In [81]:
```
import numpy as np
import matplotlib
import matplotlib.pyplot as plt
import pandas as pd
from pandas import read_csv
import math
import warnings
warnings.filterwarnings("ignore")
import seaborn as sns
import statsmodels.api as sm

from sklearn.preprocessing import MinMaxScaler
from statsmodels.tools.eval_measures import rmse
from sklearn.metrics import mean_squared_error
from sklearn.metrics import mean_absolute_error
from keras.callbacks import EarlyStopping
from keras.layers import ConvLSTM2D
```

In [82]:
```
plt.rcParams["figure.figsize"] = (15,4)
matplotlib.rcParams['axes.labelsize'] = 14
matplotlib.rcParams['xtick.labelsize'] = 12
matplotlib.rcParams['ytick.labelsize'] = 12
matplotlib.rcParams['text.color'] = 'k'
```

In [83]:
```
data = pd.read_excel(r'POCM_HISTORICAL.xlsx')
df1 = data[["Checkout_Date","Order_Quantity"]]
```

In [84]:
```
df1
```

Out[84]:

	Checkout_Date	Order_Quantity
0	2018-12-04	1
1	2018-12-04	1
2	2018-12-05	5
3	2018-12-05	10
4	2018-12-05	20

	Checkout_Date	Order_Quantity
...	...	...
160572	2020-05-15	5
160573	2020-05-15	20
160574	2020-05-15	2
160575	2020-05-15	3
160576	2020-05-15	2

160577 rows × 2 columns

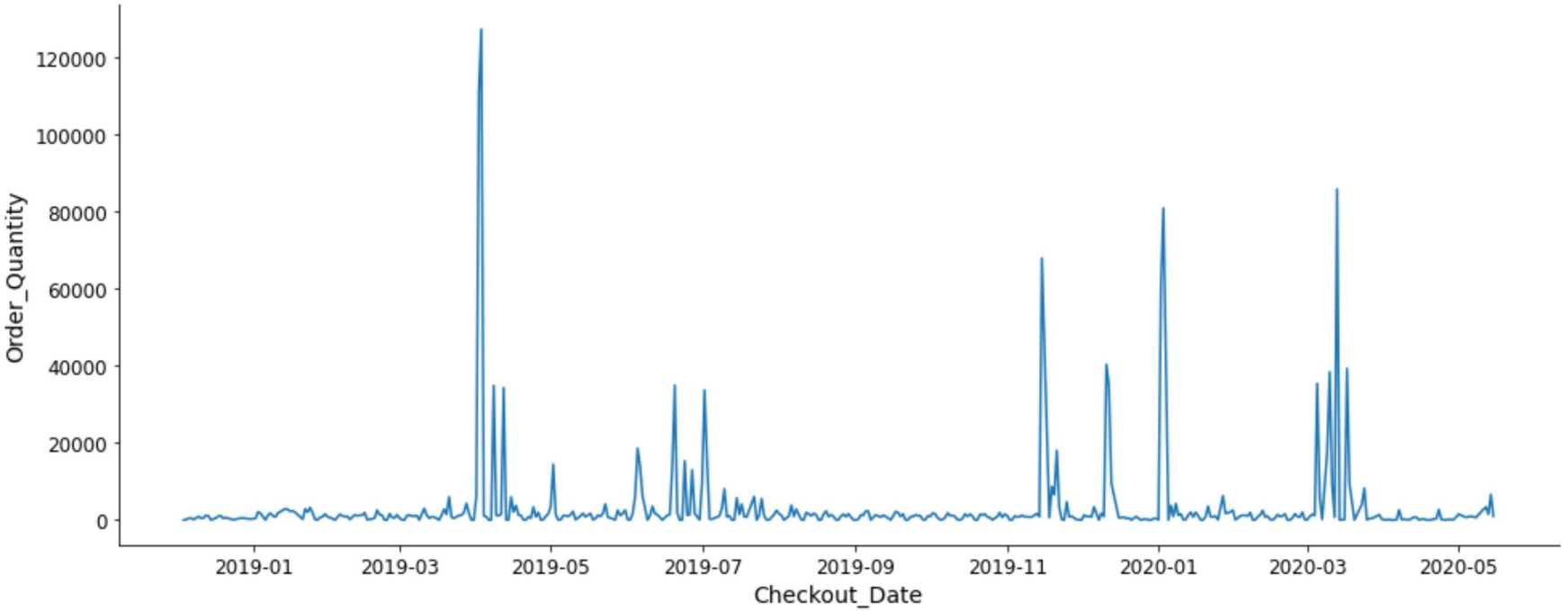
In [85]:
```
df = df1.groupby(["Checkout_Date"])['Order_Quantity'].sum().reset_index()
df.tail()
```

Out[85]:

	Checkout_Date	Order_Quantity
465	2020-05-11	2749
466	2020-05-12	3366
467	2020-05-13	1491
468	2020-05-14	6631
469	2020-05-15	1018

In [86]:
```
sns.relplot(x = 'Checkout_Date', y = 'Order_Quantity', data = df, kind = 'line', height = 5.5, aspect = 2.5)
```

Out[86]:
```
<seaborn.axisgrid.FacetGrid at 0x7fd20dffe990>
```



In [87]:

```
#Stationarity checking

from statsmodels.tsa.stattools import adfuller
#Ho: It is non stationary
#H1: It is stationary

def adf_test(Order_Quantity):
    result=adfuller(Order_Quantity)
    labels = ['ADF stats','p-value','#Lags Used','Num of Observations']
    for value,label in zip(result,labels):
        print(label+' : '+str(value) )
    if result[1] <= 0.05:
        print("The test rejects the null hypothesis and is stationary")
    else:
        print("The test validates the null hypothesis and is non-stationary ")

adf_test(df['Order_Quantity'])
```

```
ADF stats : -10.147197874904466
p-value : 8.127120756465757e-18
#Lags Used : 2
```

```
Num of Observations : 467
The test rejects the null hypothesis and is stationary
```

In [88]:

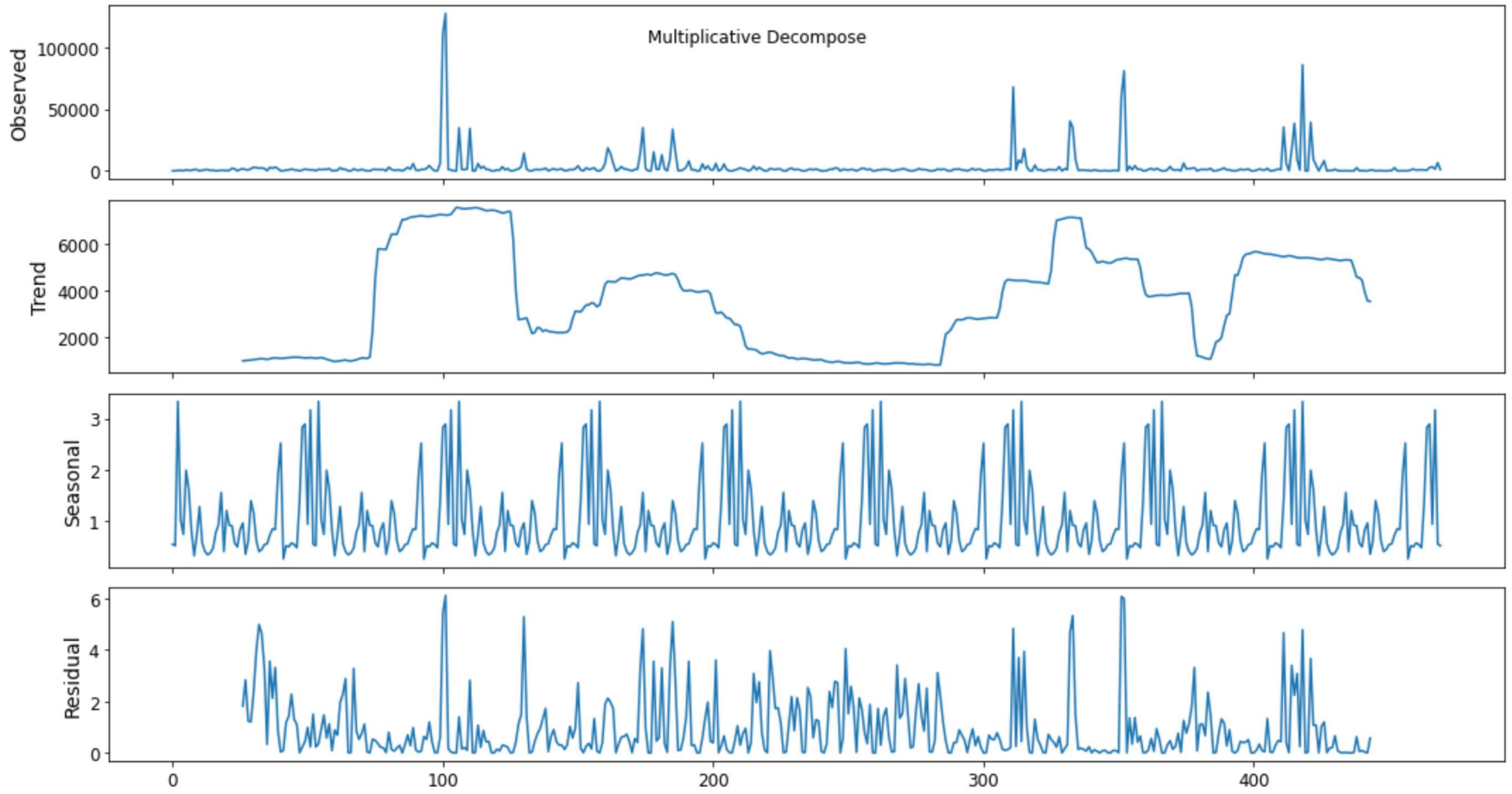
```
from statsmodels.tsa.seasonal import seasonal_decompose
#Additive decomposition
add_result = seasonal_decompose(df['Order_Quantity'],model='additive',freq = 52) # annual=1,Quaterly=4,monthly=12,weekly=

#Multiplicative decompostion
mul_result = seasonal_decompose(df['Order_Quantity'], model='multiplicative',freq=52)
```

In [89]:

```
# Plot
plt.rcParams.update({'figure.figsize': (15,8)})
mul_result.plot().suptitle('\nMultiplicative Decompose', fontsize=12)
```

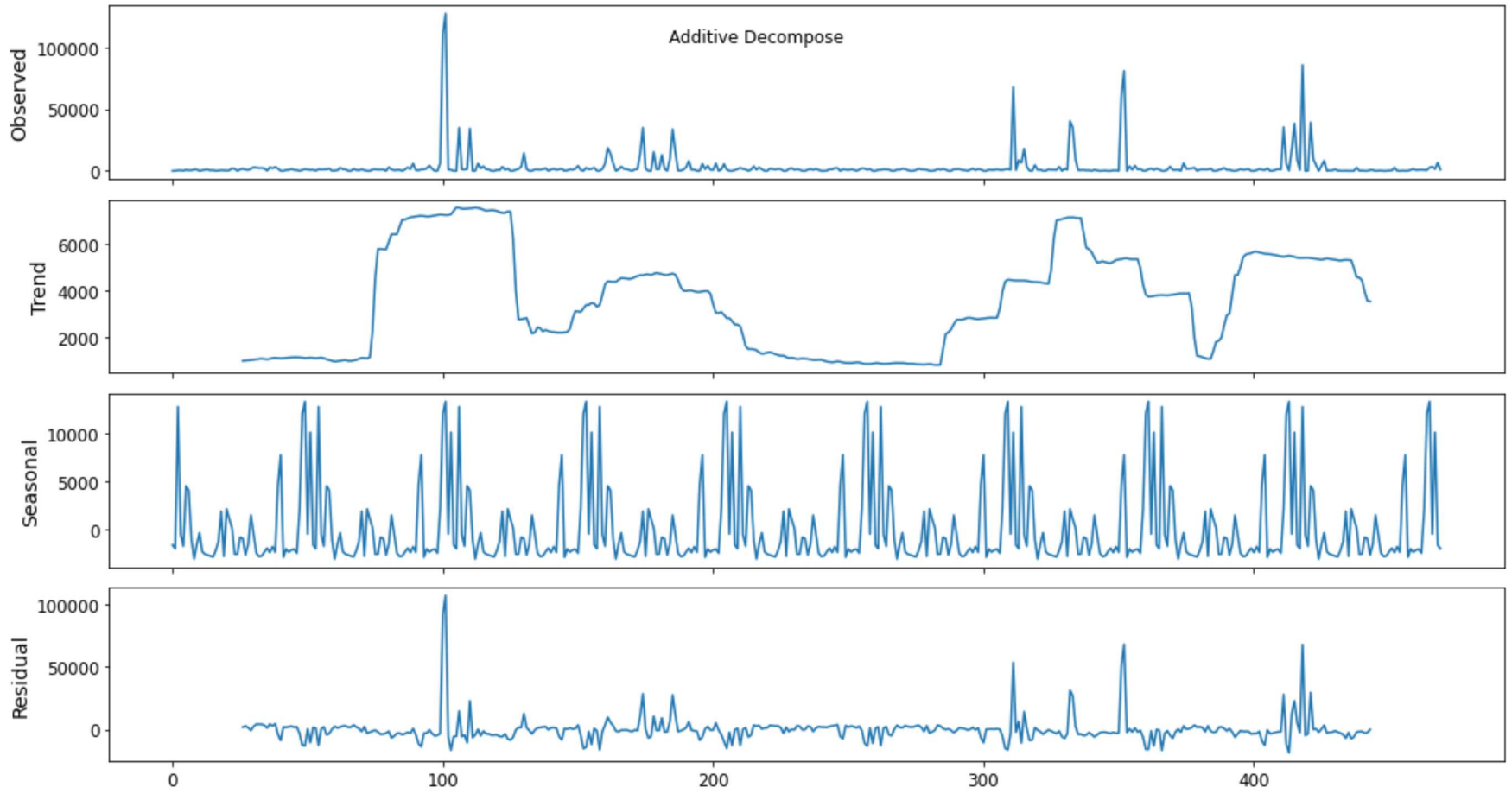
Out[89]:

```
Text(0.5, 0.98, '\nMultiplicative Decompose')
```



In [90]:

```
add_result.plot().suptitle('\nAdditive Decompose', fontsize=12)
plt.show()
```



In [91]:

```
#Additive
new_df_add = pd.concat([add_result.seasonal, add_result.trend, add_result.resid, add_result.observed], axis=1)
new_df_add.columns = ['seasoanilty', 'trend', 'residual', 'actual_values']
new_df_add.head(5)
```

Out[91]:

	seasoanilty	trend	residual	actual_values
0	-1669.239861	NaN	NaN	2
1	-2052.713418	NaN	NaN	79
2	12828.157976	NaN	NaN	422
3	-567.880486	NaN	NaN	468

	seasoanilty	trend	residual	actual_values
4	-1794.624476	NaN	NaN	111

In [92]:
```
#Multiplicative
new_df_mult = pd.concat([mul_result.seasonal, mul_result.trend, mul_result.resid, mul_result.observed], axis=1)
new_df_mult.columns = ['seasoanilty', 'trend', 'residual', 'actual_values']
new_df_mult.head(5)
```

Out[92]:

	seasoanilty	trend	residual	actual_values
0	0.556888	NaN	NaN	2
1	0.527839	NaN	NaN	79
2	3.329785	NaN	NaN	422
3	1.034855	NaN	NaN	468
4	0.748085	NaN	NaN	111

In [93]:
```
len(df)
```

Out[93]: 470

In [94]:
```
#The next step is to split the data into training and test sets

#selecting a time specific data, we can consider it as our train data
daily1 = df.loc[df['Checkout_Date'].between('2018-12-04','2020-04-30', inclusive=True)].set_index('Checkout_Date')

#we can consider it as test data after training and getting prediction of the previous data.
daily2 = df.loc[df['Checkout_Date'].between('2020-05-01','2020-05-15', inclusive=True)].set_index('Checkout_Date')
```

In [95]:
```
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
scaler.fit(daily1)
scaled_train= scaler.transform(daily1)
scaled_test= scaler.transform(daily2)
scaled_train[:5]
```

Out[95]:
```
array([[7.84972487e-06],
       [6.12278540e-04],
       [3.30473417e-03],
       [3.66582151e-03],
       [8.63469735e-04]])
```

In [97]:
```
from keras.preprocessing.sequence import TimeseriesGenerator

#define generator
n_input = 3
n_features= 1
generator = TimeseriesGenerator(scaled_train,scaled_train, length=n_input, batch_size=1)
```

In [98]:
```
X,y = generator[0]
print(f'Given the Array: \n{X.flatten()}')
print(f'Predict this y: \n{y}')
```

```
Given the Array:
[7.84972487e-06 6.12278540e-04 3.30473417e-03]
Predict this y:
[[0.00366582]]
```

In [99]:
```
X.shape
```

Out[99]:
```
(1, 3, 1)
```

In [100…
```
# now taking input or step size as 20

n_input = 20
generator = TimeseriesGenerator(scaled_train,scaled_train, length=n_input, batch_size=1)
```

In [101…
```
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import LSTM, Flatten

#defining model
model = Sequential()
model.add(LSTM(200, activation='relu', input_shape=(n_input, n_features)))
model.add(Dense(1))
```

```
model.compile(optimizer='adam', loss='mse')

model.summary()
```

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 lstm_1 (LSTM)                (None, 200)               161600

 dense_1 (Dense)              (None, 1)                 201

=================================================================
Total params: 161,801
Trainable params: 161,801
Non-trainable params: 0
_________________________________________________________________
```

In [102…

```
model.fit_generator(generator,epochs=200)
```

```
Epoch 1/200
439/439 [==============================] - 8s 16ms/step - loss: 0.0086
Epoch 2/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0084
Epoch 3/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0082
Epoch 4/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0083
Epoch 5/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0078
Epoch 6/200
439/439 [==============================] - 7s 15ms/step - loss: 0.0076
Epoch 7/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0075
Epoch 8/200
439/439 [==============================] - 7s 15ms/step - loss: 0.0073
Epoch 9/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0073
Epoch 10/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0071
Epoch 11/200
439/439 [==============================] - 7s 15ms/step - loss: 0.0070
Epoch 12/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0071
Epoch 13/200
439/439 [==============================] - 7s 15ms/step - loss: 0.0069
Epoch 14/200
```

```
439/439 [==============================] - 7s 15ms/step - loss: 0.0068
Epoch 15/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0069
Epoch 16/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0066
Epoch 17/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0072
Epoch 18/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0067
Epoch 19/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0067
Epoch 20/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0067
Epoch 21/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0067
Epoch 22/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0066
Epoch 23/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0067
Epoch 24/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0066
Epoch 25/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0067
Epoch 26/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0065
Epoch 27/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0067
Epoch 28/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0066
Epoch 29/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0064
Epoch 30/200
439/439 [==============================] - 7s 16ms/step - loss: 0.0063
Epoch 31/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0066
Epoch 32/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0063
Epoch 33/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0064
Epoch 34/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0063
Epoch 35/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0064
Epoch 36/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0067
Epoch 37/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0064
Epoch 38/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0062
```

```
Epoch 39/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0065
Epoch 40/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0061
Epoch 41/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0061
Epoch 42/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0062
Epoch 43/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0060
Epoch 44/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0062
Epoch 45/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0059
Epoch 46/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0065
Epoch 47/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0062
Epoch 48/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0061
Epoch 49/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0061
Epoch 50/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0061
Epoch 51/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0059
Epoch 52/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0060
Epoch 53/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0064
Epoch 54/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0061
Epoch 55/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0061
Epoch 56/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0061
Epoch 57/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0060
Epoch 58/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0062
Epoch 59/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0058
Epoch 60/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0057
Epoch 61/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0059
Epoch 62/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0060
Epoch 63/200
```

```
439/439 [==============================] - 8s 18ms/step - loss: 0.0060
Epoch 64/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0060
Epoch 65/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0057
Epoch 66/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0056
Epoch 67/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0060
Epoch 68/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0059
Epoch 69/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0062
Epoch 70/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0060
Epoch 71/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0059
Epoch 72/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0058
Epoch 73/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0056
Epoch 74/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0058
Epoch 75/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0057
Epoch 76/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0057
Epoch 77/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0070
Epoch 78/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0060
Epoch 79/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0061
Epoch 80/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0058
Epoch 81/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0057
Epoch 82/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0056
Epoch 83/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0056
Epoch 84/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0056
Epoch 85/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0056
Epoch 86/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0056
Epoch 87/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0055
```

```
Epoch 88/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0055
Epoch 89/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0052
Epoch 90/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0056
Epoch 91/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0054
Epoch 92/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0055
Epoch 93/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0051
Epoch 94/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0059
Epoch 95/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0056
Epoch 96/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0053
Epoch 97/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0061
Epoch 98/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0052
Epoch 99/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0054
Epoch 100/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0050
Epoch 101/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0049
Epoch 102/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0049
Epoch 103/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0054
Epoch 104/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0052
Epoch 105/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0051
Epoch 106/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0050
Epoch 107/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0076
Epoch 108/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0058
Epoch 109/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0053
Epoch 110/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0053
Epoch 111/200
439/439 [==============================] - 7s 17ms/step - loss: 0.0053
Epoch 112/200
```

```
439/439 [==============================] - 8s 17ms/step - loss: 0.0054
Epoch 113/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0050
Epoch 114/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0048
Epoch 115/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0049
Epoch 116/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0047
Epoch 117/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0048
Epoch 118/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0046
Epoch 119/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0051
Epoch 120/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0047
Epoch 121/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0056
Epoch 122/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0047
Epoch 123/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0045
Epoch 124/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0046
Epoch 125/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0051
Epoch 126/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0045
Epoch 127/200
439/439 [==============================] - 8s 19ms/step - loss: 0.0047
Epoch 128/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0044
Epoch 129/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0065
Epoch 130/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0047
Epoch 131/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0048
Epoch 132/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0050
Epoch 133/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0044
Epoch 134/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0046
Epoch 135/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0047
Epoch 136/200
439/439 [==============================] - 8s 19ms/step - loss: 0.0047
```

```
Epoch 137/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0046
Epoch 138/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0044
Epoch 139/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0046
Epoch 140/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0045
Epoch 141/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0044
Epoch 142/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0042
Epoch 143/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0043
Epoch 144/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0043
Epoch 145/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0044
Epoch 146/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0047
Epoch 147/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0046
Epoch 148/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0046
Epoch 149/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0041
Epoch 150/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0044
Epoch 151/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0042
Epoch 152/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0043
Epoch 153/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0042
Epoch 154/200
439/439 [==============================] - 8s 19ms/step - loss: 0.0047
Epoch 155/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0045
Epoch 156/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0041
Epoch 157/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0042
Epoch 158/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0045
Epoch 159/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0050
Epoch 160/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0043
Epoch 161/200
```

```
439/439 [==============================] - 8s 18ms/step - loss: 0.0041
Epoch 162/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0041
Epoch 163/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0042
Epoch 164/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0043
Epoch 165/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0044
Epoch 166/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0041
Epoch 167/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0046
Epoch 168/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0044
Epoch 169/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0041
Epoch 170/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0040
Epoch 171/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0040
Epoch 172/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0042
Epoch 173/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0044
Epoch 174/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0042
Epoch 175/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0042
Epoch 176/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0041
Epoch 177/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0040
Epoch 178/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0040
Epoch 179/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0041
Epoch 180/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0040
Epoch 181/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0039
Epoch 182/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0040
Epoch 183/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0039
Epoch 184/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0043
Epoch 185/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0047
```

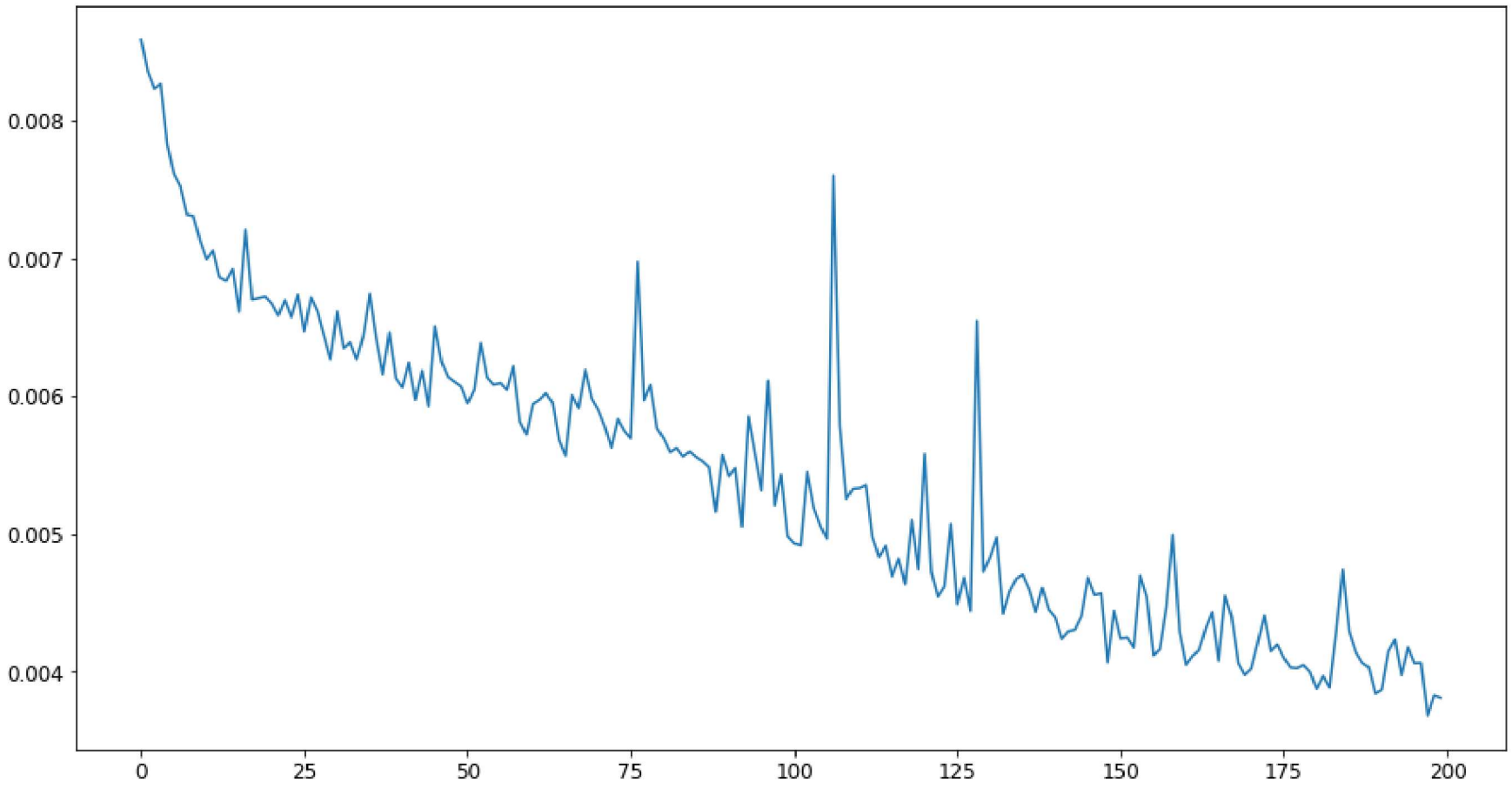
```
Epoch 186/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0043
Epoch 187/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0041
Epoch 188/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0041
Epoch 189/200
439/439 [==============================] - 8s 17ms/step - loss: 0.0040
Epoch 190/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0038
Epoch 191/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0039
Epoch 192/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0041
Epoch 193/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0042
Epoch 194/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0040
Epoch 195/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0042
Epoch 196/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0041
Epoch 197/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0041
Epoch 198/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0037
Epoch 199/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0038
Epoch 200/200
439/439 [==============================] - 8s 18ms/step - loss: 0.0038
```

Out[102… `<keras.callbacks.History at 0x7fd205239a90>`

In [104…

```
loss_per_epoch = model.history.history['loss']
plt.plot(range(len(loss_per_epoch)),loss_per_epoch)
```

Out[104… `[<matplotlib.lines.Line2D at 0x7fd20526f190>]`



In [105…
```
last_train_batch = scaled_train[-20:]
```

In [106…
```
last_train_batch = last_train_batch.reshape((1, n_input, n_features))
```

In [107…
```
model.predict(last_train_batch)
```

Out[107…
```
array([[0.01645548]], dtype=float32)
```

In [110…
```
scaled_test[0]     #We got value almost close to the orginal value
```

```
array([0.01179029])
```

Out[110…

In [111…
```
test_predictions = []

first_eval_batch = scaled_train[-n_input:]
current_batch = first_eval_batch.reshape((1, n_input, n_features))

for i in range(len(daily2)):

    # get the prediction value for the first batch
    current_pred = model.predict(current_batch)[0]

    # append the prediction into the array
    test_predictions.append(current_pred)

    # use the prediction to update the batch and remove the first value
    current_batch = np.append(current_batch[:,1:,:],[[current_pred]],axis=1)
```

In [112…
```
test_predictions
```

Out[112…
```
[array([0.01645548], dtype=float32),
 array([0.01905567], dtype=float32),
 array([0.0193505], dtype=float32),
 array([0.01914279], dtype=float32),
 array([0.01835018], dtype=float32),
 array([0.01875078], dtype=float32),
 array([0.01809148], dtype=float32),
 array([0.01403989], dtype=float32),
 array([0.01035471], dtype=float32),
 array([0.01053466], dtype=float32),
 array([0.01131531], dtype=float32)]
```

In [113…
```
true_predictions = scaler.inverse_transform(test_predictions)
```

In [114…
```
daily2['Predictions'] = true_predictions
```

In [122…
```
daily2.tail()
```

Out[122…

	Order_Quantity	Predictions

Checkout_Date	Order_Quantity	Predictions
Checkout_Date		
2020-05-11	2749	2305.728299
2020-05-12	3366	1789.584051
2020-05-13	1491	1320.117903
2020-05-14	6631	1343.041580
2020-05-15	1018	1442.491334

In [118…

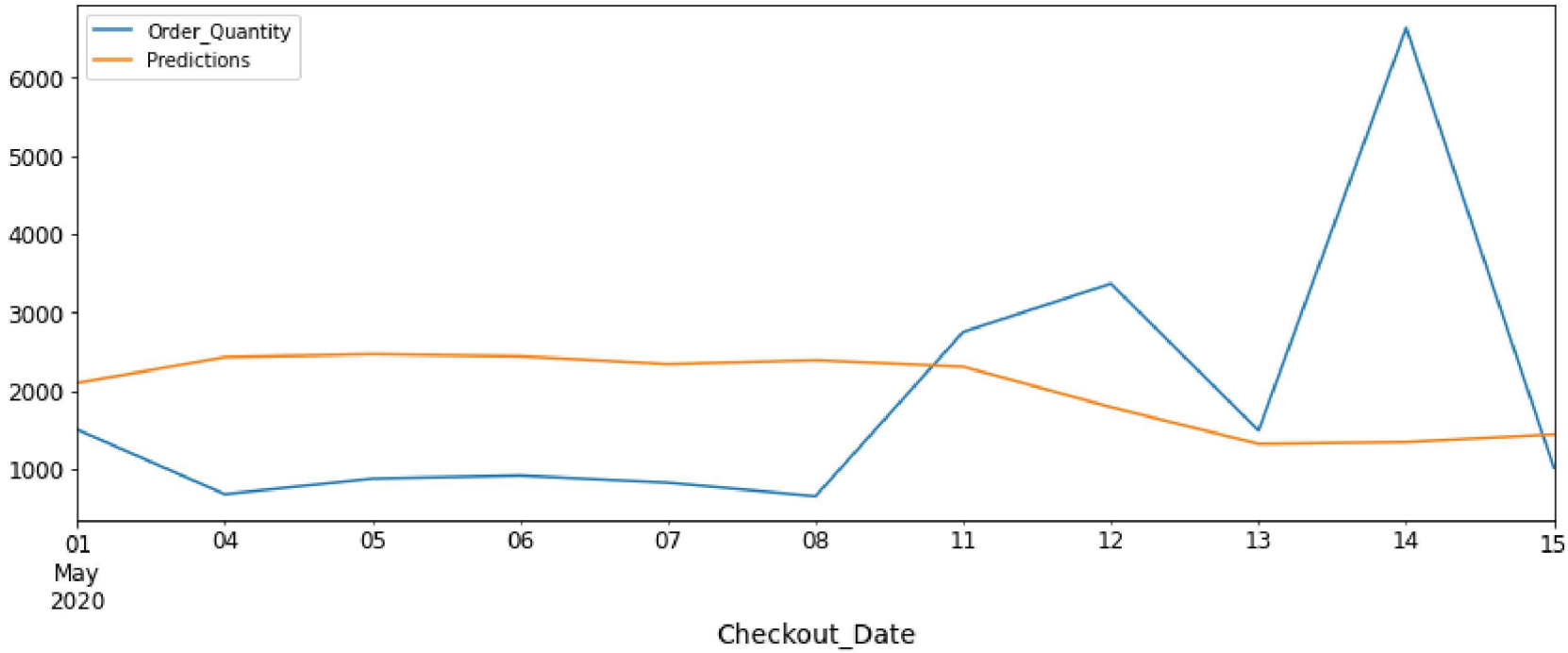
```
print(daily2.Order_Quantity.mean())
print(daily2.Predictions.mean())
```

```
1882.5454545454545
2032.8194330564954
```

In [119…

```
daily2.plot(figsize = (14,5))
```

Out[119… `<matplotlib.axes._subplots.AxesSubplot at 0x7fd20e68a210>`



In [120…

```
# Checking Evaluation/Performance Metrics

lstm_rmse_error = rmse(daily2['Order_Quantity'], daily2["Predictions"])
lstm_mse_error = lstm_rmse_error**2

print(f'MSE Error: {lstm_mse_error}\nRMSE Error: {lstm_rmse_error}')
```

```
MSE Error: 4038424.572664323
RMSE Error: 2009.5831838130819
```