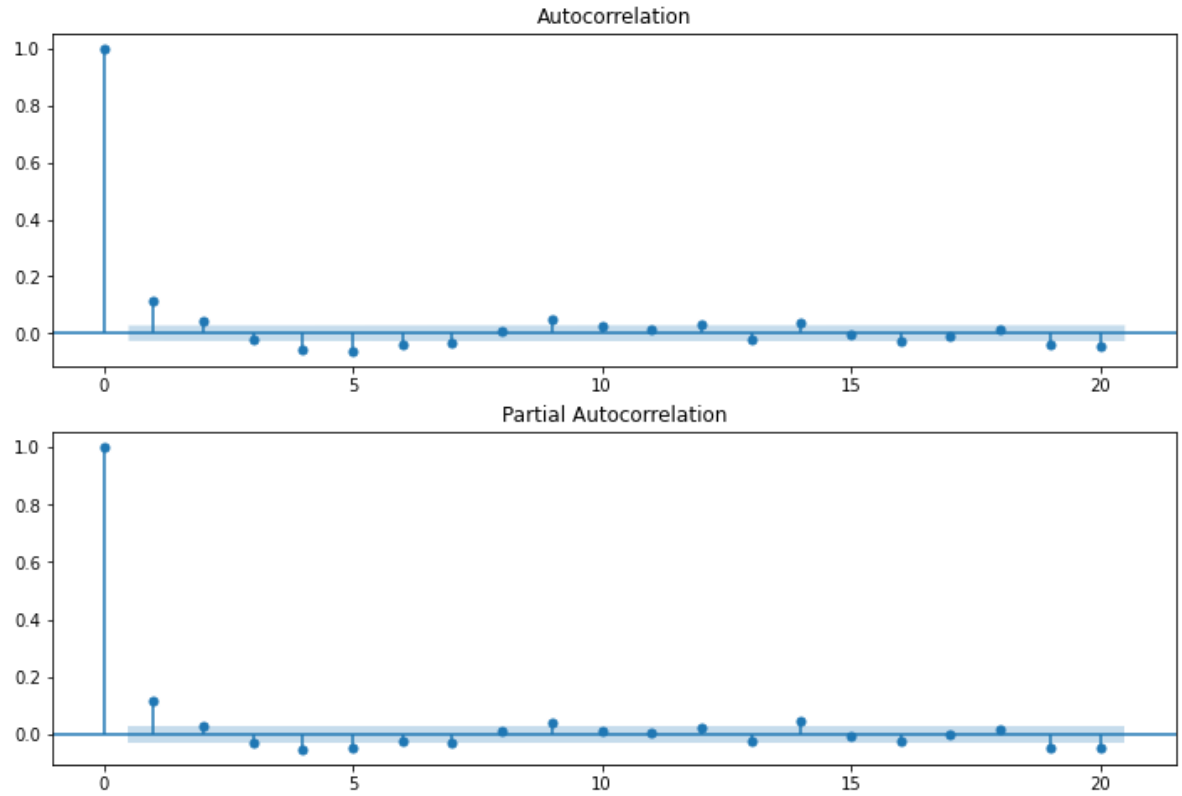
In [17]:

```
%matplotlib inline
fig = plt.figure(figsize=(12,8))
ax1 = fig.add_subplot(211)
fig = sm.graphics.tsa.plot_acf(df['Prev Close'], lags=400, ax=ax1)
ax2 = fig.add_subplot(212)
fig = sm.graphics.tsa.plot_pacf(df['Prev Close'], lags=400, ax=ax2)
```



In [18]:

```
%matplotlib inline
fig = plt.figure(figsize=(12,8))
ax1 = fig.add_subplot(211)
fig = sm.graphics.tsa.plot_acf(df['Seasonal_Difference'].dropna(), lags=20, ax=ax1)
ax2 = fig.add_subplot(212)
fig = sm.graphics.tsa.plot_pacf(df['Seasonal_Difference'].dropna(), lags=20, ax=ax2)
```



In [19]:

```
import statsmodels.api as sm
from statsmodels.tsa.arima_model import ARMA
# fit model
ARMAmodel = ARMA(df['Prev Close'], order=(1, 1))
ARmodel_fit = ARMAmodel.fit(disp=False)
```

```
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'
```

In [20]:

```
actuals = df['Prev Close'][5195:5198]
actuals
```

Out[20]:

```
Date
2020-11-25    840.50
2020-11-26    824.70
2020-11-27    842.05
Name: Prev Close, dtype: float64
```

In [21]:

```
ypredicted = ARmodel_fit.predict(5195,5197) # end point included
print(ypredicted)
```

```
Date
2020-11-25    840.816454
2020-11-26    839.701625
2020-11-27    822.084907
dtype: float64
```

In [22]:

```
from sklearn.metrics import mean_absolute_error
mae = mean_absolute_error(actuals, ypredicted)
print('MAE: %f' % mae)
```

```
MAE: 11.761057
```

In [23]:

```
import itertools
i = j = range(0, 4)
ij = itertools.product(i,j)
for parameters in ij:
    try:
        mod = ARMA(df['Prev Close'],order=parameters)
        results = mod.fit()
        ypredicted = results.predict(5195,5197) # end point included
        mae = mean_absolute_error(actuals, ypredicted)
        print('ARMA{} - MAE:{}'.format(parameters, mae))
        #print('ARMA{} - AIC:{}'.format(parameters, results.aic))
    except:
        continue
```

```
ARMA(0, 0) - MAE:199.71027318199322

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARMA(0, 1) - MAE:99.9363831281892
ARMA(1, 0) - MAE:11.669738965977672

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARMA(1, 1) - MAE:11.76105728366872

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARMA(1, 2) - MAE:12.615716369636175

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARMA(1, 3) - MAE:12.623903317085668
```

```
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARMA(2, 0) - MAE:12.090466547247464

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARMA(2, 1) - MAE:12.294889797515225

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARMA(2, 2) - MAE:12.616296311534446

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARMA(2, 3) - MAE:12.732819519446062

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARMA(3, 0) - MAE:12.453068815368132

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\base\model.py:547: He
ssianInversionWarning: Inverting hessian failed, no bse or cov_params availa
ble
  warnings.warn('Inverting hessian failed, no bse or cov_params '
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\base\model.py:566: Co
nvergenceWarning: Maximum Likelihood optimization failed to converge. Check
mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARMA(3, 1) - MAE:12.035107695709598

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\base\model.py:566: Co
nvergenceWarning: Maximum Likelihood optimization failed to converge. Check
mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'
```

```
ARMA(3, 2) - MAE:12.845047704618082
```

In [24]:

```
ARMAmodel = ARMA(df['Prev Close'], order=(1, 0))
ARmodel_fit = ARMAmodel.fit()
ypredicted = ARmodel_fit.predict(5195,5197)
print(ypredicted)
mae = mean_absolute_error(actuals, ypredicted)
print('MAE: %f' % mae)
print(ARmodel_fit.aic)
```

```
Date
2020-11-25    838.589881
2020-11-26    839.885693
2020-11-27    824.136595
dtype: float64
MAE: 11.669739
50899.88642152843

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'
```

In [25]:

```
actuals
```

Out[25]:

```
Date
2020-11-25    840.50
2020-11-26    824.70
2020-11-27    842.05
Name: Prev Close, dtype: float64
```

In [26]:

```
# make prediction
ypredicted = ARmodel_fit.predict(len(df), len(df)+3,typ='levels')
print(ypredicted)
```

```
5198    841.430700
5199    840.813394
5200    840.198078
5201    839.584744
dtype: float64

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:376: ValueWarning: No supported index is available. Prediction results wil
l be given with an integer index beginning at `start`.
  warnings.warn('No supported index is available.'
```

In [27]:

```
from statsmodels.tsa.arima_model import ARIMA
```

In [28]:

```
ARIMAmodel = ARIMA(df['Prev Close'], order=(1, 1, 1))
ARIMA_model_fit = ARIMAmodel.fit()

ypredicted = ARIMA_model_fit.predict(len(df)-3, len(df)-1, typ='levels')
print(ypredicted)
```

```
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

Date
2020-11-25    842.290151
2020-11-26    841.201231
2020-11-27    822.691851
dtype: float64
```

In [29]:

```
mae = mean_absolute_error(actuals, ypredicted)
print('MAE: %f' % mae)
print(ARIMA_model_fit.aic)
```

```
MAE: 12.549844
50809.162094743864
```

In [30]:

```
# make prediction
ypredicted = ARIMA_model_fit.predict(len(df), len(df)+3, typ='levels')
print(ypredicted)
```

```
5197    844.202722
5198    844.881953
5199    845.139224
5200    845.275658
dtype: float64

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:376: ValueWarning: No supported index is available. Prediction results wil
l be given with an integer index beginning at `start`.
  warnings.warn('No supported index is available.'
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:376: ValueWarning: No supported index is available. Prediction results wil
l be given with an integer index beginning at `start`.
  warnings.warn('No supported index is available.'
```

In [31]:

```
import itertools
p= d = q = range(0, 4)
pdq = itertools.product(p,d,q)
for parameters in pdq:
    try:
        ARIMAmodel = ARIMA(df['Prev Close'], order=parameters)
        results = ARIMAmodel.fit()
        ypredicted = results.predict(5195,5197) # end point included
        mae = mean_absolute_error(actuals, ypredicted)
        print('ARIMA{} - MAE:{}'.format(parameters, mae))
    #print('ARMA{} - AIC:{}'.format(parameters, results.aic))
    except:
        continue
```

```
ated frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARIMA(2, 0, 1) - MAE:12.294889797515225

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.
py:581: ValueWarning: A date index has been provided, but it has no associ
ated frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARIMA(2, 0, 2) - MAE:12.616296311534446



C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.
py:581: ValueWarning: A date index has been provided, but it has no associ
ated frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

ARIMA(2, 0, 3) - MAE:12.732819519446062
```

In [32]:

```
ARIMAmodel = ARIMA(df['Prev Close'], order=(1, 0, 0))
ARIMA_model_fit = ARIMAmodel.fit()

ypredicted = ARIMA_model_fit.predict(len(df)-3, len(df)-1, typ='levels')
print(ypredicted)
```

```
C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:581: ValueWarning: A date index has been provided, but it has no associate
d frequency information and so will be ignored when e.g. forecasting.
  warnings.warn('A date index has been provided, but it has no'

Date
2020-11-25    838.589881
2020-11-26    839.885693
2020-11-27    824.136595
dtype: float64
```

In [33]:

```
mae = mean_absolute_error(actuals, ypredicted)
print('MAE: %f' % mae)
print(ARIMA_model_fit.aic)
```

```
MAE: 11.669739
50899.88642152843
```

In [34]:

```
# make prediction
ypredicted = ARIMA_model_fit.predict(len(df), len(df)+3, typ='levels')
print(ypredicted)
```

```
5198    841.430700
5199    840.813394
5200    840.198078
5201    839.584744
dtype: float64

C:\Users\divya\anaconda3\lib\site-packages\statsmodels\tsa\base\tsa_model.p
y:376: ValueWarning: No supported index is available. Prediction results wil
l be given with an integer index beginning at `start`.
  warnings.warn('No supported index is available.'
```

In [ ]: