

In [7]:

```

1 from pandas import read_csv
2 from pandas import datetime
3 from pandas import DataFrame
4 from pandas import concat
5 from matplotlib import pyplot
6 from sklearn.metrics import mean_squared_error
7 import pandas as pd

```

C:\Users\Laptop\Anaconda3\lib\site-packages\ipykernel\_launcher.py:2: FutureWarning: The pandas.datetime class is deprecated and will be removed from pandas in a future version. Import from datetime module instead.

In [6]:

```

1 dane = pd.read_csv('C:/Users/Laptop/Desktop/Grypa/Dane/total.csv', index_col='data')
2 dane1 = dane[(dane.index > '2012-08-16' ) & (dane.index < '2020-03-16')]
3
4 len(dane1)
5 series = dane1['Total']
6 series = concat([series.shift(1), series], axis=1)
7 series.columns = ['t-1', 't+1']
8 #del series['data']
9 series

```

Out[6]:

	t-1	t+1
data		
2012-08-23	NaN	9932
2012-09-01	9932.0	10530
2012-09-08	10530.0	14375
2012-09-16	14375.0	23150
2012-09-23	23150.0	27864
...	...	...
2020-02-08	193482.0	200194
2020-02-16	200194.0	211983
2020-02-23	211983.0	215994
2020-03-01	215994.0	211572
2020-03-08	211572.0	158677

363 rows × 2 columns

In [13]:

```
1 X=series.values
2 X
```

Out[13]:

```
array([[ nan,   9932.],
       [ 9932., 10530.],
       [10530., 14375.],
       [14375., 23150.],
       [23150., 27864.],
       [27864., 34232.],
       [34232., 45357.],
       [45357., 43668.],
       [43668., 56433.],
       [56433., 46712.],
       [46712., 50854.],
       [50854., 47865.],
       [47865., 58867.],
       [58867., 58130.],
       [58130., 61968.],
       [61968., 62147.],
       [62147., 78779.],
       [78779., 134358.]])
```

In [37]:

```
1 #series
```

In [9]:

```
1 X= series.values
2 train, test = X[1:267], X[267:]
3 #train
```

In [10]:

```
1 train_x, train_y = train[:,0], train[:,1]
```

In [11]:

```
1 test_x, test_y = test[:,0], test[:,1]
```

In [32]:

```
1 #train_x
```

In [33]:

```
1 #train_y
```

In [34]:

```
1 #test_x
```

In [35]:

```
1 #test_y
```

In [26]:

```
1 def model_persistence(x):  
2     return x
```

In [30]:

```
1 predictions=list()  
2 for x in train_x:  
3     wynik = model_persistence(x)  
4     predictions.append(wynik)  
5 test_score = mean_squared_error(train_y, predictions)  
6 print('Residual: %.3f' % test_score)
```

Residual: 414751537.086

In [31]:

```
1 predictions=list()  
2 for x in test_x:  
3     wynik = model_persistence(x)  
4     predictions.append(wynik)  
5 test_score = mean_squared_error(test_y, predictions)  
6 print('MSE: %.3f' % test_score)
```

MSE: 480572958.646