Barclays Case Study Competion

1. Importing the libraries

- Numpy - for all the mathematical operations
- Pandas - for loading the dataset and data preprocessing
- Matplotlib - for performing Data Visualization
- Datetime - for operations related to BusinessDate
- Sklearn - for the Linear Regression algorithm

In [1]:
```
# Importing the Libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import datetime
from sklearn.linear_model import LinearRegression
```

2. Loading the Dataset

In [2]:
```
# Loading the Dataset
df = pd.read_excel('dataset.xlsx', index=None)
df.head()
```

Out[2]:

	BusinessDate	SEDOL	Counterparty_Account_ID	Position_Quantity_SD
0	2019-05-02	5BDN21B	11009	121000.0
1	2019-05-02	5BDN21B	14120	928200.0
2	2019-05-02	5BDN21B	16109	1452000.0
3	2019-05-02	5BDN21B	16140	-40600.0
4	2019-05-02	5BDN21B	62004	10000.0

3. Grouping the Dataset

Grouping the dataset according to the SEDOLs and BusinessDate and taking sum of all the data on a particular date

In [3]:
```
df_grp = df.groupby(['SEDOL', 'BusinessDate']).sum()
df_grp
```

Out[3]:

		Position_Quantity_SD
SEDOL	**BusinessDate**	
	2019-05-02	2254208.0
	2019-05-03	2249508.0
5BDN21B	2019-05-06	2303108.0
	2019-05-07	2206708.0
	2019-05-08	2188708.0
...	...	...
	2019-09-04	3913907.0
	2019-09-05	3677907.0
74ZI41B	2019-09-06	3713907.0
	2019-09-09	1874907.0
	2019-09-10	2195907.0

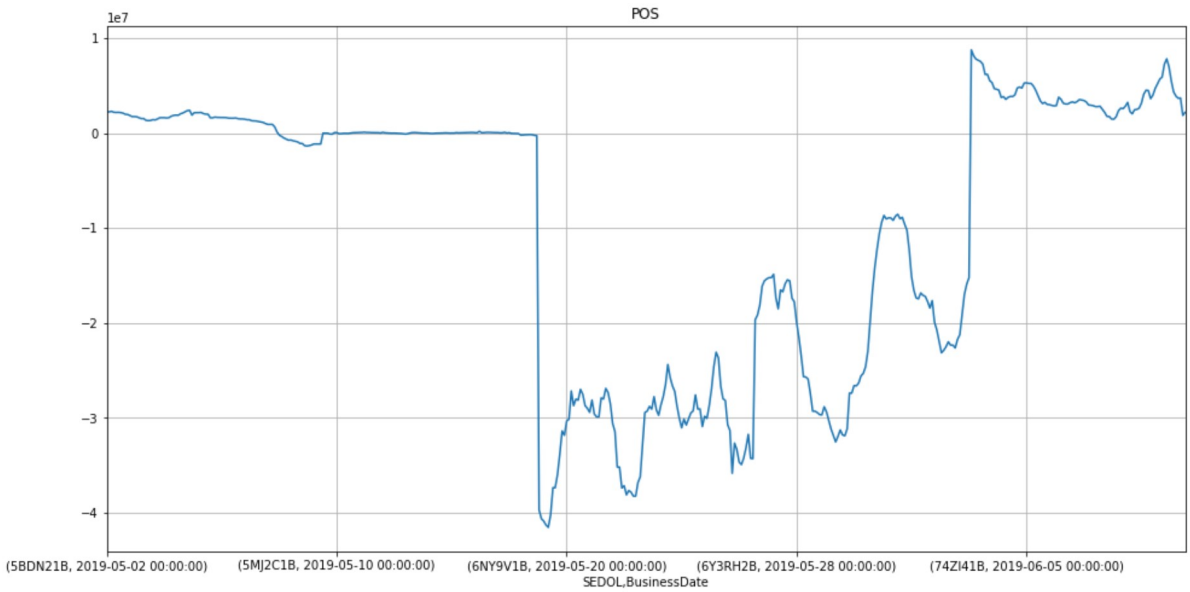
470 rows × 1 columns

In [4]:
```
# Different Sedols present
names=df.SEDOL.unique().tolist()
names
```

Out[4]:
```
['5BDN21B', '5MJ2C1B', '6NY9V1B', '6Y3RH2B', '74ZI41B']
```

In [5]:
```
# Visualization of the complete Data
df_grp['Position_Quantity_SD'].plot.line(label='BC', figsize=(16,8), title='POS', grid=True)
```

Out[5]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x208af553160>
```



4. Dividing the dataset

Dividing the dataset into 5 different Sedols and performimg the following steps on the resulting dataset.

- Creating a DataFrame of the SEDOL
- Grouping the resulting Dataframe according the BusinessDate and adding the Position_Quantity_SD of the same BusinessDates
- Plotting the graph
- Creating a window of 30 and dividing the Training and Testing Data
- Training the model using Linear Regression and plotting the Training and Testing curves (for the first SEDOL - tried different Regression algorithms but didn't see any significant improvements)
- Applying Simple Moving Average (SMA) using the rolling() function of the Pandas DataFrame with window size = 30 and plotting the graph
- Applying Exponential Moving Average (EMA) using the ewm() function of the Pandas DataFrame with window size = 30 and plotting the combined graph of SMA and EMA

SEDOL1

In [6]:
```
df_sedol1 = df.loc[df.SEDOL=='5BDN21B']
df_sedol1.head()
```

Out[6]:

	BusinessDate	SEDOL	Counterparty_Account_ID	Position_Quantity_SD
0	2019-05-02	5BDN21B	11009	121000.0
1	2019-05-02	5BDN21B	14120	928200.0
2	2019-05-02	5BDN21B	16109	1452000.0
3	2019-05-02	5BDN21B	16140	-40600.0
4	2019-05-02	5BDN21B	62004	10000.0

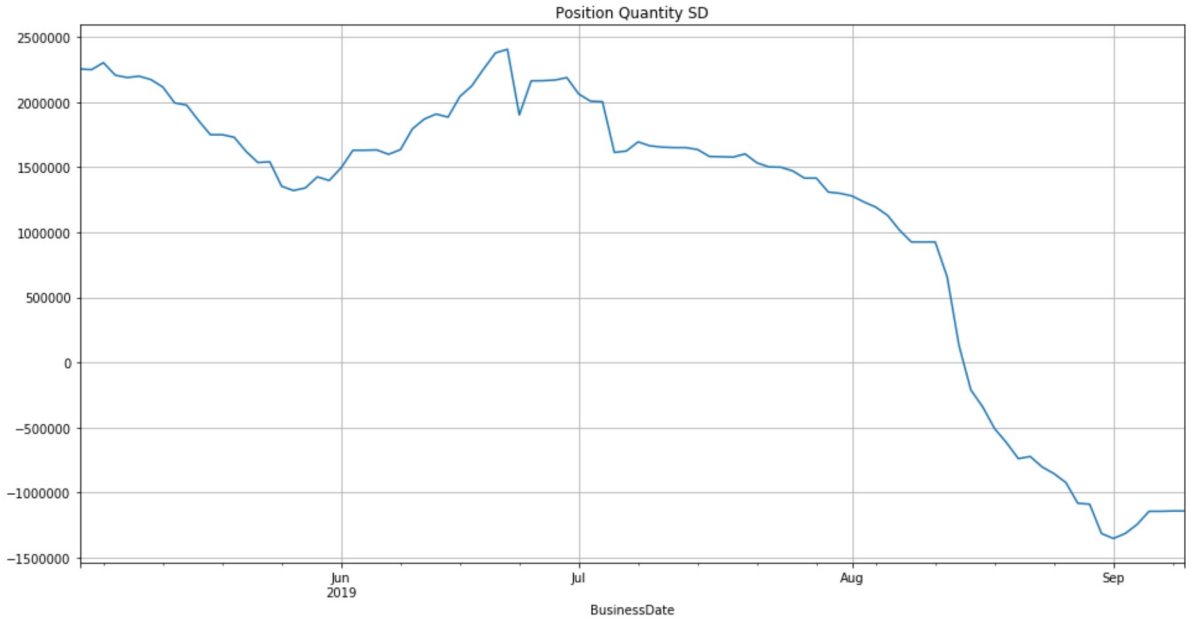
In [7]:
```
df_sedol1_edit = df_sedol1.groupby(['BusinessDate']).sum()
df_sedol1_edit.head()
```

Out[7]:

	Position_Quantity_SD
BusinessDate	
2019-05-02	2254208.0
2019-05-03	2249508.0
2019-05-06	2303108.0
2019-05-07	2206708.0
2019-05-08	2188708.0

In [8]:
```
df_sedol1_edit['Position_Quantity_SD'].plot.line(label='SEDOL1', figsize=(16,8), title='Position Quantity SD', grid=True)
```

Out[8]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x208a43ff518>
```



In [9]:
```
#Train the model on the last 30 days and predict the label for the 31th day
window = 30

num_samples = len(df_sedol1_edit) - window
indices = np.arange(num_samples).astype(np.int)[:,None] + np.arange(window + 1).astype(np.int)
len(indices)
```

Out[9]:
```
64
```

In [10]:
```
data = df_sedol1_edit['Position_Quantity_SD'].values[indices]
data
```

Out[10]:
```
array([[ 2254208.,  2249508.,  2303108., ...,  1793908.,  1869608.,
         1908308.],
       [ 2249508.,  2303108.,  2206708., ...,  1869608.,  1908308.,
         1884608.],
       [ 2303108.,  2206708.,  2188708., ...,  1908308.,  1884608.,
         2043008.],
       ...,
       [ 1416208.,  1416208.,  1309208., ..., -1244092., -1143592.,
        -1143592.],
       [ 1416208.,  1309208.,  1299108., ..., -1143592., -1143592.,
        -1141192.],
       [ 1309208.,  1299108.,  1279008., ..., -1143592., -1141192.,
        -1141192.]])
```

In [11]:
```
X = data[:, :-1]
y = data[:, -1]
```

In [12]:
```
split_frac = 0.8
split_indices = int(split_frac * num_samples)
X_train = X[:split_indices]
y_train = y[:split_indices]
X_test = X[split_indices:]
y_test = y[split_indices:]
split_indices
```

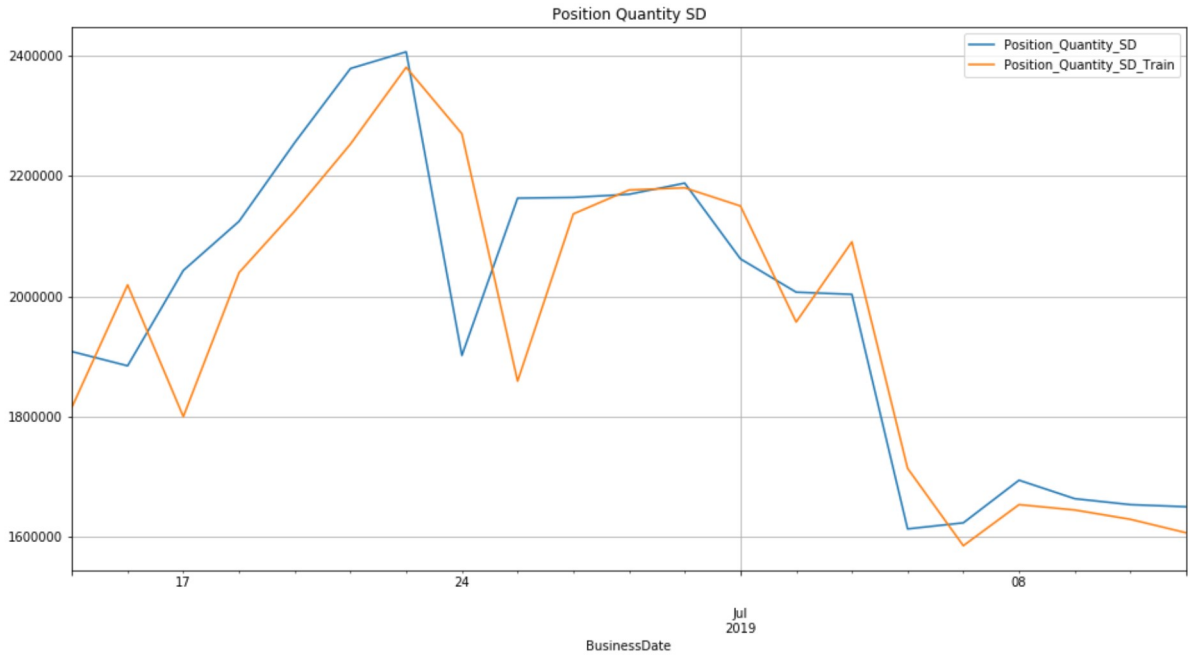
Out[12]: 51

In [13]:
```
#Train
linear_reg_model = LinearRegression()
linear_reg_model.fit(X_train, y_train)

#Inferences
y_pred_train_linear_reg = linear_reg_model.predict(X_train)
y_pred_linear_reg = linear_reg_model.predict(X_test)
```

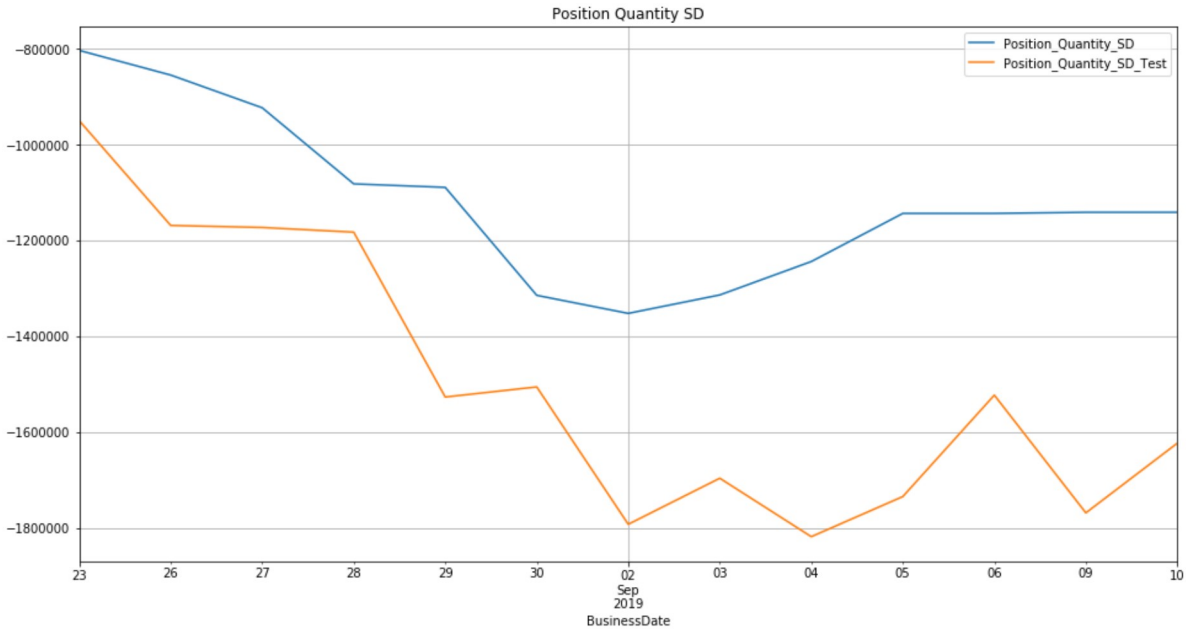
In [14]:
```
#Plot the graph for it has trained on the training data
df_linear = df_sedol1_edit.copy()
df_linear = df_linear.iloc[window:split_indices]
df_linear['Position_Quantity_SD_Train'] = y_pred_train_linear_reg[:-window]
df_linear.plot(label='SEDOL1', figsize=(16, 8), title='Position Quantity SD', grid=
True)
```

Out[14]: <matplotlib.axes._subplots.AxesSubplot at 0x208b1b24e48>



In [15]:
```
#Plot the graph for the testing data
df_linear = df_sedol1_edit.copy()
df_linear = df_linear.iloc[split_indices+window:]
df_linear['Position_Quantity_SD_Test'] = y_pred_linear_reg
df_linear.plot(label='SEDOL1', figsize=(16, 8), title='Position Quantity SD', grid=
True)
```

Out[15]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x208b1e37978>
```



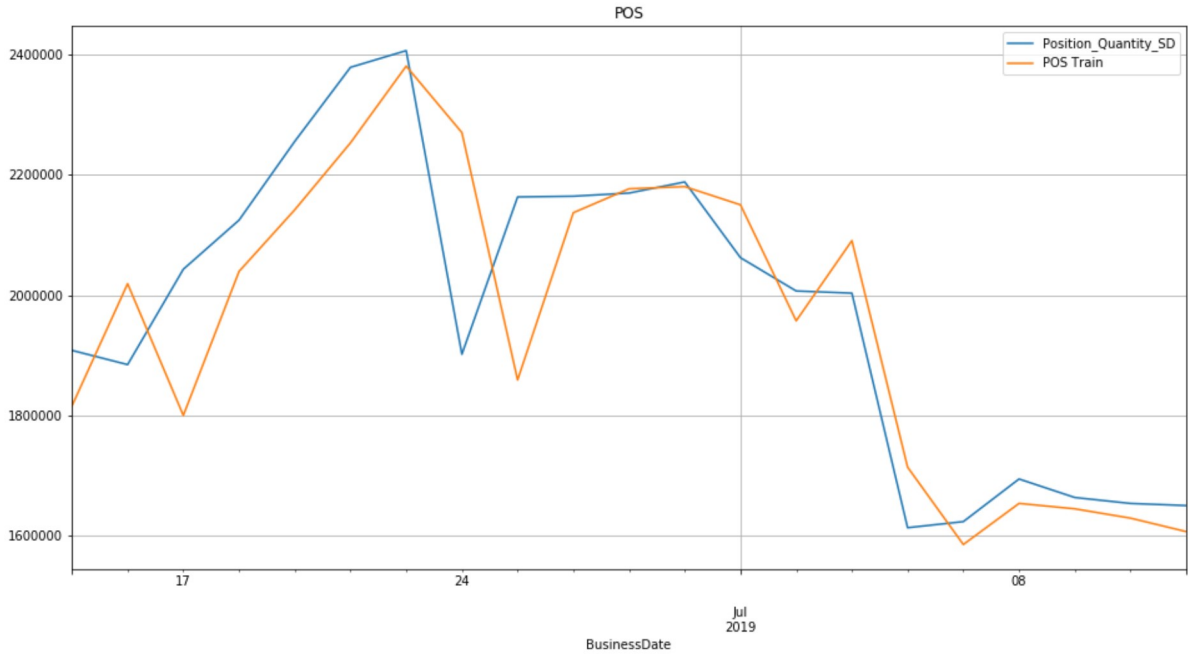
In [16]:
```
from sklearn.linear_model import Ridge

#Train
ridge_model = Ridge()
ridge_model.fit(X_train, y_train)

#Inferences
y_pred_train_ridge = ridge_model.predict(X_train)
y_pred_ridge = ridge_model.predict(X_test)
```

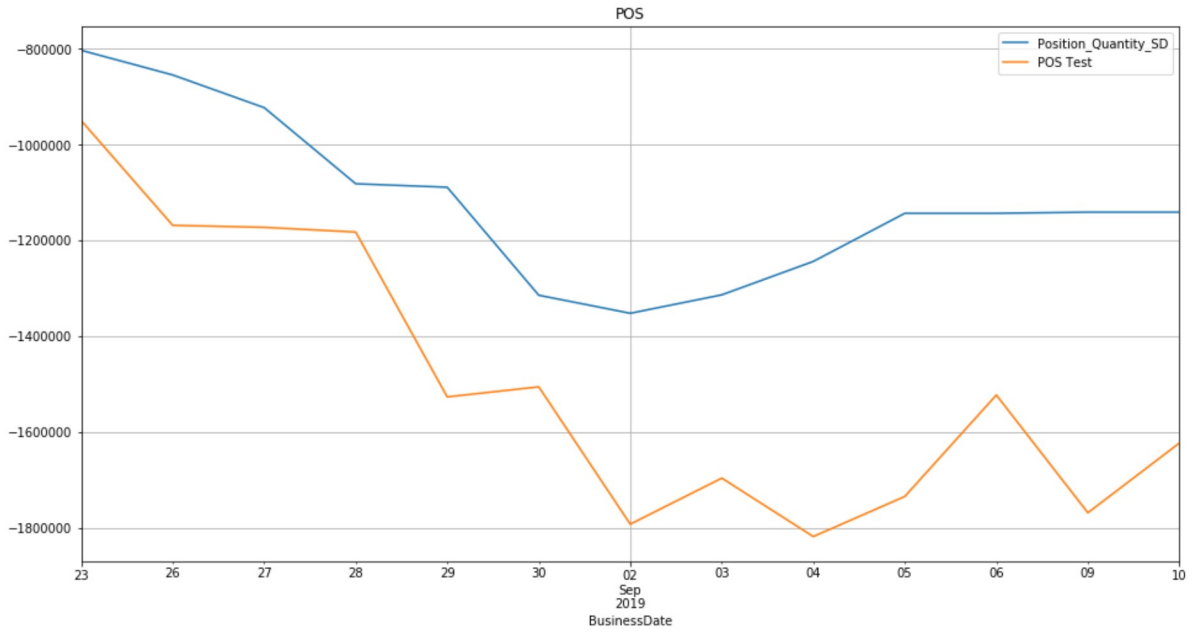
In [17]:
```
#Plot the graph for it has trained on the training data
df_ridge = df_sedol1_edit.copy()
df_ridge = df_ridge.iloc[window:split_indices]
df_ridge['POS Train'] = y_pred_train_ridge[:-window]
df_ridge.plot(label='SEDOL1', figsize=(16, 8), title='POS', grid=True)
```

Out[17]: <matplotlib.axes._subplots.AxesSubplot at 0x208b1bef1d0>



In [18]:
```
#Plot the graph for the testing data
df_ridge = df_sedol1_edit.copy()
df_ridge = df_ridge.iloc[split_indices+window:]
df_ridge['POS Test'] = y_pred_ridge
df_ridge.plot(label='SEDOL1', figsize=(16, 8), title='POS', grid=True)
```

Out[18]: <matplotlib.axes._subplots.AxesSubplot at 0x208b20d8f28>



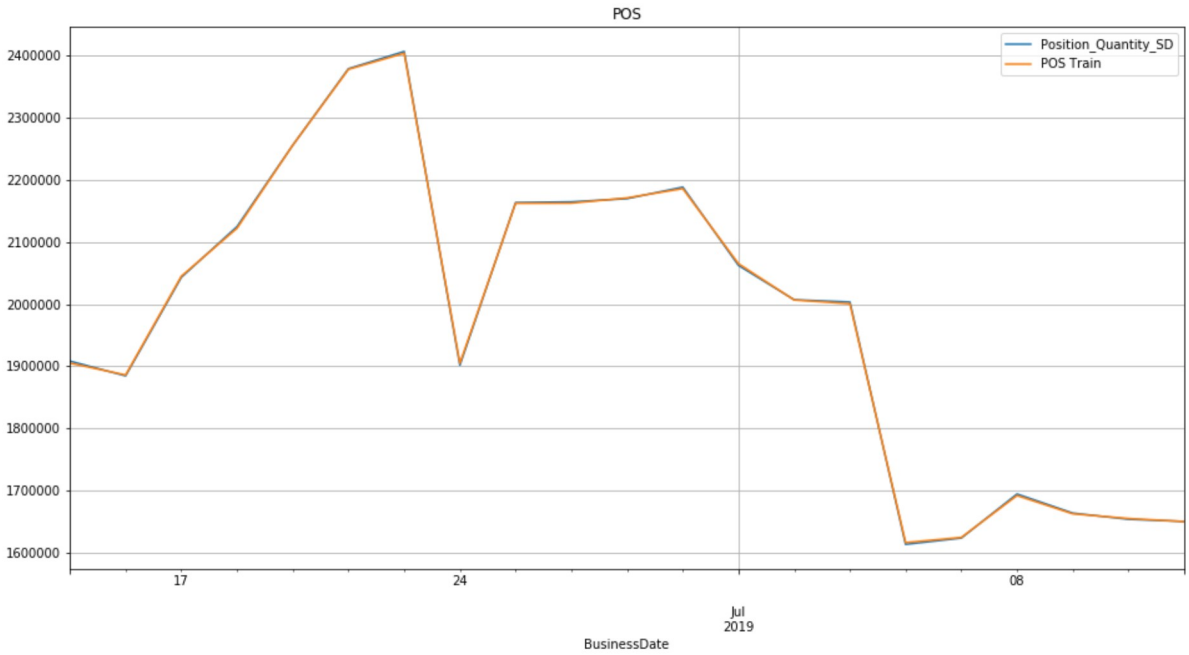
In [19]:
```
from sklearn.ensemble import GradientBoostingRegressor

#Train
gb_model = GradientBoostingRegressor()
gb_model.fit(X_train, y_train)

#Inferences
y_pred_train_gb = gb_model.predict(X_train)
y_pred_gb = gb_model.predict(X_test)
```

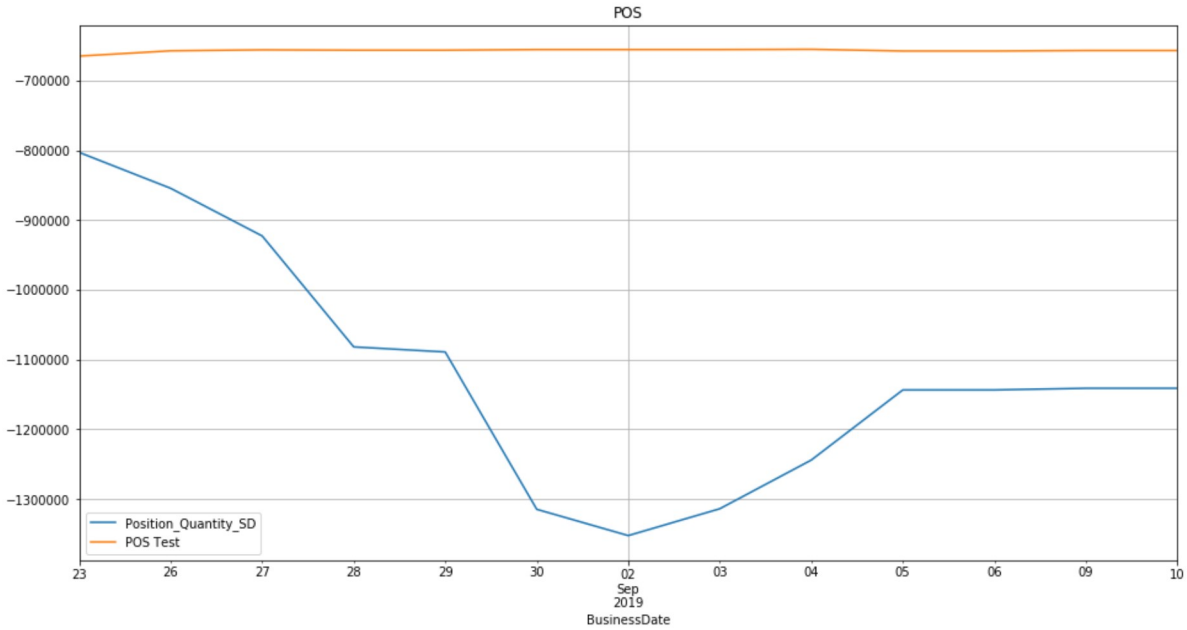
In [20]:
```
#Plot the graph for it has trained on the training data
df_gb = df_sedol1_edit.copy()
df_gb = df_gb.iloc[window:split_indices]
df_gb['POS Train'] = y_pred_train_gb[:-window]
df_gb.plot(label='SEDOL1', figsize=(16, 8), title='POS', grid=True)
```

Out[20]: <matplotlib.axes._subplots.AxesSubplot at 0x208b2112cc0>



In [21]:
```
#Plot the graph for the testing data
df_gb = df_sedol1_edit.copy()
df_gb = df_gb.iloc[split_indices+window:]
df_gb['POS Test'] = y_pred_gb
df_gb.plot(label='SEDOL1', figsize=(16, 8), title='POS', grid=True)
```

Out[21]: <matplotlib.axes._subplots.AxesSubplot at 0x208b2760f28>



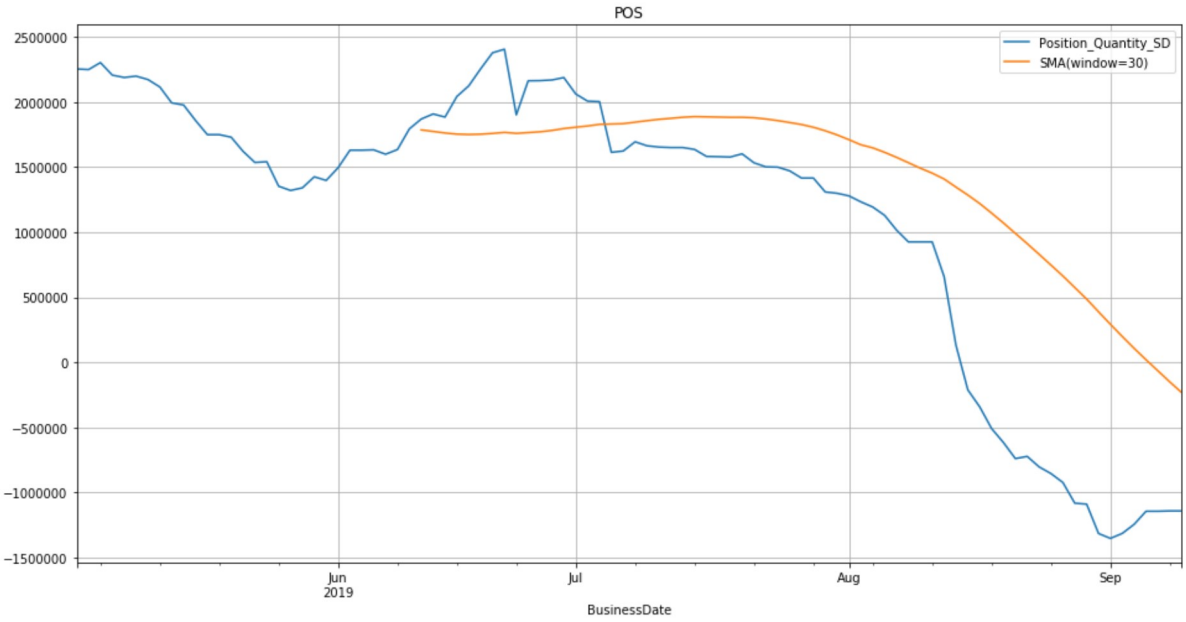
In [22]:
```
df_sedol1_edit['SMA(window=30)'] = df_sedol1_edit.Position_Quantity_SD.rolling(wind
ow=30).mean()
df_sedol1_edit.tail()
```

Out[22]:

BusinessDate	Position_Quantity_SD	SMA(window=30)
2019-09-04	-1244092.0	107194.666667
2019-09-05	-1143592.0	20021.333333
2019-09-06	-1143592.0	-65305.333333
2019-09-09	-1141192.0	-150552.000000
2019-09-10	-1141192.0	-232232.000000

In [23]:
```
df_sedol1_edit.plot(label='SEDOL1', figsize=(16, 8), title='POS', grid=True)
```

Out[23]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x208af17d908>
```



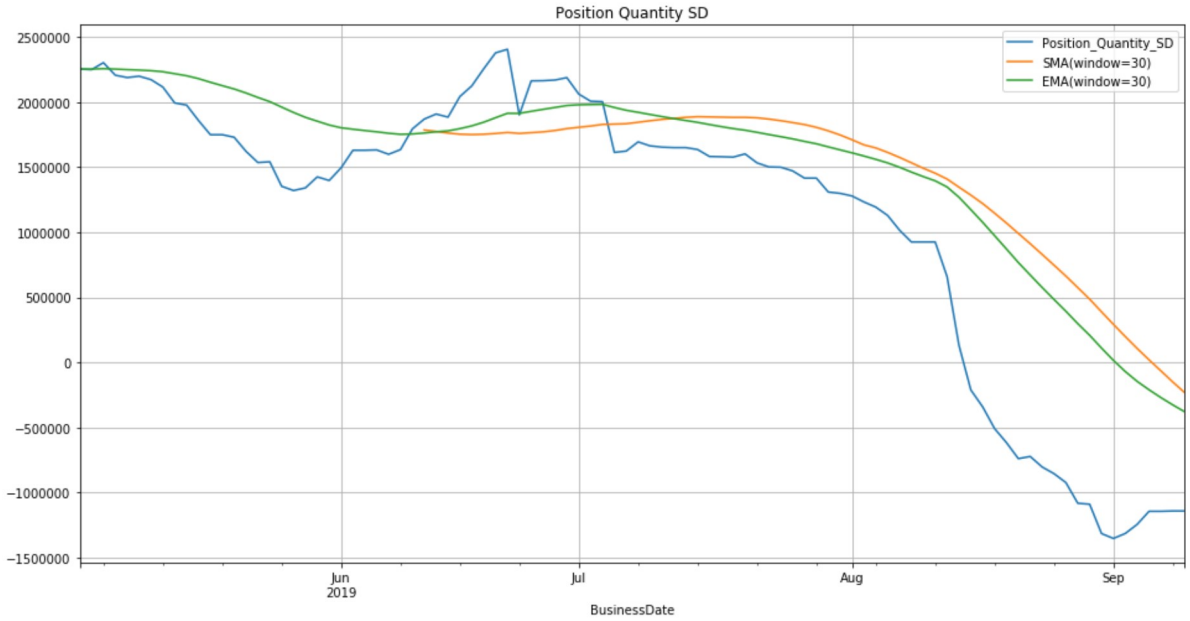
In [24]:
```
df_sedol1_edit['EMA(window=30)'] = df_sedol1_edit.Position_Quantity_SD.ewm(span=30,
adjust=False).mean()
df_sedol1_edit.head()
```

Out[24]:

BusinessDate	Position_Quantity_SD	SMA(window=30)	EMA(window=30)
2019-05-02	2254208.0	NaN	2.254208e+06
2019-05-03	2249508.0	NaN	2.253905e+06
2019-05-06	2303108.0	NaN	2.257079e+06
2019-05-07	2206708.0	NaN	2.253829e+06
2019-05-08	2188708.0	NaN	2.249628e+06

In [25]:
```
df_sedol1_edit.plot(label='SEDOL1', figsize=(16, 8), title='Position Quantity SD',
grid=True)
```

Out[25]: <matplotlib.axes._subplots.AxesSubplot at 0x208b25b2c88>



SEDOL2

In [26]:
```
df_sedol2 = df.loc[df.SEDOL=='5MJ2C1B']
df_sedol2.head()
```

Out[26]:

	BusinessDate	SEDOL	Counterparty_Account_ID	Position_Quantity_SD
8	2019-05-02	5MJ2C1B	0010V	200.0
9	2019-05-02	5MJ2C1B	1003V	100.0
10	2019-05-02	5MJ2C1B	12280	10000.0
11	2019-05-02	5MJ2C1B	25SJP	-200.0
12	2019-05-02	5MJ2C1B	3210P	-151400.0

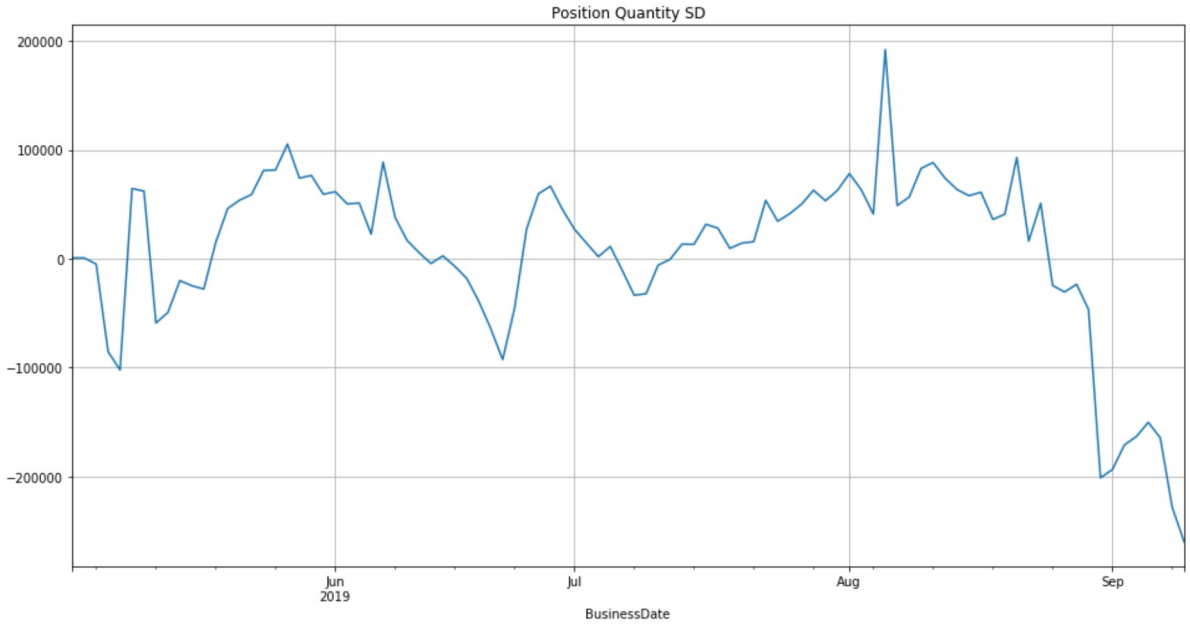
In [27]:
```
df_sedol2_edit = df_sedol2.groupby(['BusinessDate']).sum()
df_sedol2_edit.head()
```

Out[27]:

BusinessDate	Position_Quantity_SD
2019-05-02	732.0
2019-05-03	732.0
2019-05-06	-4902.0
2019-05-07	-85468.0
2019-05-08	-102202.0

In [28]:
```
df_sedol2_edit['Position_Quantity_SD'].plot.line(label='SEDOL2', figsize=(16,8), ti
tle='Position Quantity SD', grid=True)
```

Out[28]: <matplotlib.axes._subplots.AxesSubplot at 0x208af19ccf8>



In [29]:
```
#Train the model on the last 30 days and predict the label for the 31st day
window = 30

num_samples = len(df_sedol2_edit) - window
indices = np.arange(num_samples).astype(np.int)[:,None] + np.arange(window + 1).ast
ype(np.int)
len(indices)
```

Out[29]: 64

In [30]:
```
data = df_sedol2_edit['Position_Quantity_SD'].values[indices]
data
```

Out[30]:
```
array([[    732.,     732.,   -4902., ...,   16903.,    5703.,   -4402.],
       [    732.,   -4902.,  -85468., ...,    5703.,   -4402.,    2669.],
       [  -4902.,  -85468., -102202., ...,   -4402.,    2669.,   -7031.],
       ...,
       [  50113.,   62928.,   53219., ..., -163315., -150291., -164515.],
       [  62928.,   53219.,   62813., ..., -150291., -164515., -228391.],
       [  53219.,   62813.,   78214., ..., -164515., -228391., -260215.]])
```

In [31]:
```
X = data[:, :-1]
y = data[:, -1]
```

In [32]:
```
split_frac = 0.8
split_indices = int(split_frac * num_samples)
X_train = X[:split_indices]
y_train = y[:split_indices]
X_test = X[split_indices:]
y_test = y[split_indices:]
split_indices
```

Out[32]: 51

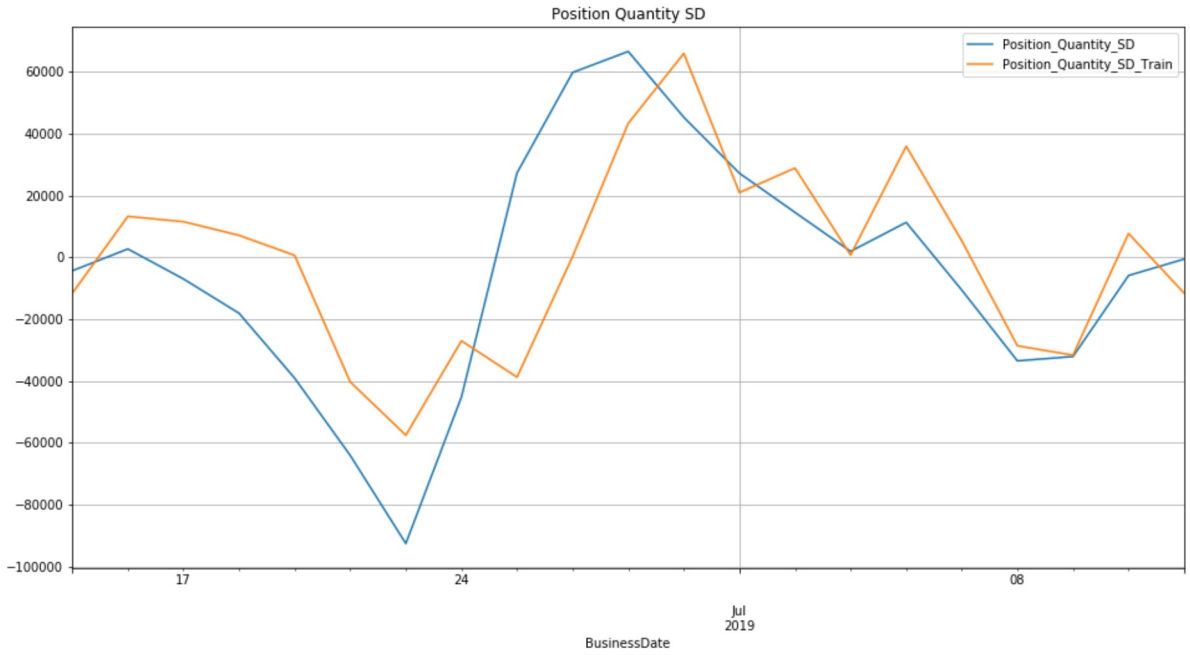
In [33]:
```
from sklearn.linear_model import LinearRegression

#Train
linear_reg_model = LinearRegression()
linear_reg_model.fit(X_train, y_train)

#Inferences
y_pred_train_linear_reg = linear_reg_model.predict(X_train)
y_pred_linear_reg = linear_reg_model.predict(X_test)
```

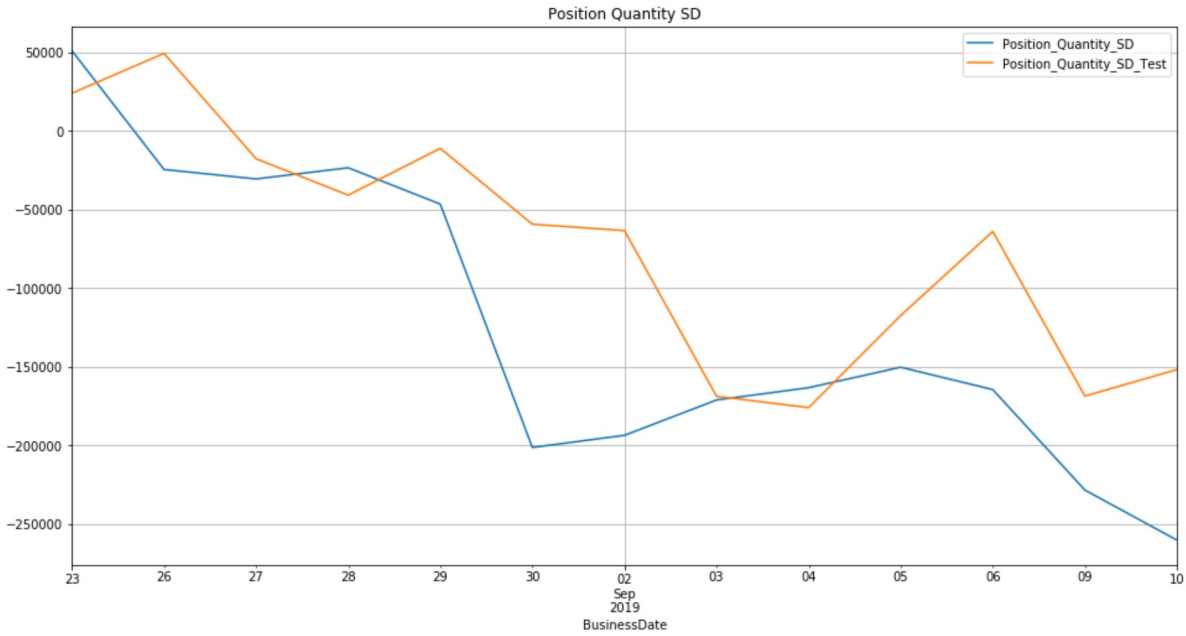
In [34]:
```
#Plot the graph for it has trained on the training data
df_linear = df_sedol2_edit.copy()
df_linear = df_linear.iloc[window:split_indices]
df_linear['Position_Quantity_SD_Train'] = y_pred_train_linear_reg[:-window]
df_linear.plot(label='SEDOL2', figsize=(16, 8), title='Position Quantity SD', grid=
True)
```

Out[34]: <matplotlib.axes._subplots.AxesSubplot at 0x208af399b70>



In [35]:
```
#Plot the graph for the testing data
df_linear = df_sedol2_edit.copy()
df_linear = df_linear.iloc[split_indices+window:]
df_linear['Position_Quantity_SD_Test'] = y_pred_linear_reg
df_linear.plot(label='SEDOL2', figsize=(16, 8), title='Position Quantity SD', grid=
True)
```

Out[35]: <matplotlib.axes._subplots.AxesSubplot at 0x208af1a4f98>



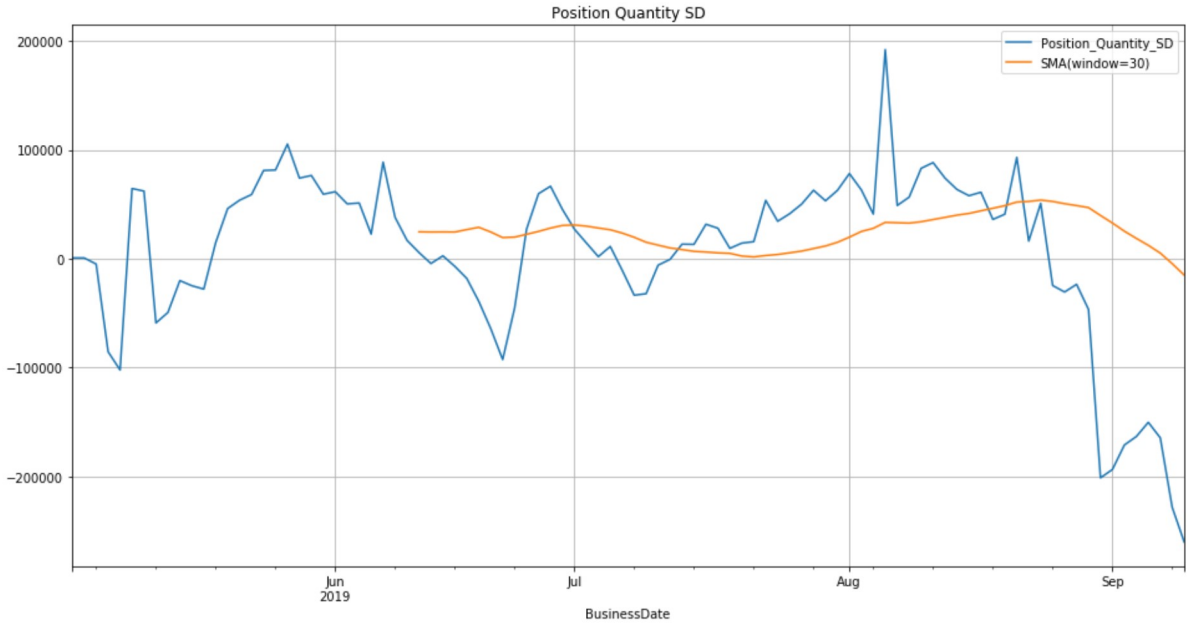
In [36]:
```
df_sedol2_edit['SMA(window=30)'] = df_sedol2_edit.Position_Quantity_SD.rolling(wind
ow=30).mean()
df_sedol2_edit.tail()
```

Out[36]:

BusinessDate	Position_Quantity_SD	SMA(window=30)
2019-09-04	-163315.0	18702.700000
2019-09-05	-150291.0	12315.900000
2019-09-06	-164515.0	5161.633333
2019-09-09	-228391.0	-4549.000000
2019-09-10	-260215.0	-14996.800000

In [37]:
```
df_sedol2_edit.plot(label='SEDOL2', figsize=(16, 8), title='Position Quantity SD',
grid=True)
```

Out[37]: <matplotlib.axes._subplots.AxesSubplot at 0x208af467ac8>



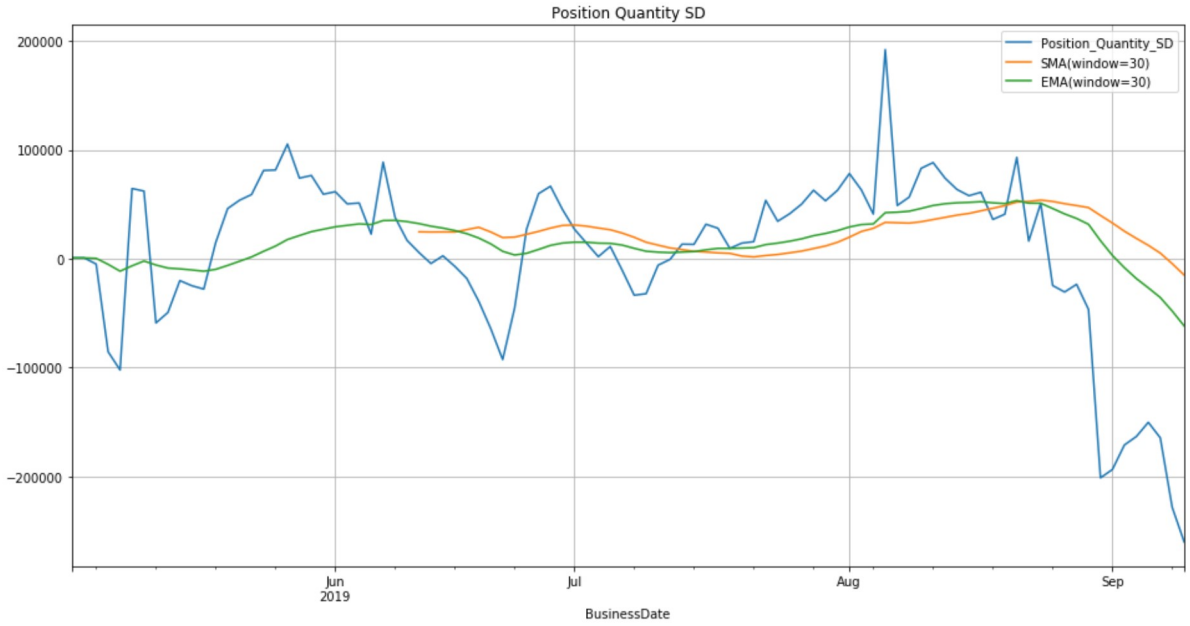
In [38]:
```
df_sedol2_edit['EMA(window=30)'] = df_sedol2_edit.Position_Quantity_SD.ewm(span=30,
adjust=False).mean()
df_sedol2_edit.tail()
```

Out[38]:

BusinessDate	Position_Quantity_SD	SMA(window=30)	EMA(window=30)
2019-09-04	-163315.0	18702.700000	-18221.328766
2019-09-05	-150291.0	12315.900000	-26741.952717
2019-09-06	-164515.0	5161.633333	-35630.536412
2019-09-09	-228391.0	-4549.000000	-48066.695354
2019-09-10	-260215.0	-14996.800000	-61753.682750

In [39]:
```
df_sedol2_edit.plot(label='SEDOL2', figsize=(16, 8), title='Position Quantity SD',
grid=True)
```

Out[39]: <matplotlib.axes._subplots.AxesSubplot at 0x208b29f4828>



SEDOL3

In [40]:
```
df_sedol3 = df.loc[df.SEDOL=='6NY9V1B']
df_sedol3.head()
```

Out[40]:

	BusinessDate	SEDOL	Counterparty_Account_ID	Position_Quantity_SD
32	2019-05-02	6NY9V1B	0130V	346000.0
33	2019-05-02	6NY9V1B	10240	-511000.0
34	2019-05-02	6NY9V1B	10240	-3281000.0
35	2019-05-02	6NY9V1B	10321	336530.0
36	2019-05-02	6NY9V1B	11009	964.0

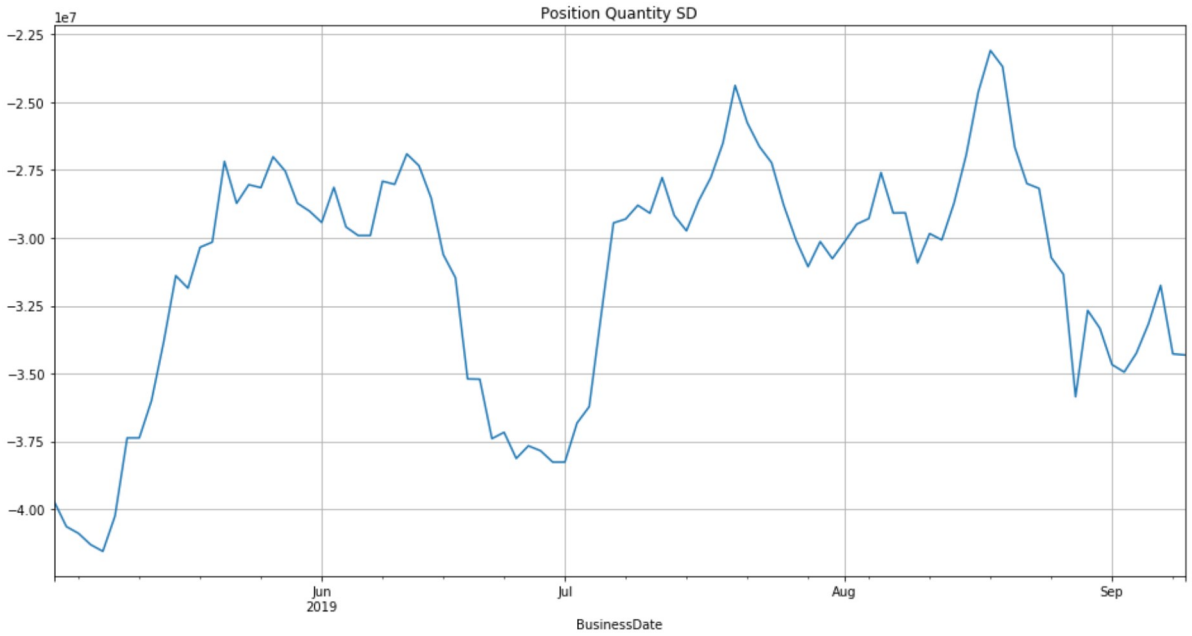
In [41]:
```
df_sedol3_edit = df_sedol3.groupby(['BusinessDate']).sum()
df_sedol3_edit.head()
```

Out[41]:

	Position_Quantity_SD
BusinessDate	
2019-05-02	-3.972417e+07
2019-05-03	-4.064518e+07
2019-05-06	-4.089717e+07
2019-05-07	-4.131317e+07
2019-05-08	-4.155817e+07

In [42]:
```
df_sedol3_edit['Position_Quantity_SD'].plot.line(label='SEDOL3', figsize=(16,8), ti
tle='Position Quantity SD', grid=True)
```

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x208b2cfcf60>



In [43]:
```
#Train the model on the last 30 days and predict the label for the 31st day
window = 30

num_samples = len(df_sedol3_edit) - window
indices = np.arange(num_samples).astype(np.int)[:,None] + np.arange(window + 1).ast
ype(np.int)
len(indices)
```

Out[43]: 64

In [44]:
```
data = df_sedol3_edit['Position_Quantity_SD'].values[indices]
data
```

Out[44]:
```
array([[-39724168.        , -40645177.81059113, -40897168.        , ...,
        -28026744.        , -26904744.        , -27339744.        ],
       [-40645177.81059113, -40897168.        , -41313168.        , ...,
        -26904744.        , -27339744.        , -28530744.        ],
       [-40897168.        , -41313168.        , -41558168.        , ...,
        -27339744.        , -28530744.        , -30616744.        ],
       ...,
       [-30067708.        , -31063708.        , -30137708.        , ...,
        -34253708.        , -33168708.        , -31753708.        ],
       [-31063708.        , -30137708.        , -30765708.        , ...,
        -33168708.        , -31753708.        , -34278708.        ],
       [-30137708.        , -30765708.        , -30144708.        , ...,
        -31753708.        , -34278708.        , -34318708.        ]])
```

In [45]:
```
X = data[:, :-1]
y = data[:, -1]
```

In [46]:
```
split_frac = 0.8
split_indices = int(split_frac * num_samples)
X_train = X[:split_indices]
y_train = y[:split_indices]
X_test = X[split_indices:]
y_test = y[split_indices:]
split_indices
```

Out[46]: 51

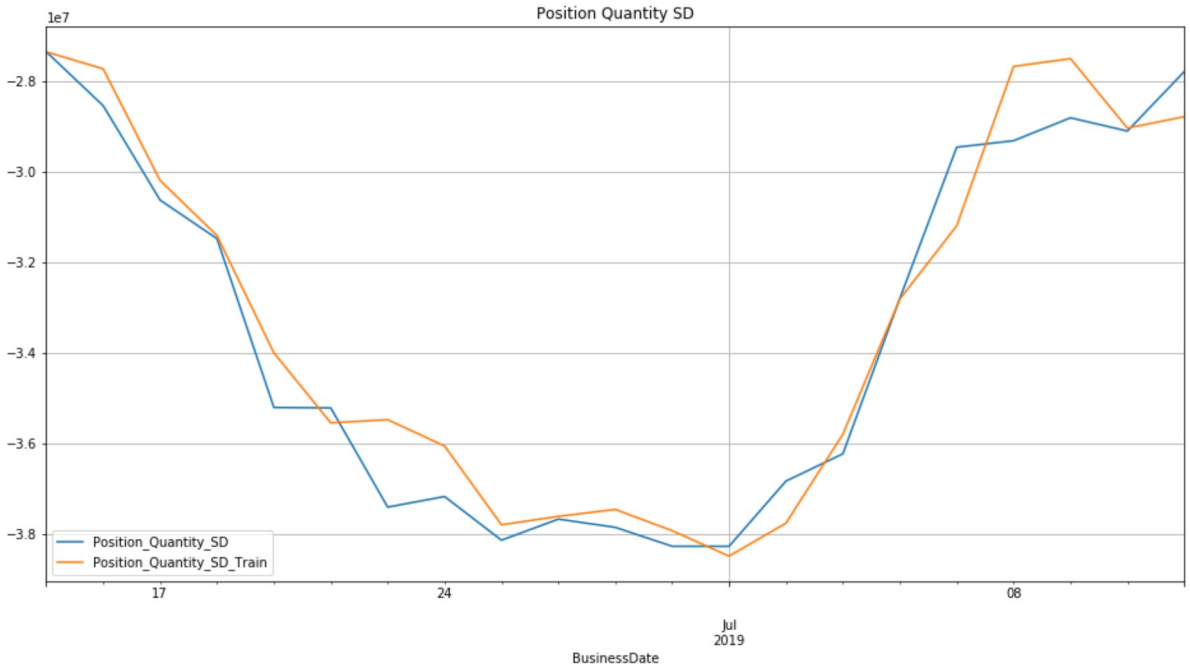
In [47]:
```
from sklearn.linear_model import LinearRegression

#Train
linear_reg_model = LinearRegression()
linear_reg_model.fit(X_train, y_train)

#Inferences
y_pred_train_linear_reg = linear_reg_model.predict(X_train)
y_pred_linear_reg = linear_reg_model.predict(X_test)
```

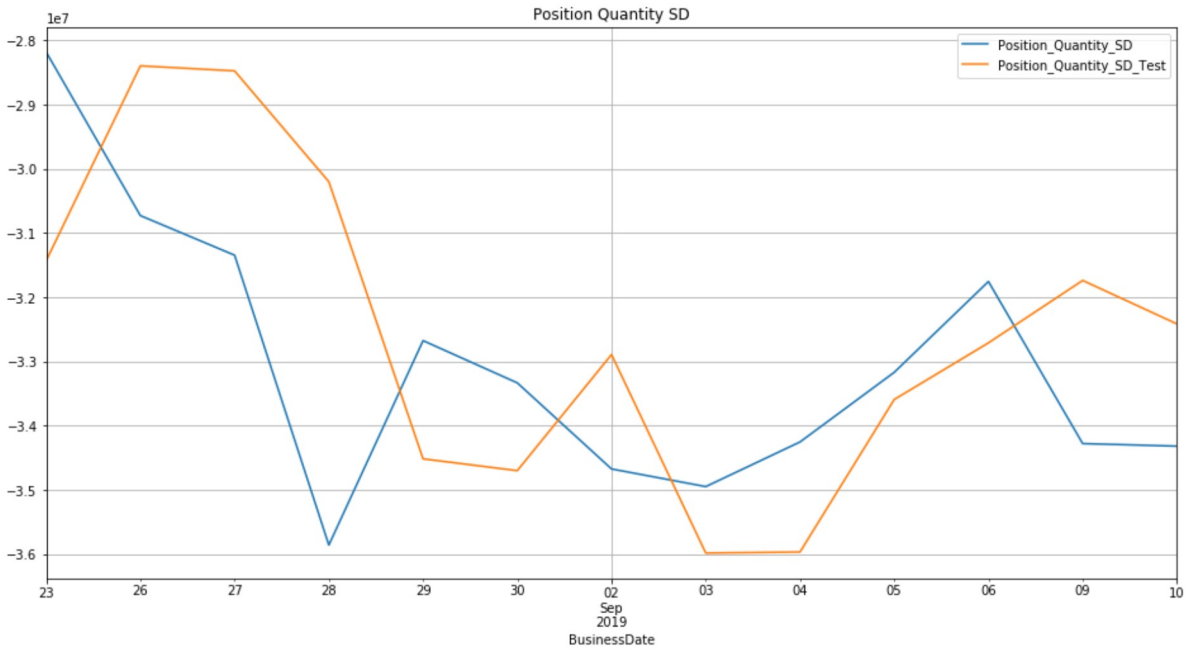
In [48]:
```
#Plot the graph for it has trained on the training data
df_linear = df_sedol3_edit.copy()
df_linear = df_linear.iloc[window:split_indices]
df_linear['Position_Quantity_SD_Train'] = y_pred_train_linear_reg[:-window]
df_linear.plot(label='SEDOL3', figsize=(16, 8), title='Position Quantity SD', grid=
True)
```

Out[48]: <matplotlib.axes._subplots.AxesSubplot at 0x208af3cb9e8>



In [49]:
```
#Plot the graph for the testing data
df_linear = df_sedol3_edit.copy()
df_linear = df_linear.iloc[split_indices+window:]
df_linear['Position_Quantity_SD_Test'] = y_pred_linear_reg
df_linear.plot(label='SEDOL3', figsize=(16, 8), title='Position Quantity SD', grid=
True)
```

Out[49]: <matplotlib.axes._subplots.AxesSubplot at 0x208b3bccf60>



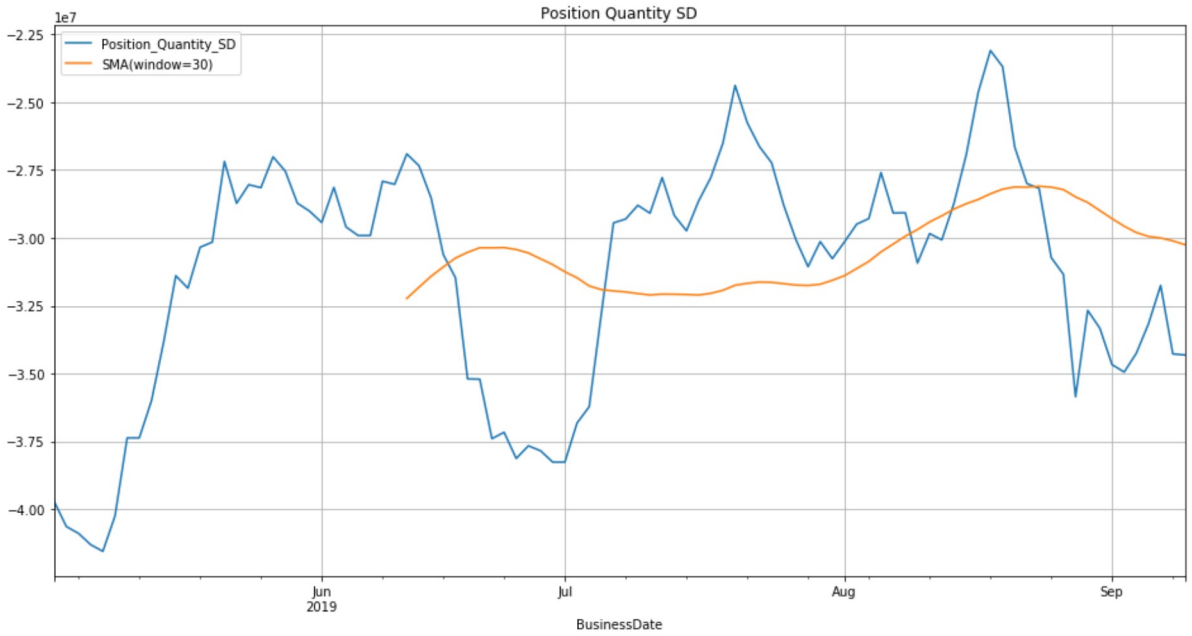
In [50]:
```
df_sedol3_edit['SMA(window=30)'] = df_sedol3_edit.Position_Quantity_SD.rolling(wind
ow=30).mean()
df_sedol3_edit.tail()
```

Out[50]:

BusinessDate	Position_Quantity_SD	SMA(window=30)
2019-09-04	-34253708.0	-2.980320e+07
2019-09-05	-33168708.0	-2.994886e+07
2019-09-06	-31753708.0	-3.000506e+07
2019-09-09	-34278708.0	-3.011223e+07
2019-09-10	-34318708.0	-3.025160e+07

In [51]:
```
df_sedol3_edit.plot(label='SEDOL3', figsize=(16, 8), title='Position Quantity SD',
grid=True)
```

Out[51]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x208b4085e48>
```



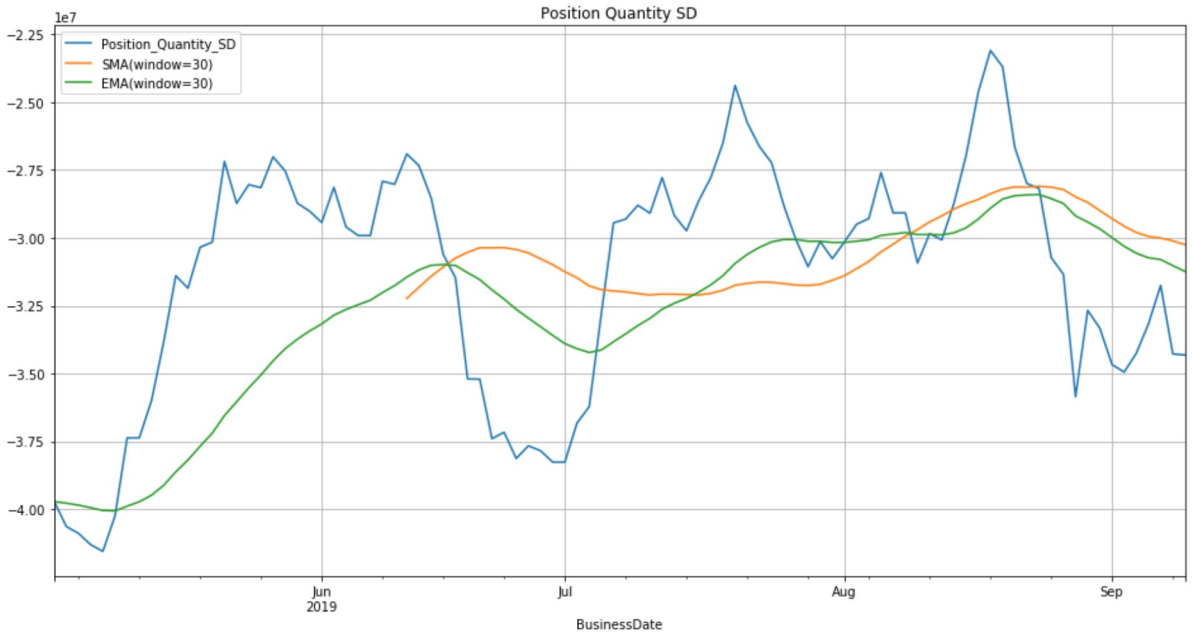
In [52]:
```
df_sedol3_edit['EMA(window=30)'] = df_sedol3_edit.Position_Quantity_SD.ewm(span=30,
adjust=False).mean()
df_sedol3_edit.tail()
```

Out[52]:

BusinessDate	Position_Quantity_SD	SMA(window=30)	EMA(window=30)
2019-09-04	-34253708.0	-2.980320e+07	-3.056616e+07
2019-09-05	-33168708.0	-2.994886e+07	-3.073407e+07
2019-09-06	-31753708.0	-3.000506e+07	-3.079985e+07
2019-09-09	-34278708.0	-3.011223e+07	-3.102430e+07
2019-09-10	-34318708.0	-3.025160e+07	-3.123684e+07

In [53]:
```
df_sedol3_edit.plot(label='SEDOL3', figsize=(16, 8), title='Position Quantity SD',
grid=True)
```

Out[53]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x208b40c5978>
```



SEDOL4

In [54]:
```
df_sedol4 = df.loc[df.SEDOL=='6Y3RH2B']
df_sedol4.head()
```

Out[54]:

	BusinessDate	SEDOL	Counterparty_Account_ID	Position_Quantity_SD
64	2019-05-02	6Y3RH2B	10240	-112000.0
65	2019-05-02	6Y3RH2B	10240	-373000.0
66	2019-05-02	6Y3RH2B	10321	173000.0
67	2019-05-02	6Y3RH2B	1310P	-202000.0
68	2019-05-02	6Y3RH2B	1310V	520000.0

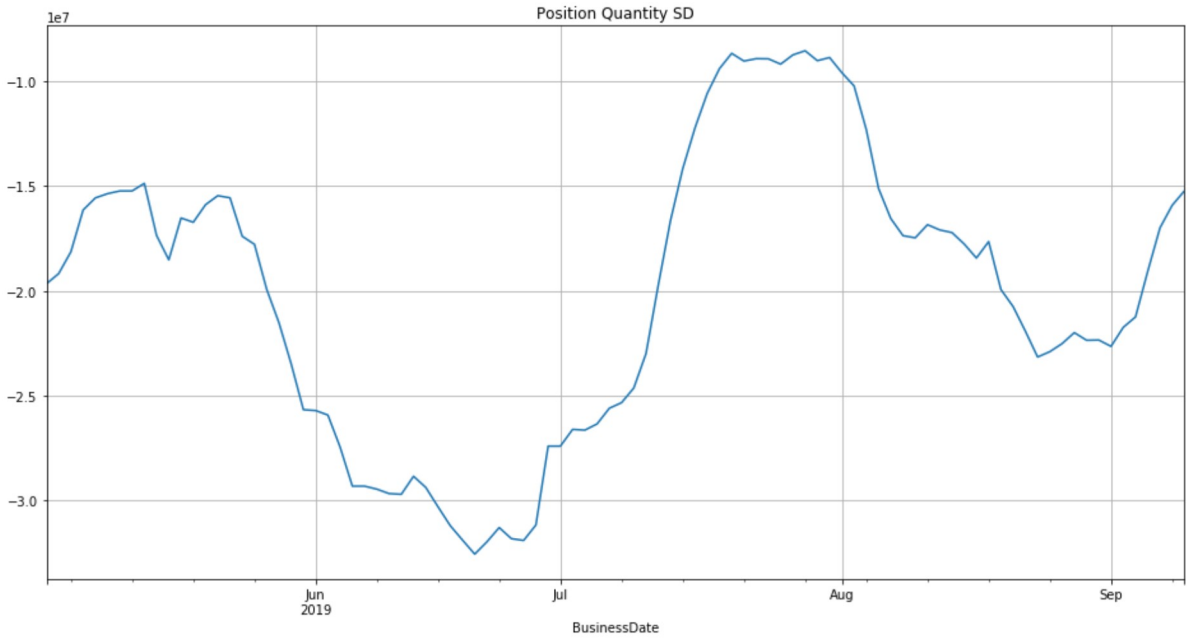
In [55]:
```
df_sedol4_edit = df_sedol4.groupby(['BusinessDate']).sum()
df_sedol4_edit.head()
```

Out[55]:

	Position_Quantity_SD
BusinessDate	
2019-05-02	-19662000.0
2019-05-03	-19176000.0
2019-05-06	-18134000.0
2019-05-07	-16142000.0
2019-05-08	-15565000.0

In [56]:
```
df_sedol4_edit['Position_Quantity_SD'].plot.line(label='SEDOL4', figsize=(16,8), ti
tle='Position Quantity SD', grid=True)
```

Out[56]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x208b4585e80>
```



In [57]:
```
#Train the model on the last 30 days and predict the label for the 31st day
window = 30

num_samples = len(df_sedol4_edit) - window
indices = np.arange(num_samples).astype(np.int)[:,None] + np.arange(window + 1).ast
ype(np.int)
len(indices)
```

Out[57]:
```
64
```

In [58]:
```
data = df_sedol4_edit['Position_Quantity_SD'].values[indices]
data
```

Out[58]:
```
array([[-19662000., -19176000., -18134000., ..., -29666000., -29700000.,
        -28841000.],
       [-19176000., -18134000., -16142000., ..., -29700000., -28841000.,
        -29372000.],
       [-18134000., -16142000., -15565000., ..., -28841000., -29372000.,
        -30296000.],
       ...,
       [ -8747000.,  -8549000.,  -9023000., ..., -21241000., -19065000.,
        -16994000.],
       [ -8549000.,  -9023000.,  -8872000., ..., -19065000., -16994000.,
        -15916000.],
       [ -9023000.,  -8872000.,  -9592000., ..., -16994000., -15916000.,
        -15230000.]])
```

In [59]:
```
X = data[:, :-1]
y = data[:, -1]
```

In [60]:
```
split_frac = 0.8
split_indices = int(split_frac * num_samples)
X_train = X[:split_indices]
y_train = y[:split_indices]
X_test = X[split_indices:]
y_test = y[split_indices:]
split_indices
```

Out[60]: 51

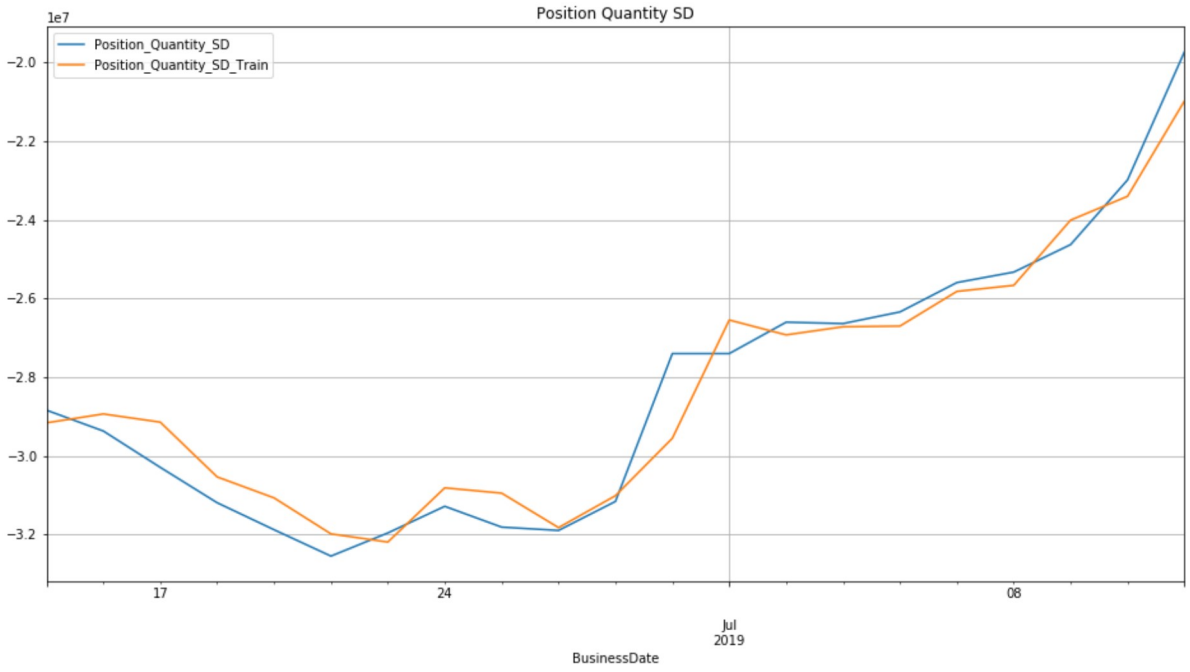
In [61]:
```
from sklearn.linear_model import LinearRegression

#Train
linear_reg_model = LinearRegression()
linear_reg_model.fit(X_train, y_train)

#Inferences
y_pred_train_linear_reg = linear_reg_model.predict(X_train)
y_pred_linear_reg = linear_reg_model.predict(X_test)
```

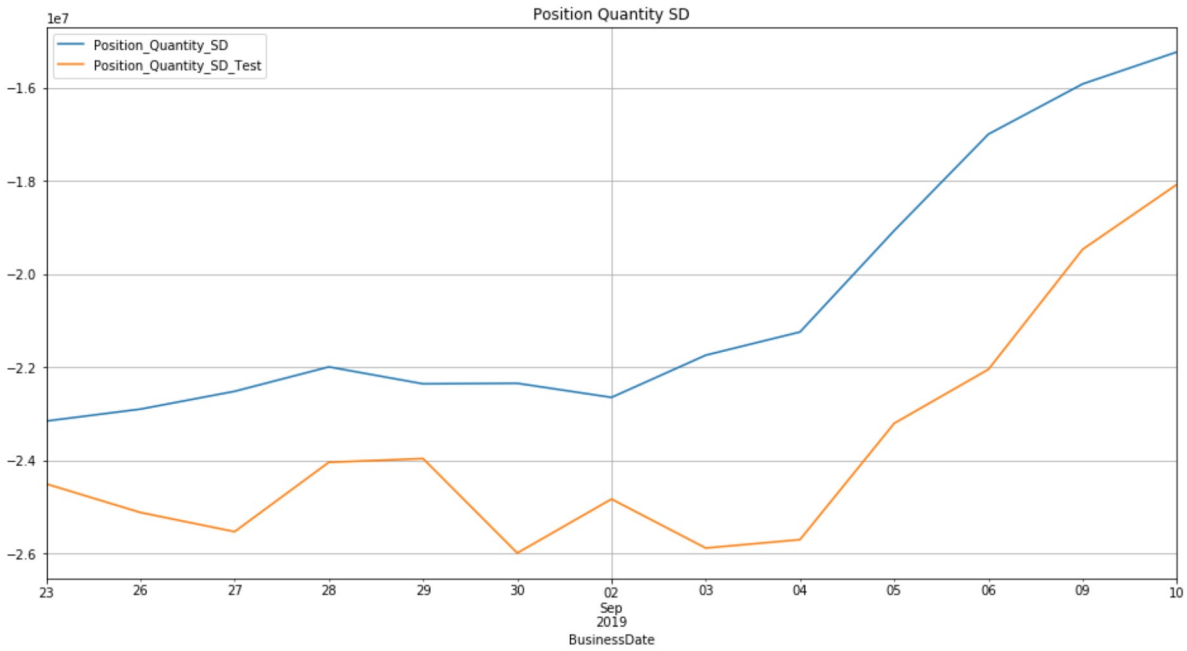
In [62]:
```
#Plot the graph for it has trained on the training data
df_linear = df_sedol4_edit.copy()
df_linear = df_linear.iloc[window:split_indices]
df_linear['Position_Quantity_SD_Train'] = y_pred_train_linear_reg[:-window]
df_linear.plot(label='SEDOL4', figsize=(16, 8), title='Position Quantity SD', grid=
True)
```

Out[62]: <matplotlib.axes._subplots.AxesSubplot at 0x208b45f76a0>



In [63]:
```
#Plot the graph for testing data
df_linear = df_sedol4_edit.copy()
df_linear = df_linear.iloc[split_indices+window:]
df_linear['Position_Quantity_SD_Test'] = y_pred_linear_reg
df_linear.plot(label='SEDOL4', figsize=(16, 8), title='Position Quantity SD', grid=
True)
```

Out[63]: <matplotlib.axes._subplots.AxesSubplot at 0x208b5a270f0>



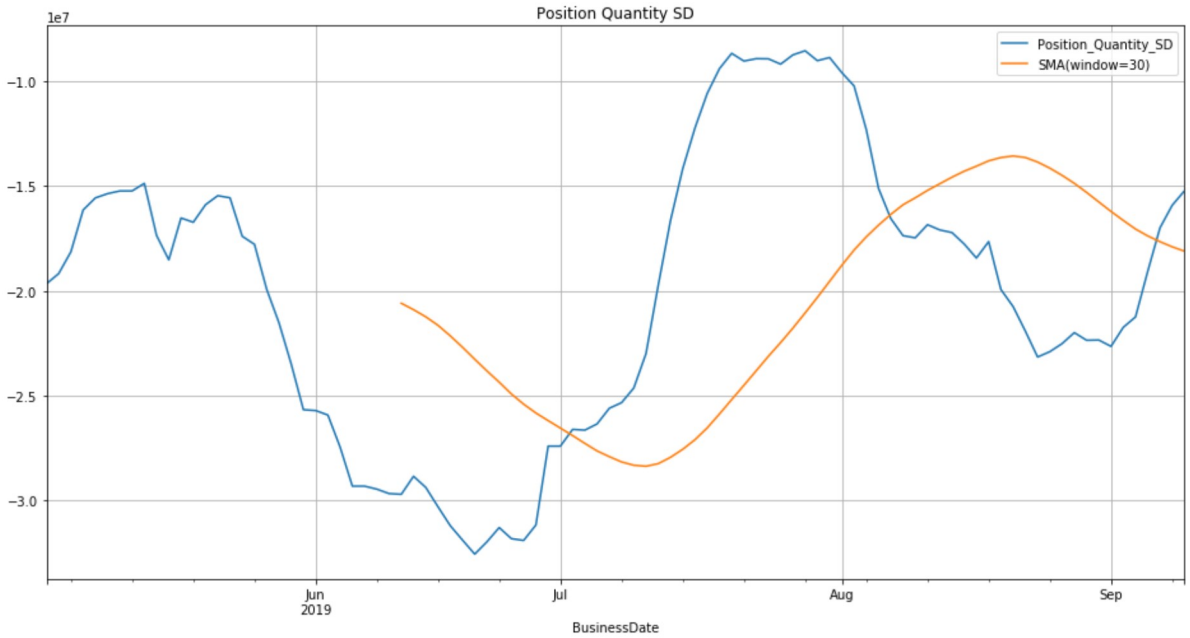
In [64]:
```
df_sedol4_edit['SMA(window=30)'] = df_sedol4_edit.Position_Quantity_SD.rolling(wind
ow=30).mean()
df_sedol4_edit.tail()
```

Out[64]:

BusinessDate	Position_Quantity_SD	SMA(window=30)
2019-09-04	-21241000.0	-1.704940e+07
2019-09-05	-19065000.0	-1.737863e+07
2019-09-06	-16994000.0	-1.765353e+07
2019-09-09	-15916000.0	-1.789910e+07
2019-09-10	-15230000.0	-1.810600e+07

In [65]:
```
df_sedol4_edit.plot(label='SEDOL4', figsize=(16, 8), title='Position Quantity SD',
grid=True)
```

Out[65]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x208b5abd978>
```



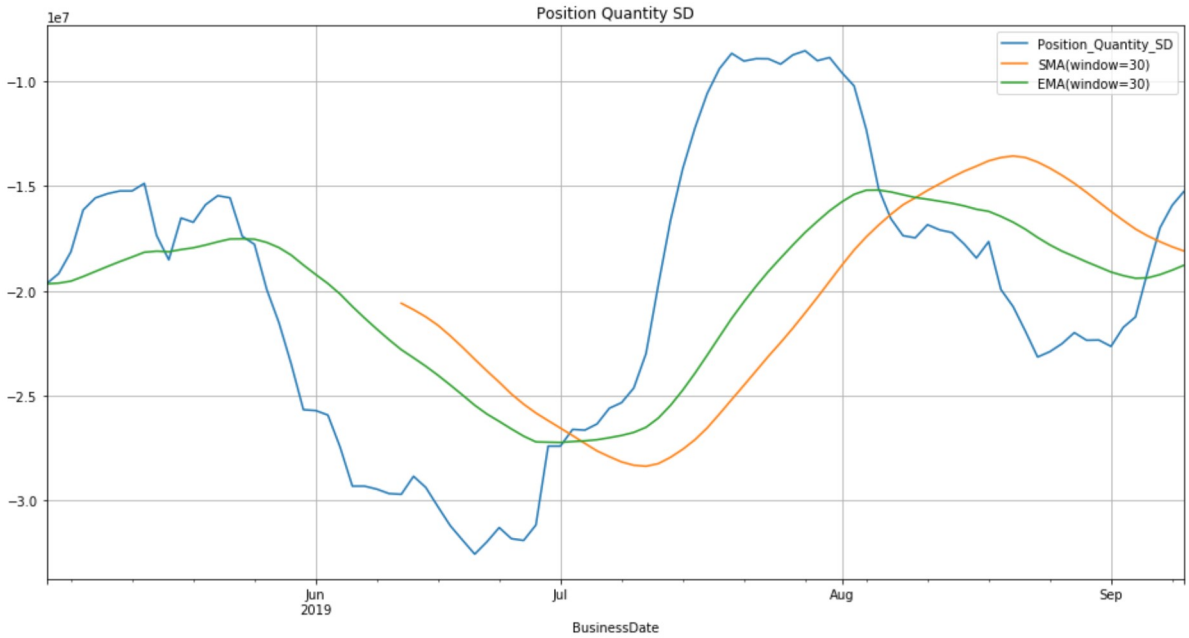
In [66]:
```
df_sedol4_edit['EMA(window=30)'] = df_sedol4_edit.Position_Quantity_SD.ewm(span=30,
adjust=False).mean()
df_sedol4_edit.tail()
```

Out[66]:

BusinessDate	Position_Quantity_SD	SMA(window=30)	EMA(window=30)
2019-09-04	-21241000.0	-1.704940e+07	-1.939856e+07
2019-09-05	-19065000.0	-1.737863e+07	-1.937704e+07
2019-09-06	-16994000.0	-1.765353e+07	-1.922329e+07
2019-09-09	-15916000.0	-1.789910e+07	-1.900992e+07
2019-09-10	-15230000.0	-1.810600e+07	-1.876605e+07

In [67]:
```
df_sedol4_edit.plot(label='SEDOL4', figsize=(16, 8), title='Position Quantity SD',
grid=True)
```

Out[67]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x208b5f61b38>
```



SEDOL5

In [68]:
```
df_sedol5 = df.loc[df.SEDOL=='74ZI41B']
df_sedol5.head()
```

Out[68]:

	BusinessDate	SEDOL	Counterparty_Account_ID	Position_Quantity_SD
87	2019-05-02	74ZI41B	1003V	697000.0
88	2019-05-02	74ZI41B	10240	4000.0
89	2019-05-02	74ZI41B	10240	557000.0
90	2019-05-02	74ZI41B	10321	433000.0
91	2019-05-02	74ZI41B	12210	164000.0

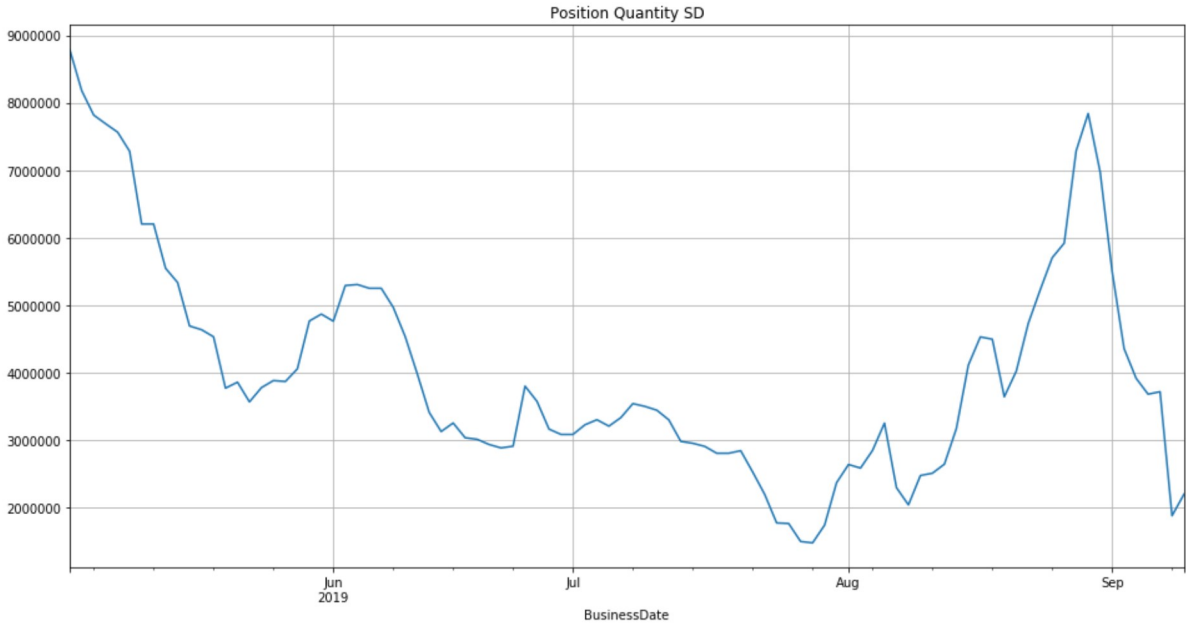
In [69]:
```
df_sedol5_edit = df_sedol5.groupby(['BusinessDate']).sum()
df_sedol5_edit.head()
```

Out[69]:

BusinessDate	Position_Quantity_SD
2019-05-02	8785907.0
2019-05-03	8177907.0
2019-05-06	7815907.0
2019-05-07	7686907.0
2019-05-08	7561907.0

In [70]:
```
df_sedol5_edit['Position_Quantity_SD'].plot.line(label='SEDOL5', figsize=(16,8), title='Position Quantity SD', grid=True)
```

Out[70]: <matplotlib.axes._subplots.AxesSubplot at 0x208b2ce1cc0>



In [71]:
```
#Train the model on the last 30 days and predict the label for the 31st day
window = 30

num_samples = len(df_sedol5_edit) - window
indices = np.arange(num_samples).astype(np.int)[:,None] + np.arange(window + 1).astype(np.int)
len(indices)
```

Out[71]: 64

In [72]:
```
data = df_sedol5_edit['Position_Quantity_SD'].values[indices]
data
```

Out[72]:
```
array([[8785907., 8177907., 7815907., ..., 4529907., 3982907., 3407907.],
       [8177907., 7815907., 7686907., ..., 3982907., 3407907., 3123907.],
       [7815907., 7686907., 7561907., ..., 3407907., 3123907., 3250907.],
       ...,
       [1493907., 1472907., 1735907., ..., 3913907., 3677907., 3713907.],
       [1472907., 1735907., 2364907., ..., 3677907., 3713907., 1874907.],
       [1735907., 2364907., 2634907., ..., 3713907., 1874907., 2195907.]])
```

In [73]:
```
X = data[:, :-1]
y = data[:, -1]
```

In [74]:
```
split_frac = 0.8
split_indices = int(split_frac * num_samples)
X_train = X[:split_indices]
y_train = y[:split_indices]
X_test = X[split_indices:]
y_test = y[split_indices:]
split_indices
```

Out[74]: 51

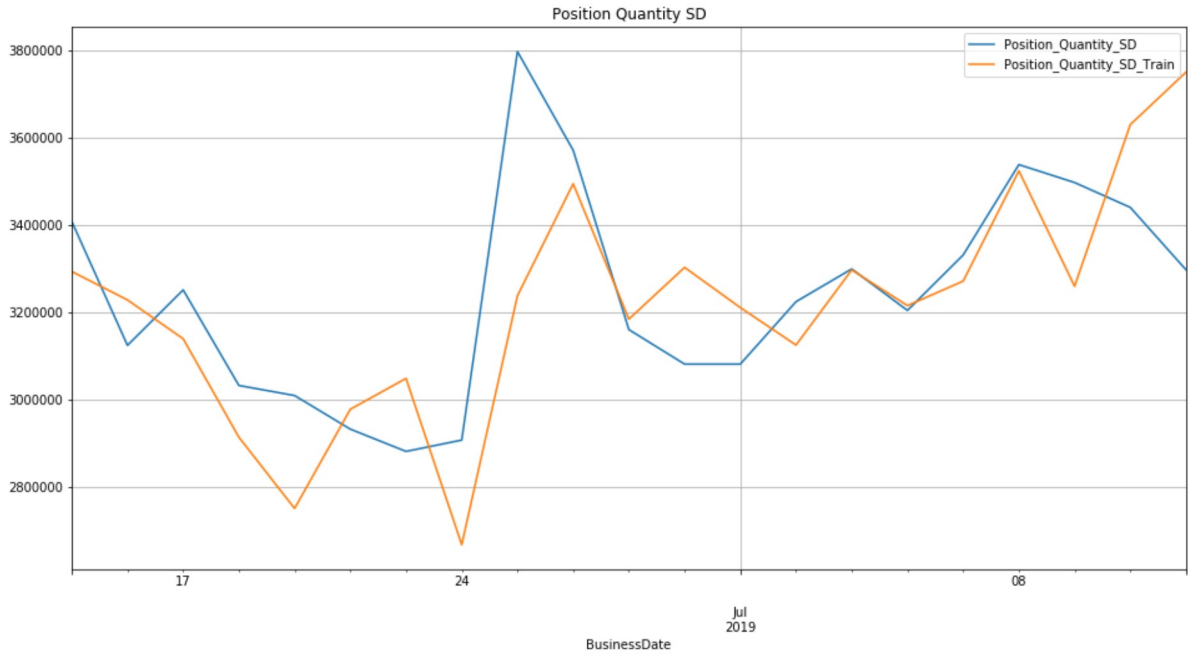
In [75]:
```
from sklearn.linear_model import LinearRegression

#Train
linear_reg_model = LinearRegression()
linear_reg_model.fit(X_train, y_train)

#Inferences
y_pred_train_linear_reg = linear_reg_model.predict(X_train)
y_pred_linear_reg = linear_reg_model.predict(X_test)
```

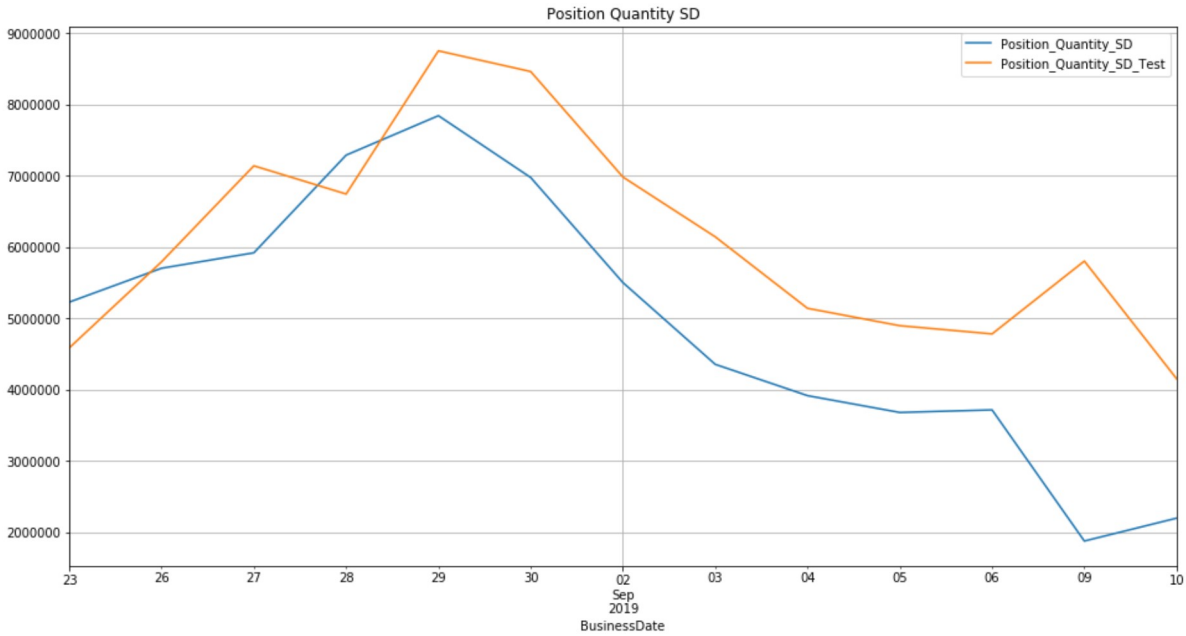
In [76]:
```
#Plot the graph for it has trained on the training data
df_linear = df_sedol5_edit.copy()
df_linear = df_linear.iloc[window:split_indices]
df_linear['Position_Quantity_SD_Train'] = y_pred_train_linear_reg[:-window]
df_linear.plot(label='SEDOL5', figsize=(16, 8), title='Position Quantity SD', grid=
True)
```

Out[76]: <matplotlib.axes._subplots.AxesSubplot at 0x208b672ba90>



In [77]:
```
#Plot the graph for the testing data
df_linear = df_sedol5_edit.copy()
df_linear = df_linear.iloc[split_indices+window:]
df_linear['Position_Quantity_SD_Test'] = y_pred_linear_reg
df_linear.plot(label='SEDOL5', figsize=(16, 8), title='Position Quantity SD', grid=
True)
```

Out[77]: <matplotlib.axes._subplots.AxesSubplot at 0x208b6496f60>



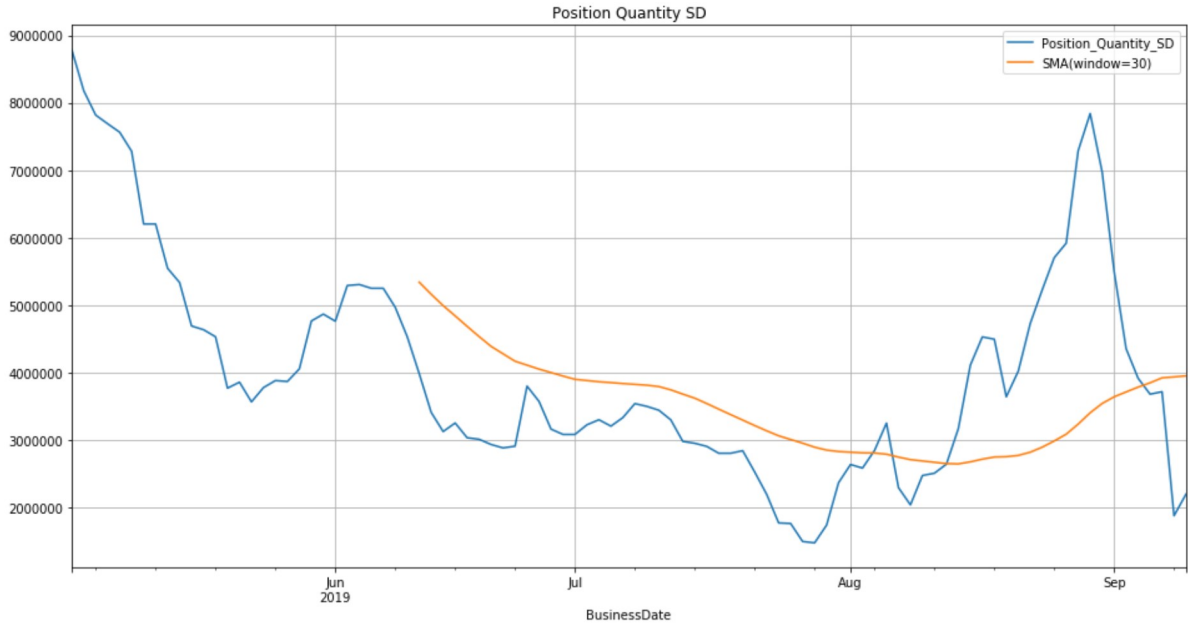
In [78]:
```
df_sedol5_edit['SMA(window=30)'] = df_sedol5_edit.Position_Quantity_SD.rolling(wind
ow=30).mean()
df_sedol5_edit.tail()
```

Out[78]:

BusinessDate	Position_Quantity_SD	SMA(window=30)
2019-09-04	3913907.0	3.782407e+06
2019-09-05	3677907.0	3.846307e+06
2019-09-06	3713907.0	3.920307e+06
2019-09-09	1874907.0	3.933707e+06
2019-09-10	2195907.0	3.949040e+06

In [79]:
```
df_sedol5_edit.plot(label='SEDOL5', figsize=(16, 8), title='Position Quantity SD',
grid=True)
```

Out[79]: <matplotlib.axes._subplots.AxesSubplot at 0x208b6986390>



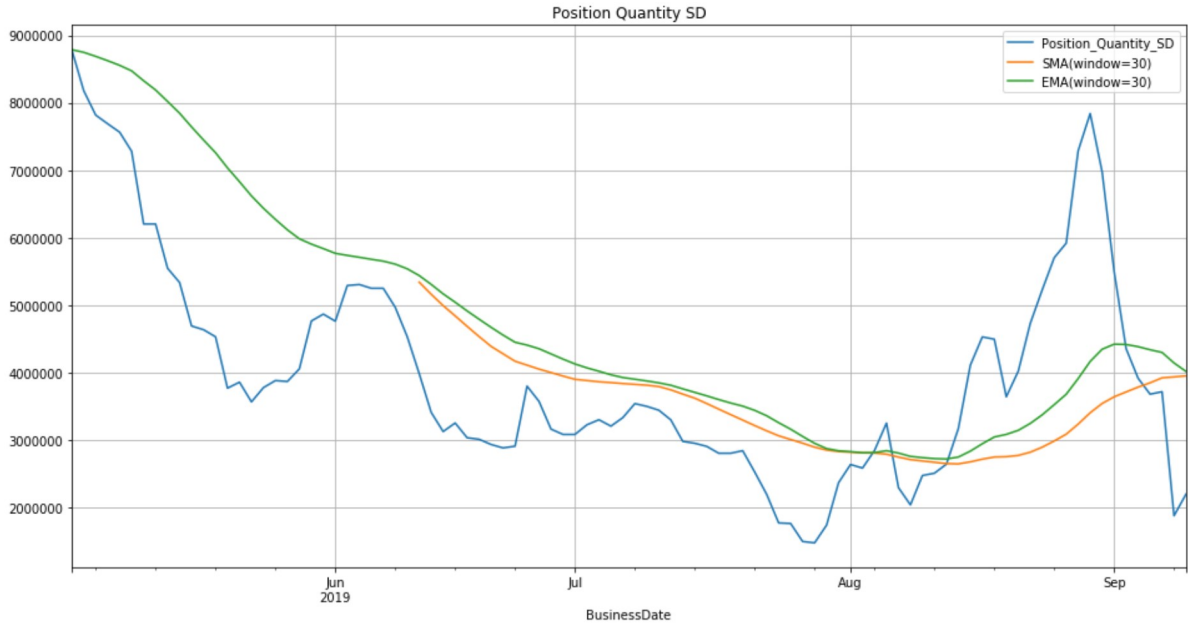
In [80]:
```
df_sedol5_edit['EMA(window=30)'] = df_sedol5_edit.Position_Quantity_SD.ewm(span=30,
adjust=False).mean()
df_sedol5_edit.tail()
```

Out[80]:

BusinessDate	Position_Quantity_SD	SMA(window=30)	EMA(window=30)
2019-09-04	3913907.0	3.782407e+06	4.382346e+06
2019-09-05	3677907.0	3.846307e+06	4.336899e+06
2019-09-06	3713907.0	3.920307e+06	4.296706e+06
2019-09-09	1874907.0	3.933707e+06	4.140461e+06
2019-09-10	2195907.0	3.949040e+06	4.015006e+06

In [81]:
```
df_sedol5_edit.plot(label='SEDOL5', figsize=(16, 8), title='Position Quantity SD',
grid=True)
```

Out[81]: <matplotlib.axes._subplots.AxesSubplot at 0x208b6986940>



In [ ]: