SUYASH PRATAP SINGH(181B226)

TASKS:-

Download the csv file for time series data.

1. Pre-process the data to check about any missing data.
2. Plot the time series to visualise it.
3. Identify the trend, seasonality and remainder parts of the time series.
4. Select the forecasting model and identify the parameters for that model.
5. Forecast the series for next 20 periods.

In [1]:

```
import warnings
import itertools
import numpy as np
import matplotlib.pyplot as plt
warnings.filterwarnings("ignore")
plt.style.use('fivethirtyeight')
import pandas as pd
import statsmodels.api as sm
import matplotlib
matplotlib.rcParams['axes.labelsize'] = 14
matplotlib.rcParams['xtick.labelsize'] = 12
matplotlib.rcParams['ytick.labelsize'] = 12
matplotlib.rcParams['text.color'] = 'k'
```

In [2]:

```
df=pd.read_csv(r'C:\Users\Admin\Downloads\Suyash Pratap singh.csv')
```

In [3]:

```
df.head()
```

Out[3]:

	Date	Close	Volume
0	2015-01-02	38.0061	6906098
1	2015-01-05	37.2781	11623796
2	2015-01-06	36.9748	7664340
3	2015-01-07	37.8848	9732554
4	2015-01-08	38.4961	13170548

In [4]:

```
df.sample(5)
```

Out[4]:

	Date	Close	Volume
611	2017-06-07	61.4568	8364994
59	2015-03-30	44.9551	6830270
165	2015-08-28	52.4219	7584826
582	2017-04-26	59.3348	8525419
71	2015-04-16	45.1940	5312499

In [5]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1006 entries, 0 to 1005
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   Date    1006 non-null   object 
 1   Close   1006 non-null   float64
 2   Volume  1006 non-null   int64  
dtypes: float64(1), int64(1), object(1)
memory usage: 23.7+ KB
```

In [6]:

```
df.shape
```

Out[6]:

```
(1006, 3)
```

In [7]:

```
df.isnull().sum()
```

Out[7]:

```
Date      0
Close     0
Volume    0
dtype: int64
```

In [8]:

```
df.describe()
```

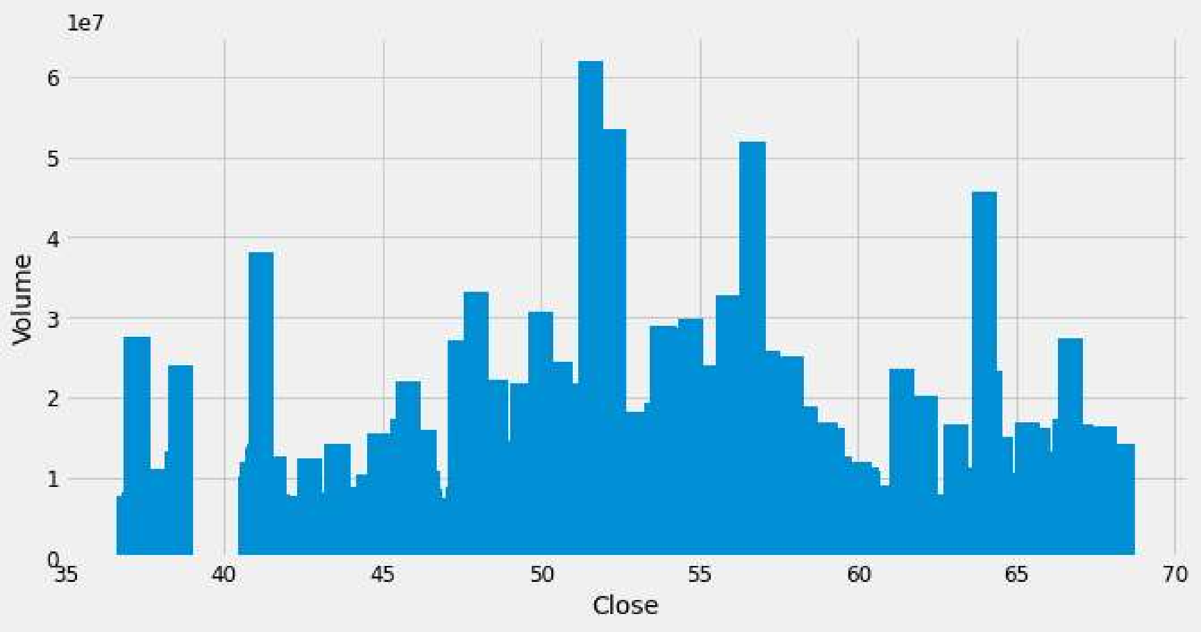
Out[8]:

	Close	Volume
count	1006.000000	1.006000e+03
mean	54.070133	9.617402e+06
std	4.849600	4.986199e+06
min	36.974800	2.215418e+06
25%	52.170075	6.828217e+06
50%	54.631600	8.338076e+06
75%	56.603475	1.091071e+07
max	68.353000	6.209107e+07

In [9]:
```
df=df.set_index('Date')
df.index=pd.to_datetime(df.index)
df.index
```

Out[9]:
```
DatetimeIndex(['2015-01-02', '2015-01-05', '2015-01-06', '2015-01-07',
               '2015-01-08', '2015-01-09', '2015-01-12', '2015-01-13',
               '2015-01-14', '2015-01-15',
               ...
               '2018-12-17', '2018-12-18', '2018-12-19', '2018-12-20',
               '2018-12-21', '2018-12-24', '2018-12-26', '2018-12-27',
               '2018-12-28', '2018-12-31'],
              dtype='datetime64[ns]', name='Date', length=1006, freq=None)
```

In [10]:
```
fig = plt.figure(figsize = (10, 5))
# creating the bar plot
plt.bar(df['Close'], df['Volume'])
plt.xlabel("Close")
plt.ylabel("Volume")
plt.show()
```



In [11]:
```
y= df['Close'].resample('MS').mean()
```

In [12]:

```
y.index
```

Out[12]:

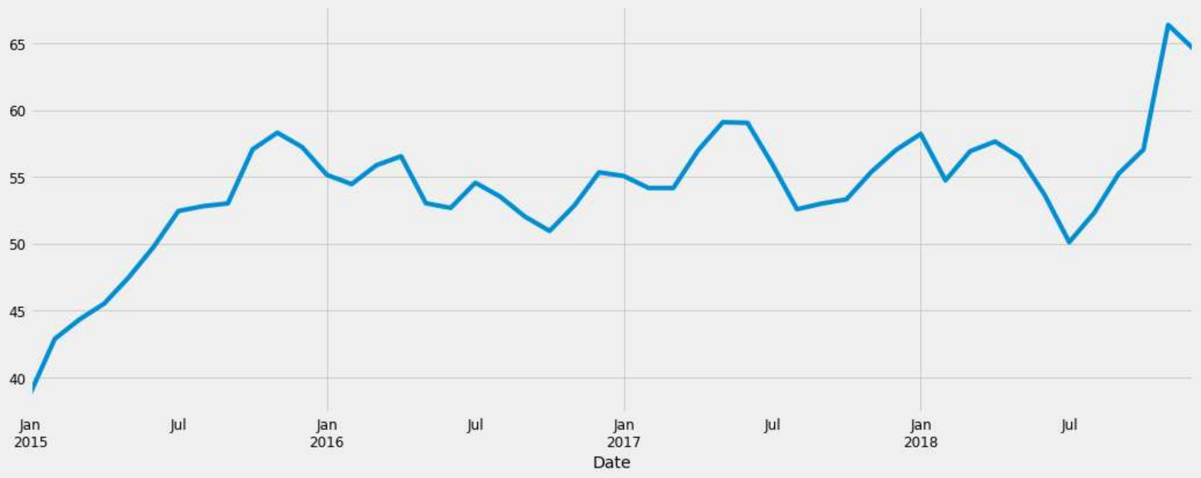
```
DatetimeIndex(['2015-01-01', '2015-02-01', '2015-03-01', '2015-04-01',
               '2015-05-01', '2015-06-01', '2015-07-01', '2015-08-01',
               '2015-09-01', '2015-10-01', '2015-11-01', '2015-12-01',
               '2016-01-01', '2016-02-01', '2016-03-01', '2016-04-01',
               '2016-05-01', '2016-06-01', '2016-07-01', '2016-08-01',
               '2016-09-01', '2016-10-01', '2016-11-01', '2016-12-01',
               '2017-01-01', '2017-02-01', '2017-03-01', '2017-04-01',
               '2017-05-01', '2017-06-01', '2017-07-01', '2017-08-01',
               '2017-09-01', '2017-10-01', '2017-11-01', '2017-12-01',
               '2018-01-01', '2018-02-01', '2018-03-01', '2018-04-01',
               '2018-05-01', '2018-06-01', '2018-07-01', '2018-08-01',
               '2018-09-01', '2018-10-01', '2018-11-01', '2018-12-01'],
              dtype='datetime64[ns]', name='Date', freq='MS')
```

In [13]:
```
y['2015':]
```

Out[13]:
```
Date
2015-01-01    38.729470
2015-02-01    42.862432
2015-03-01    44.321836
2015-04-01    45.508914
2015-05-01    47.488650
2015-06-01    49.761627
2015-07-01    52.437959
2015-08-01    52.797976
2015-09-01    53.006186
2015-10-01    57.051373
2015-11-01    58.296355
2015-12-01    57.219514
2016-01-01    55.138963
2016-02-01    54.446810
2016-03-01    55.858968
2016-04-01    56.531200
2016-05-01    53.016571
2016-06-01    52.669300
2016-07-01    54.551905
2016-08-01    53.518291
2016-09-01    52.022771
2016-10-01    50.945319
2016-11-01    52.845452
2016-12-01    55.331981
2017-01-01    55.057330
2017-02-01    54.157374
2017-03-01    54.155570
2017-04-01    56.968300
2017-05-01    59.081968
2017-06-01    59.031950
2017-07-01    55.933905
2017-08-01    52.567952
2017-09-01    52.995195
2017-10-01    53.310614
2017-11-01    55.359462
2017-12-01    57.006955
2018-01-01    58.205433
2018-02-01    54.729784
2018-03-01    56.913367
2018-04-01    57.635371
2018-05-01    56.482395
2018-06-01    53.663433
2018-07-01    50.105995
2018-08-01    52.279430
2018-09-01    55.234000
2018-10-01    57.035104
2018-11-01    66.363652
2018-12-01    64.609474
Freq: MS, Name: Close, dtype: float64
```

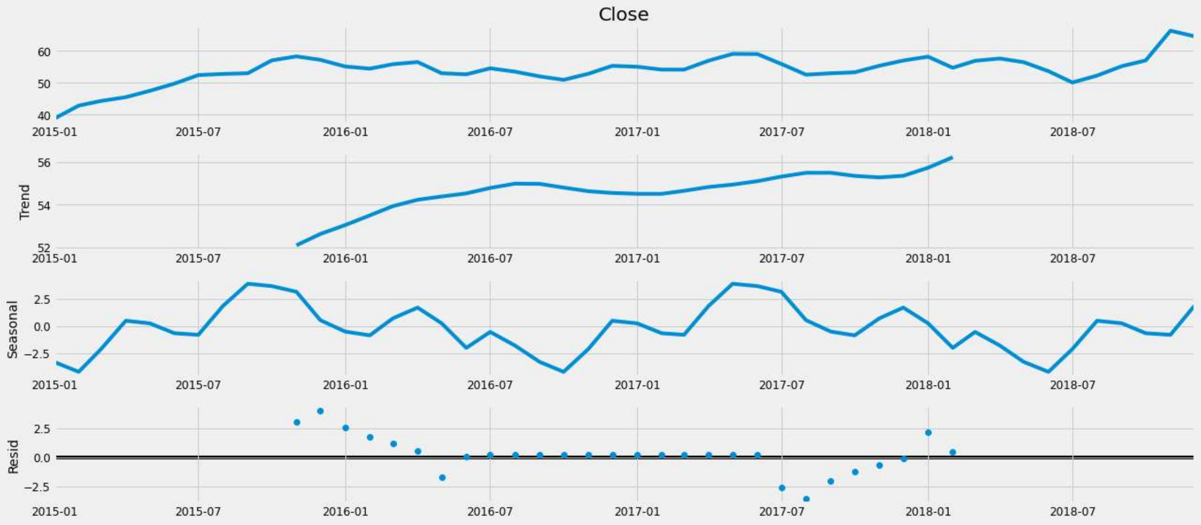
In [14]:

```
y.plot(figsize=(16,6))
plt.show()
```



In [15]:

```
from pylab import rcParams
rcParams['figure.figsize']=18,8
decomposition=sm.tsa.seasonal_decompose(y,freq=20,model='additive')
fig=decomposition.plot()
plt.show()
```



In [16]:

```
p = d = q = range(0, 2)
pdq = list(itertools.product(p, d, q))
seasonal_pdq = [(x[0], x[1], x[2], 12) for x in list(itertools.product(p, d, q
))]
print('Examples of parameter combinations for Seasonal ARIMA...')
print('SARIMAX: {} x {}'.format(pdq[1], seasonal_pdq[1]))
print('SARIMAX: {} x {}'.format(pdq[1], seasonal_pdq[2]))
print('SARIMAX: {} x {}'.format(pdq[2], seasonal_pdq[3]))
print('SARIMAX: {} x {}'.format(pdq[2], seasonal_pdq[4]))
```

```
Examples of parameter combinations for Seasonal ARIMA...
SARIMAX: (0, 0, 1) x (0, 0, 1, 12)
SARIMAX: (0, 0, 1) x (0, 1, 0, 12)
SARIMAX: (0, 1, 0) x (0, 1, 1, 12)
SARIMAX: (0, 1, 0) x (1, 0, 0, 12)
```

In [17]:

```
for param in pdq:
    for param_seasonal in seasonal_pdq:
        try:
            mod = sm.tsa.statespace.SARIMAX(y,
                                            order=param,
                                            seasonal_order=param_seasonal,
                                            enforce_stationarity=False,
                                            enforce_invertibility=False)
            results = mod.fit()
            print('ARIMA{}x{}12 - AIC:{}'.format(param, param_seasonal, result
s.aic))
        except:
            continue
```

```
ARIMA(0, 0, 0)x(0, 0, 0, 12)12 - AIC:511.3098748667409
ARIMA(0, 0, 0)x(0, 0, 1, 12)12 - AIC:1423.7357075699124
ARIMA(0, 0, 0)x(0, 1, 0, 12)12 - AIC:214.5397316938652
ARIMA(0, 0, 0)x(0, 1, 1, 12)12 - AIC:134.04872175010894
ARIMA(0, 0, 0)x(1, 0, 0, 12)12 - AIC:226.4382115730092
```

```
c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "
```

```
ARIMA(0, 0, 0)x(1, 0, 1, 12)12 - AIC:1228.3934050128803
ARIMA(0, 0, 0)x(1, 1, 0, 12)12 - AIC:138.68478106595083
ARIMA(0, 0, 0)x(1, 1, 1, 12)12 - AIC:135.99504718768947
ARIMA(0, 0, 1)x(0, 0, 0, 12)12 - AIC:443.8763087626294
```

```
c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "
```

```
ARIMA(0, 0, 1)x(0, 0, 1, 12)12 - AIC:2443.3656553527267
ARIMA(0, 0, 1)x(0, 1, 0, 12)12 - AIC:177.75111649427654
ARIMA(0, 0, 1)x(0, 1, 1, 12)12 - AIC:115.94385137454779
```

```
c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "
c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "
```

```
ARIMA(0, 0, 1)x(1, 0, 0, 12)12 - AIC:228.86737092213409
ARIMA(0, 0, 1)x(1, 0, 1, 12)12 - AIC:2195.431477427645
ARIMA(0, 0, 1)x(1, 1, 0, 12)12 - AIC:123.97845604228267
ARIMA(0, 0, 1)x(1, 1, 1, 12)12 - AIC:117.94385218127003
ARIMA(0, 1, 0)x(0, 0, 0, 12)12 - AIC:213.176667717877
ARIMA(0, 1, 0)x(0, 0, 1, 12)12 - AIC:1078.6535190110055
ARIMA(0, 1, 0)x(0, 1, 0, 12)12 - AIC:172.60742568216972
ARIMA(0, 1, 0)x(0, 1, 1, 12)12 - AIC:114.24881312049962
ARIMA(0, 1, 0)x(1, 0, 0, 12)12 - AIC:167.68051617356008
```

```
c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "
```

```
ARIMA(0, 1, 0)x(1, 0, 1, 12)12 - AIC:1328.0688001472513
ARIMA(0, 1, 0)x(1, 1, 0, 12)12 - AIC:118.430371728462
ARIMA(0, 1, 0)x(1, 1, 1, 12)12 - AIC:115.9971020082245
ARIMA(0, 1, 1)x(0, 0, 0, 12)12 - AIC:208.83045168602052
ARIMA(0, 1, 1)x(0, 0, 1, 12)12 - AIC:1135.097762467897
ARIMA(0, 1, 1)x(0, 1, 0, 12)12 - AIC:168.41059644556248
ARIMA(0, 1, 1)x(0, 1, 1, 12)12 - AIC:111.40884119948561
ARIMA(0, 1, 1)x(1, 0, 0, 12)12 - AIC:168.95742140249163
```

```
c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "

ARIMA(0, 1, 1)x(1, 0, 1, 12)12 - AIC:1124.8277132570952
ARIMA(0, 1, 1)x(1, 1, 0, 12)12 - AIC:120.20934968473047
ARIMA(0, 1, 1)x(1, 1, 1, 12)12 - AIC:113.58370873097263
ARIMA(1, 0, 0)x(0, 0, 0, 12)12 - AIC:219.92092271718795

c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "
c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "

ARIMA(1, 0, 0)x(0, 0, 1, 12)12 - AIC:1509.9225507471965
ARIMA(1, 0, 0)x(0, 1, 0, 12)12 - AIC:173.36375409127237
ARIMA(1, 0, 0)x(0, 1, 1, 12)12 - AIC:118.2329670863586
ARIMA(1, 0, 0)x(1, 0, 0, 12)12 - AIC:175.26156956913536

c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "

ARIMA(1, 0, 0)x(1, 0, 1, 12)12 - AIC:1320.0223803370407
ARIMA(1, 0, 0)x(1, 1, 0, 12)12 - AIC:118.552160325973
ARIMA(1, 0, 0)x(1, 1, 1, 12)12 - AIC:120.55058056861057
ARIMA(1, 0, 1)x(0, 0, 0, 12)12 - AIC:214.05196311379967

c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "

ARIMA(1, 0, 1)x(0, 0, 1, 12)12 - AIC:1270.3440795128126
ARIMA(1, 0, 1)x(0, 1, 0, 12)12 - AIC:167.08900780355904
ARIMA(1, 0, 1)x(0, 1, 1, 12)12 - AIC:114.60189985029386
ARIMA(1, 0, 1)x(1, 0, 0, 12)12 - AIC:172.56349593793232

c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "

ARIMA(1, 0, 1)x(1, 0, 1, 12)12 - AIC:1085.6863873305585
ARIMA(1, 0, 1)x(1, 1, 0, 12)12 - AIC:118.696907624078
ARIMA(1, 0, 1)x(1, 1, 1, 12)12 - AIC:116.681844709811
ARIMA(1, 1, 0)x(0, 0, 0, 12)12 - AIC:212.93469834117843

c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "
ARIMA(1, 1, 0)x(0, 0, 1, 12)12 - AIC:953.4455494613746
ARIMA(1, 1, 0)x(0, 1, 0, 12)12 - AIC:173.30714214765834
ARIMA(1, 1, 0)x(0, 1, 1, 12)12 - AIC:115.65281356976386
ARIMA(1, 1, 0)x(1, 0, 0, 12)12 - AIC:165.18885726746103

c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "

ARIMA(1, 1, 0)x(1, 0, 1, 12)12 - AIC:942.117953283741
ARIMA(1, 1, 0)x(1, 1, 0, 12)12 - AIC:115.82169338755774
ARIMA(1, 1, 0)x(1, 1, 1, 12)12 - AIC:117.79966515211034
ARIMA(1, 1, 1)x(0, 0, 0, 12)12 - AIC:210.54003110595292

c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "

ARIMA(1, 1, 1)x(0, 0, 1, 12)12 - AIC:1339.9553062410987
ARIMA(1, 1, 1)x(0, 1, 0, 12)12 - AIC:169.44574423827862
ARIMA(1, 1, 1)x(0, 1, 1, 12)12 - AIC:113.3576106686453
ARIMA(1, 1, 1)x(1, 0, 0, 12)12 - AIC:166.86504360435723
ARIMA(1, 1, 1)x(1, 0, 1, 12)12 - AIC:1329.6658837081486
ARIMA(1, 1, 1)x(1, 1, 0, 12)12 - AIC:117.6047449878331
ARIMA(1, 1, 1)x(1, 1, 1, 12)12 - AIC:115.41676698843274

c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "
```

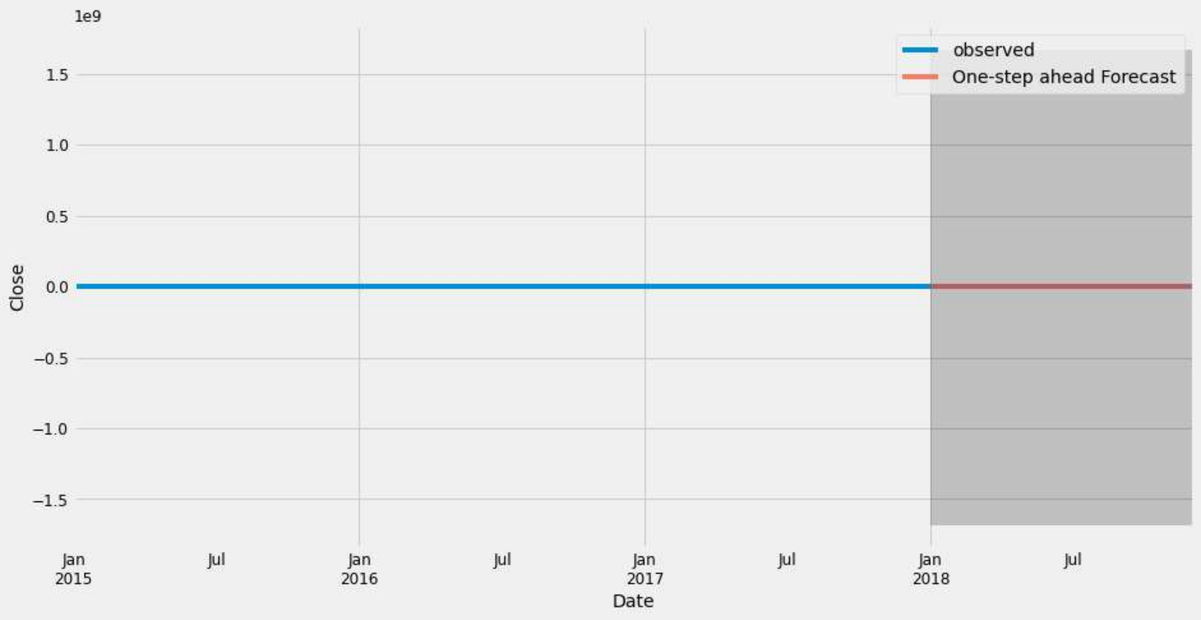
In [18]:

```
mod = sm.tsa.statespace.SARIMAX(y,
                                order=(1, 0, 0),
                                seasonal_order=(0, 0, 1, 12),
                                enforce_stationarity=False,
                                enforce_invertibility=False)
results = mod.fit()
```

```
c:\users\admin\appdata\local\programs\python\python38\lib\site-packages\stats
models\base\model.py:566: ConvergenceWarning: Maximum Likelihood optimization
failed to converge. Check mle_retvals
  warnings.warn("Maximum Likelihood optimization failed to "
```

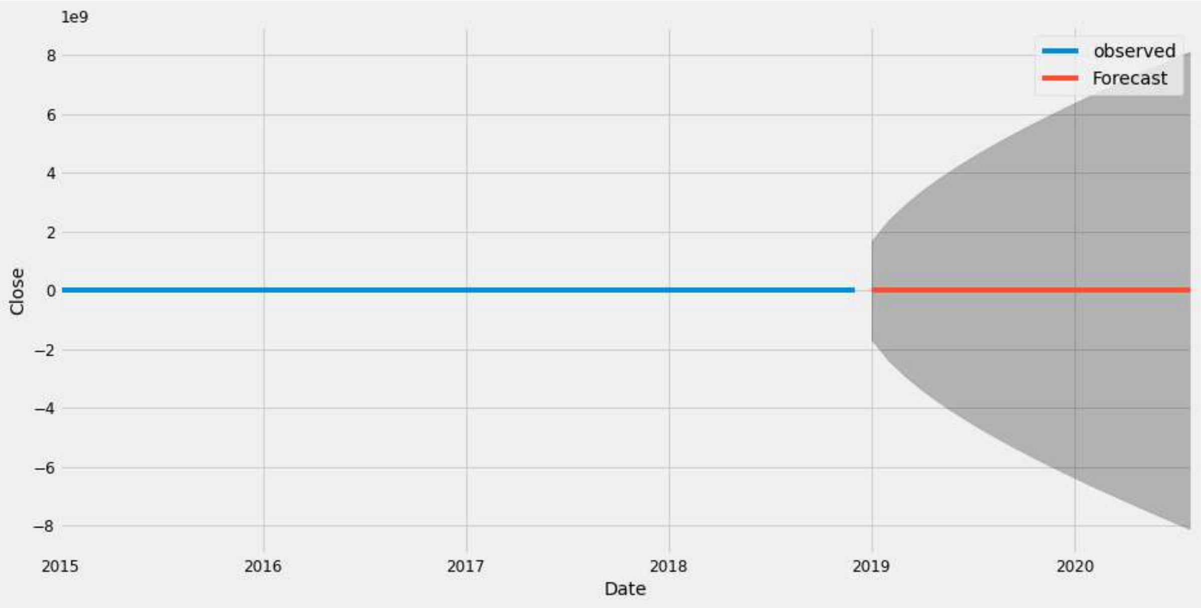
In [19]:

```
pred = results.get_prediction(start=pd.to_datetime('2018-01-01'), dynamic=False)
pred_ci = pred.conf_int()
ax = y['2015':].plot(label='observed')
pred.predicted_mean.plot(ax=ax, label='One-step ahead Forecast', alpha=.7, figsize=(14, 7))
ax.fill_between(pred_ci.index,
                pred_ci.iloc[:, 0],
                pred_ci.iloc[:, 1], color='k', alpha=.2)
ax.set_xlabel('Date')
ax.set_ylabel('Close')
plt.legend()
plt.show()
```



In [20]:

```
import matplotlib.pyplot as plt
pred_uc = results.get_forecast(steps=20)
pred_ci = pred_uc.conf_int()
ax = y.plot(label='observed', figsize=(14, 7))
pred_uc.predicted_mean.plot(ax=ax, label='Forecast')
ax.fill_between(pred_ci.index,
                pred_ci.iloc[:, 0],
                pred_ci.iloc[:, 1], color='k', alpha=.25)
ax.set_xlabel('Date')
ax.set_ylabel('Close')
plt.legend()
plt.show()
```



THANK YOU