Praktikum Data Preprocessing

Predictive Mining for Time-Series Data

Ulima Inas Shabrina(2110181048)

Assignment

1. **dataset** ← transaction.csv, dan tampilkan
2. **data** ← ambillah data pada **dataset** untuk fitur Qty, Country ("Germany"), month, year ("2011")
3. **TotalQty** ← ambillah month dari **data** dan akumulasi Qty pada bulan yang sama, dan tampilkan
4. Visualisasikan pergerakan nilai **TotalQty** dimana sumbu x=month dan sumbu y=total Qty
5. PredictedQty ← prediksikan total Qty dari **TotalQty** pada bulan Januari 2012 dengan Linear Regression

In [1]:
```
import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

#1 tampilkan transaction.csv
dataset = pd.read_csv("transaction.csv")
dataset
```

Out[1]:

	InvoiceNo	StockCode	Qty	InvoiceDate	CustomerID	Country
0	537626	22725	830	12/7/2010 14:57	12347	Iceland
1	537626	22729	948	12/7/2010 14:57	12347	Iceland
2	537626	22195	695	12/7/2010 14:57	12347	Iceland
3	542237	22725	636	1/26/2011 14:30	12347	Iceland
4	542237	22729	536	1/26/2011 14:30	12347	Iceland
...	...	...	...	...	...	...
10541	543911	21700	455	2/14/2011 12:46	17829	United Arab Emirates
10542	543911	22111	578	2/14/2011 12:46	17829	United Arab Emirates
10543	543911	22112	163	2/14/2011 12:46	17829	United Arab Emirates
10544	564428	23296	545	8/25/2011 11:27	17844	Canada
10545	564428	23294	643	8/25/2011 11:27	17844	Canada

10546 rows × 6 columns

In [5]:
```
#2 ambillah data pada dataset untuk fitur Qty, Country ("Germany"), month, year ("2011")

dataset['InvoiceDate'] = pd.to_datetime(dataset['InvoiceDate'])
data = dataset[(dataset['InvoiceDate'].dt.year == 2011) & (dataset['Country'] == 'Germany')]
data['Month'] = data['InvoiceDate'].dt.month
data['Year'] = data['InvoiceDate'].dt.year
data = pd.DataFrame(data, columns=['Qty', 'Country', 'Month', 'Year'])
data
```

	Qty	Country	Month	Year
1185	628	Germany	5	2011
1186	981	Germany	5	2011
1187	212	Germany	5	2011
1188	910	Germany	5	2011
1189	668	Germany	5	2011
...	...	...	...	...
8339	562	Germany	9	2011
8340	692	Germany	9	2011
8341	400	Germany	9	2011
8342	769	Germany	11	2011
8343	842	Germany	11	2011

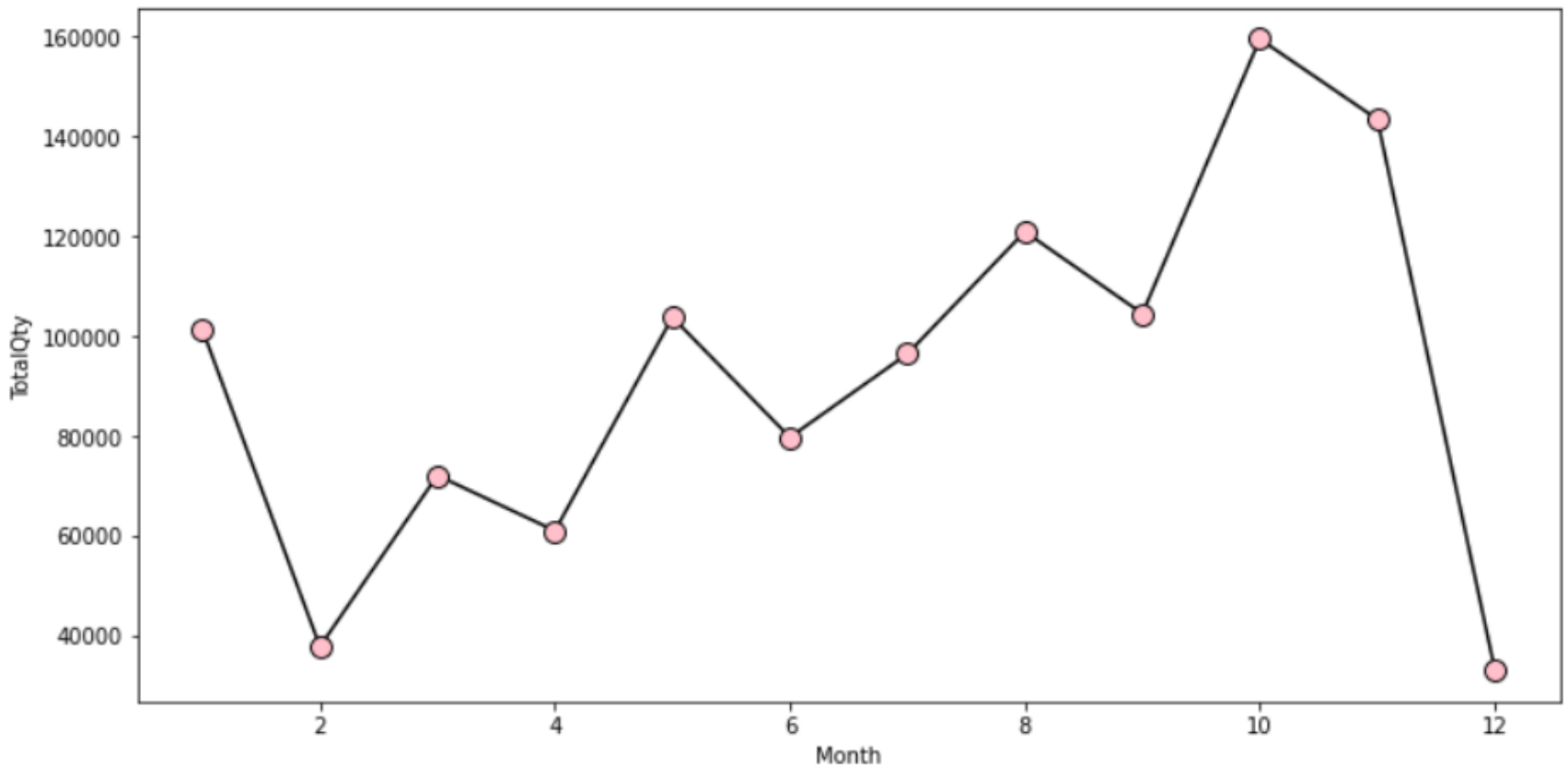
2148 rows × 4 columns

In [3]:
```
#3 ambillah month dari data dan akumulasi Qty pada bulan yang sama, dan tampilkan

data['InvoiceDate'] = pd.to_datetime(data['InvoiceDate'])
totalQty = data.groupby(data['InvoiceDate'].dt.month)['Qty'].sum().reset_index()
totalQty.columns = ['Month', 'TotalQty']
totalQty
```

Out[3]:

	Month	TotalQty
0	1	101266
1	2	37800
2	3	72084
3	4	60993
4	5	103749
5	6	79711
6	7	96423
7	8	120908
8	9	104487
9	10	159490
10	11	143409
11	12	33126

In [4]:
```
#4 Visualisasikan pergerakan nilai TotalQty dimana sumbu x=month dan sumbu y=total Qty

plt.figure(figsize=(12,6))
plt.plot(totalQty['Month'], totalQty['TotalQty'], color = 'black', marker='o', markerfacecolor = 'pink', markersize=10)
plt.xlabel('Month')
plt.ylabel('TotalQty')
plt.show()
```



In [5]:
```
#5 prediksikan total Qty dari TotalQty pada bulan Januari 2012 dengan Linear Regression
linearRegression = LinearRegression()
X = np.array(totalQty['Month']).reshape(-1,1)
Y = np.array(totalQty['TotalQty'])
linearRegression.fit(X,Y)
predictedQty = linearRegression.predict([[13]])
print('Hasil Prediksi bulan January 2012 (bulan ke 13) = ', str(predictedQty))
```

```
Hasil Prediksi bulan January 2012 (bulan ke 13) =  [117751.75757576]
```

Terimakasih