Online Retail

In [1]:

```
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
%matplotlib inline
```

K-means clustering is one of the simplest and popular unsupervised machine learning algorithms.

The algorithm works as follows:

First we initialize k points, called means, randomly. We categorize each item to its closest mean and we update the mean’s coordinates, which are the averages of the items categorized in that mean so far. We repeat the process for a given number of iterations and at the end, we have our clusters.

The data made by a UK-based, registered, non-store online,retailer between December 1, 2010, and December 9,2011,area allincluded in the transnational data set known as online retail. The company primarily offersone-of-a-kind gifts for every occasion. The companyhas a large number of wholesalers as clients.CompanyObjectiveUsing the global online retail dataset, we willdesign a clustering model and select the ideal groupof clients for the business to target. ¶

In [2]:

```
df=pd.read_csv(r"C:\Users\RAMADEVI SURIPAKA\OneDrive\Documents\online retail.csv")
df.head()
```

Out[2]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	01-12-2010 08:26	2.55	17850.0	Unitec Kingdom
1	536365	71053	WHITE METAL LANTERN	6	01-12-2010 08:26	3.39	17850.0	Unitec Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	01-12-2010 08:26	2.75	17850.0	Unitec Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	01-12-2010 08:26	3.39	17850.0	Unitec Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	01-12-2010 08:26	3.39	17850.0	Unitec Kingdom

In [3]:

```
df.head()
```

Out[3]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	01-12-2010 08:26	2.55	17850.0	Unitec Kingdom
1	536365	71053	WHITE METAL LANTERN	6	01-12-2010 08:26	3.39	17850.0	Unitec Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	01-12-2010 08:26	2.75	17850.0	Unitec Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	01-12-2010 08:26	3.39	17850.0	Unitec Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	01-12-2010 08:26	3.39	17850.0	Unitec Kingdom

In [4]:

```
df.tail()
```

Out[4]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	09-12-2011 12:50	0.85	12680.0
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	09-12-2011 12:50	2.10	12680.0
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	09-12-2011 12:50	4.15	12680.0
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	09-12-2011 12:50	4.15	12680.0
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	09-12-2011 12:50	4.95	12680.0

In [5]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype  
---  ------       --------------   -----  
 0   InvoiceNo    541909 non-null  object 
 1   StockCode    541909 non-null  object 
 2   Description  540455 non-null  object 
 3   Quantity     541909 non-null  int64  
 4   InvoiceDate  541909 non-null  object 
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   406829 non-null  float64
 7   Country      541909 non-null  object 
dtypes: float64(2), int64(1), object(5)
memory usage: 33.1+ MB
```

In [6]:

```
df.describe()
```

Out[6]:

	Quantity	UnitPrice	CustomerID
count	541909.000000	541909.000000	406829.000000
mean	9.552250	4.611114	15287.690570
std	218.081158	96.759853	1713.600303
min	-80995.000000	-11062.060000	12346.000000
25%	1.000000	1.250000	13953.000000
50%	3.000000	2.080000	15152.000000
75%	10.000000	4.130000	16791.000000
max	80995.000000	38970.000000	18287.000000

In [7]:

```
df.shape
```

Out[7]:

```
(541909, 8)
```

In [8]:

```
df.count
```

Out[8]:

```
<bound method DataFrame.count of         InvoiceNo StockCode
Description  Quantity
0         536365    85123A   WHITE HANGING HEART T-LIGHT HOLDER         6
\
1         536365     71053                  WHITE METAL LANTERN         6
2         536365    84406B       CREAM CUPID HEARTS COAT HANGER         8
3         536365    84029G  KNITTED UNION FLAG HOT WATER BOTTLE         6
4         536365    84029E       RED WOOLLY HOTTIE WHITE HEART.         6
...          ...       ...                                  ...       ...
541904    581587     22613          PACK OF 20 SPACEBOY NAPKINS        12
541905    581587     22899         CHILDREN'S APRON DOLLY GIRL          6
541906    581587     23254        CHILDRENS CUTLERY DOLLY GIRL          4
541907    581587     23255      CHILDRENS CUTLERY CIRCUS PARADE         4
541908    581587     22138        BAKING SET 9 PIECE RETROSPOT          3

             InvoiceDate  UnitPrice  CustomerID         Country
0       01-12-2010 08:26       2.55     17850.0  United Kingdom
1       01-12-2010 08:26       3.39     17850.0  United Kingdom
2       01-12-2010 08:26       2.75     17850.0  United Kingdom
3       01-12-2010 08:26       3.39     17850.0  United Kingdom
4       01-12-2010 08:26       3.39     17850.0  United Kingdom
...                  ...        ...         ...             ...
541904  09-12-2011 12:50       0.85     12680.0          France
541905  09-12-2011 12:50       2.10     12680.0          France
541906  09-12-2011 12:50       4.15     12680.0          France
541907  09-12-2011 12:50       4.15     12680.0          France
541908  09-12-2011 12:50       4.95     12680.0          France

[541909 rows x 8 columns]>
```

In [9]:

```
df.isna().sum()
```

Out[9]:

```
InvoiceNo           0
StockCode           0
Description      1454
Quantity            0
InvoiceDate         0
UnitPrice           0
CustomerID     135080
Country             0
dtype: int64
```

In [17]:

```
df.fillna(method="ffill",inplace=True)
```

In [18]:

```
df.isna().sum()
```

Out[18]:

```
InvoiceNo      0
StockCode      0
Description    0
Quantity       0
InvoiceDate    0
UnitPrice      0
CustomerID     0
Country        0
dtype: int64
```

In [19]:

```
df['InvoiceNo'].value_counts()
```

Out[19]:

```
InvoiceNo
573585     1114
581219      749
581492      731
580729      721
558475      705
           ... 
554023        1
554022        1
554021        1
554020        1
C558901       1
Name: count, Length: 25900, dtype: int64
```

In [20]:

```
df['CustomerID'].value_counts()
```

Out[20]:

```
CustomerID
17841.0    8644
14911.0    7648
12748.0    6134
14096.0    5412
14606.0    3952
           ... 
15753.0       1
14424.0       1
15562.0       1
13302.0       1
17331.0       1
Name: count, Length: 4372, dtype: int64
```

In [21]:

```
df['Quantity'].value_counts()
```

Out[21]:

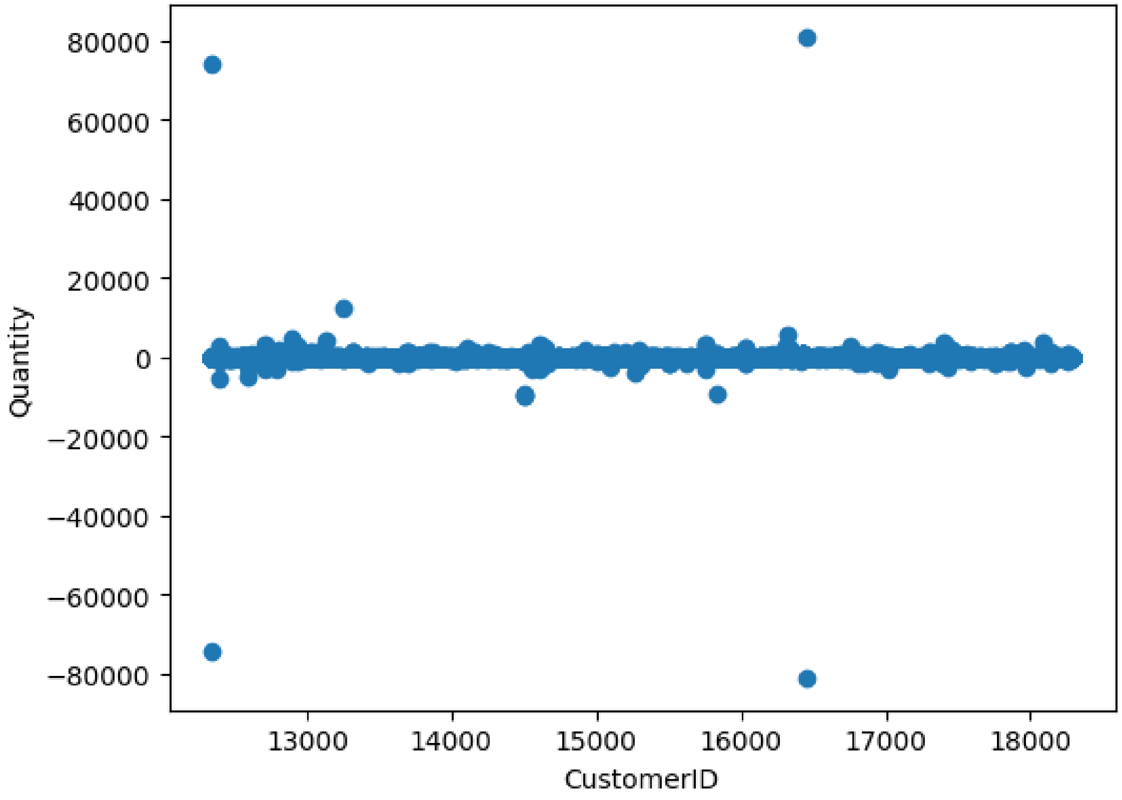
```
Quantity
 1         148227
 2          81829
 12         61063
 6          40868
 4          38484
            ...
-472            1
-161            1
-1206           1
-272            1
-80995          1
Name: count, Length: 722, dtype: int64
```

In [22]:

```
plt.scatter(df["CustomerID"],df["Quantity"])
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[22]:

```
Text(0, 0.5, 'Quantity')
```



In [23]:

```
#cols = ["InvoiceNo","StockCode","Description","Quantity","InvoiceDate","UnitPrice","Cus

#sns.pairplot(df[cols], hue="InvoiceNo")
#plt.show()
```

In [24]:

```
from sklearn.cluster import KMeans
km=KMeans()
km
```

Out[24]:

```
▾ KMeans

KMeans()
```

In [25]:

```
y_predicted=km.fit_predict(df[["CustomerID","Quantity"]])
y_predicted
```

```
C:\Users\RAMADEVI SURIPAKA\AppData\Local\Programs\Python\Python310\lib\sit
e-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default valu
e of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
it` explicitly to suppress the warning
  warnings.warn(
```

Out[25]:

```
array([3, 3, 3, ..., 2, 2, 2])
```

In [26]:

```
df["cluster"]=y_predicted
df.head()
```

Out[26]:

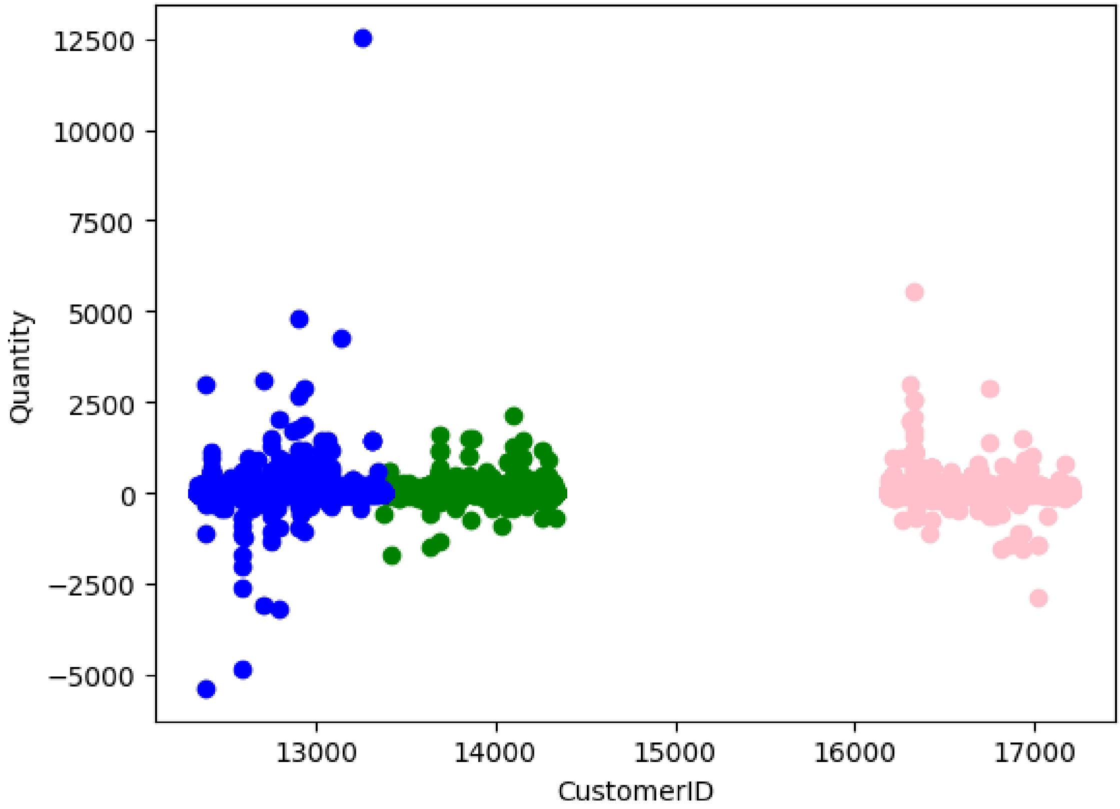
	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	01-12-2010 08:26	2.55	17850.0	Unitec Kingdom
1	536365	71053	WHITE METAL LANTERN	6	01-12-2010 08:26	3.39	17850.0	Unitec Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	01-12-2010 08:26	2.75	17850.0	Unitec Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	01-12-2010 08:26	3.39	17850.0	Unitec Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	01-12-2010 08:26	3.39	17850.0	Unitec Kingdom

In [27]:

```
df1=df[df.cluster==0]
df2=df[df.cluster==1]
df3=df[df.cluster==2]
plt.scatter(df1["CustomerID"],df1["Quantity"],color="pink")
plt.scatter(df2["CustomerID"],df2["Quantity"],color="green")
plt.scatter(df3["CustomerID"],df3["Quantity"],color="blue")
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[27]:

```
Text(0, 0.5, 'Quantity')
```



In [28]:

```
from sklearn.preprocessing import MinMaxScaler
```

In [29]:

```
km=KMeans()
km
```

Out[29]:

```
▾ KMeans
KMeans()
```

In [30]:

```
y_predicted=km.fit_predict(df[["CustomerID","Quantity"]])
y_predicted
```

```
C:\Users\RAMADEVI SURIPAKA\AppData\Local\Programs\Python\Python310\lib\sit
e-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default valu
e of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
it` explicitly to suppress the warning
  warnings.warn(
```

Out[30]:

```
array([3, 3, 3, ..., 1, 1, 1])
```

In [31]:

```
df["cluster"]=y_predicted
df.head()
```

Out[31]:

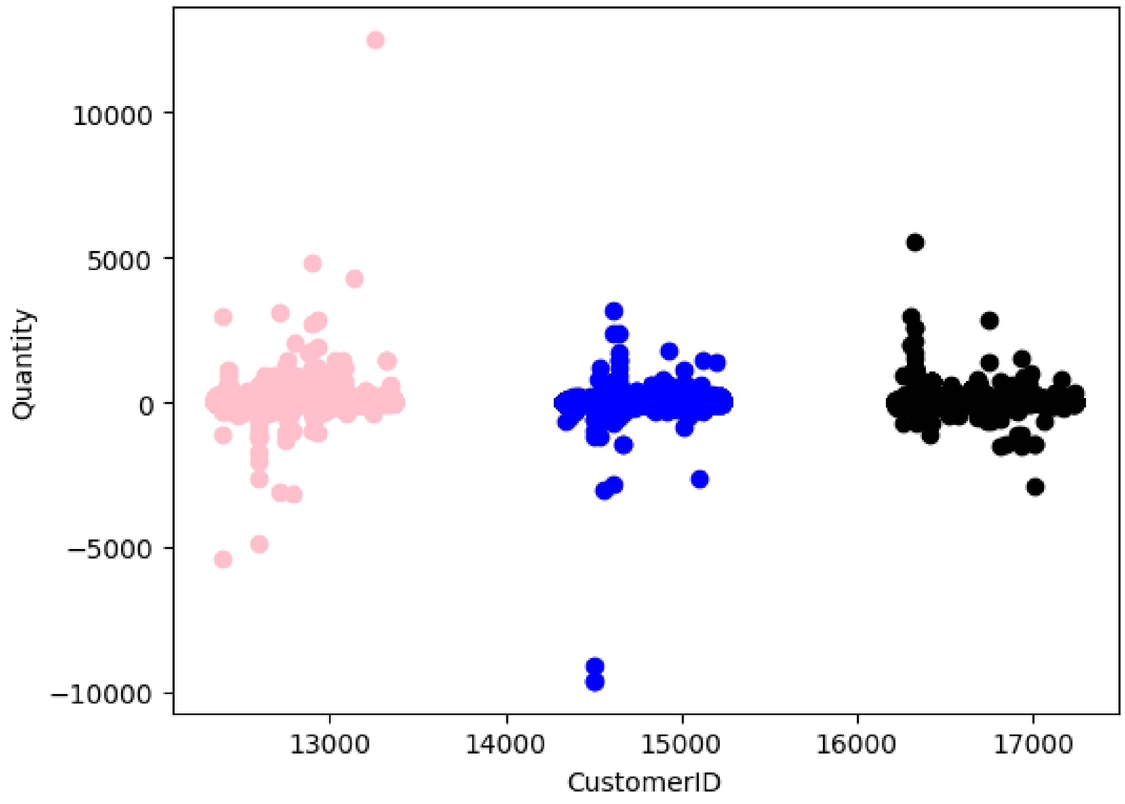
	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	01-12-2010 08:26	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	01-12-2010 08:26	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	01-12-2010 08:26	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	01-12-2010 08:26	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	01-12-2010 08:26	3.39	17850.0	United Kingdom

In [32]:

```
df1=df[df.cluster==0]
df2=df[df.cluster==1]
df3=df[df.cluster==2]
plt.scatter(df1["CustomerID"],df1["Quantity"],color="black")
plt.scatter(df2["CustomerID"],df2["Quantity"],color="pink")
plt.scatter(df3["CustomerID"],df3["Quantity"],color="blue")
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[32]:

```
Text(0, 0.5, 'Quantity')
```



In [33]:

```
from sklearn.preprocessing import MinMaxScaler
```

In [34]:

```
Scaler=MinMaxScaler()
```

In [35]:

```
Scaler.fit(df[["CustomerID"]])
df["CustomerID"]=Scaler.transform(df[["CustomerID"]])
df.head()
```

Out[35]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	01-12-2010 08:26	2.55	0.926443	Unitec Kingdom
1	536365	71053	WHITE METAL LANTERN	6	01-12-2010 08:26	3.39	0.926443	Unitec Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	01-12-2010 08:26	2.75	0.926443	Unitec Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	01-12-2010 08:26	3.39	0.926443	Unitec Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	01-12-2010 08:26	3.39	0.926443	Unitec Kingdom

In [36]:

```
Scaler.fit(df[["Quantity"]])
df["Quantity"]=Scaler.transform(df[["Quantity"]])
df.head()
```

Out[36]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Countr
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	0.500037	01-12-2010 08:26	2.55	0.926443	Unitec Kingdon
1	536365	71053	WHITE METAL LANTERN	0.500037	01-12-2010 08:26	3.39	0.926443	Unitec Kingdon
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	0.500049	01-12-2010 08:26	2.75	0.926443	Unitec Kingdon
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	0.500037	01-12-2010 08:26	3.39	0.926443	Unitec Kingdon
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	0.500037	01-12-2010 08:26	3.39	0.926443	Unitec Kingdon

In [37]:

```
km=KMeans()
km
```

Out[37]:

```
▾ KMeans

KMeans()
```

In [38]:

```
y_predicted=km.fit_predict(df[["CustomerID","Quantity"]])
y_predicted
```

```
C:\Users\RAMADEVI SURIPAKA\AppData\Local\Programs\Python\Python310\lib\sit
e-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default valu
e of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
it` explicitly to suppress the warning
  warnings.warn(
```

Out[38]:

```
array([4, 4, 4, ..., 7, 7, 7])
```

In [39]:

```
df["New cluster"]=y_predicted
df.head()
```

Out[39]:

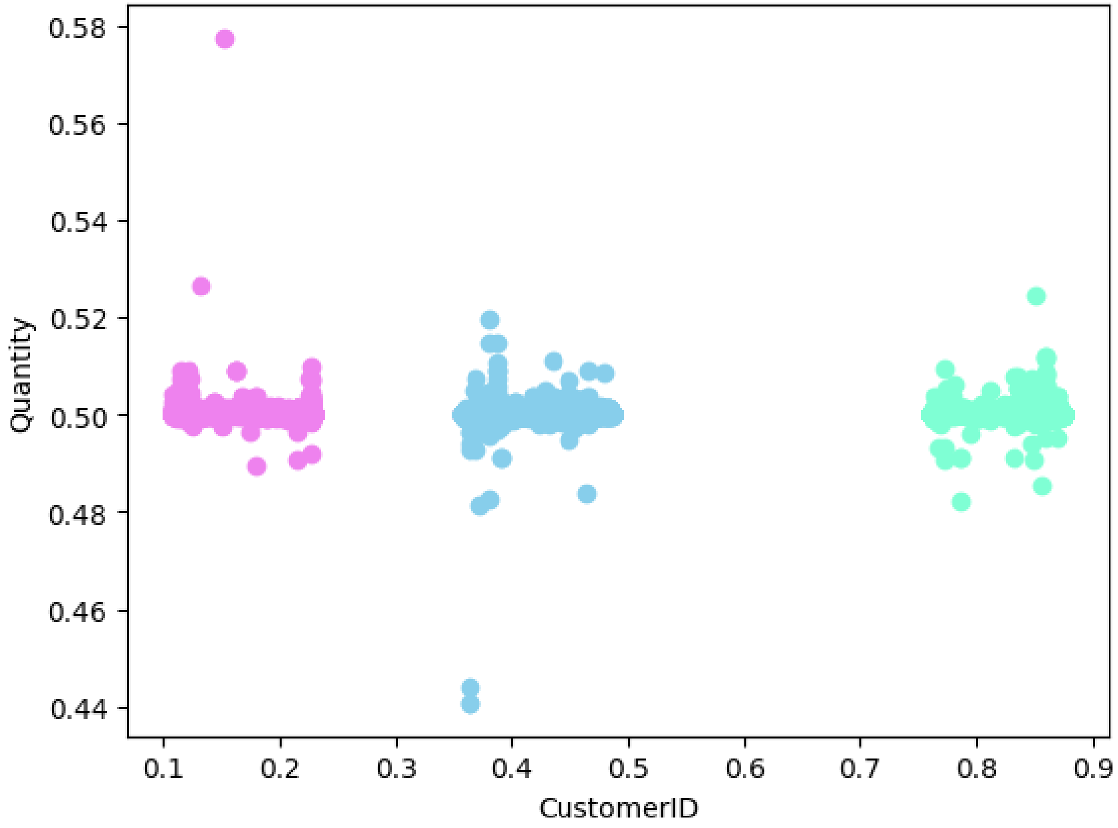
	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	0.500037	01-12-2010 08:26	2.55	0.926443	Unitec Kingdon
1	536365	71053	WHITE METAL LANTERN	0.500037	01-12-2010 08:26	3.39	0.926443	Unitec Kingdon
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	0.500049	01-12-2010 08:26	2.75	0.926443	Unitec Kingdon
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	0.500037	01-12-2010 08:26	3.39	0.926443	Unitec Kingdon
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	0.500037	01-12-2010 08:26	3.39	0.926443	Unitec Kingdon

In [40]:

```
df1=df[df["New cluster"]==0]
df2=df[df["New cluster"]==1]
df3=df[df["New cluster"]==2]
plt.scatter(df1["CustomerID"],df1["Quantity"],color="aquamarine")
plt.scatter(df2["CustomerID"],df2["Quantity"],color="violet")
plt.scatter(df3["CustomerID"],df3["Quantity"],color="skyblue")
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[40]:

```
Text(0, 0.5, 'Quantity')
```



In [41]:

```
km.cluster_centers_
```

Out[41]:

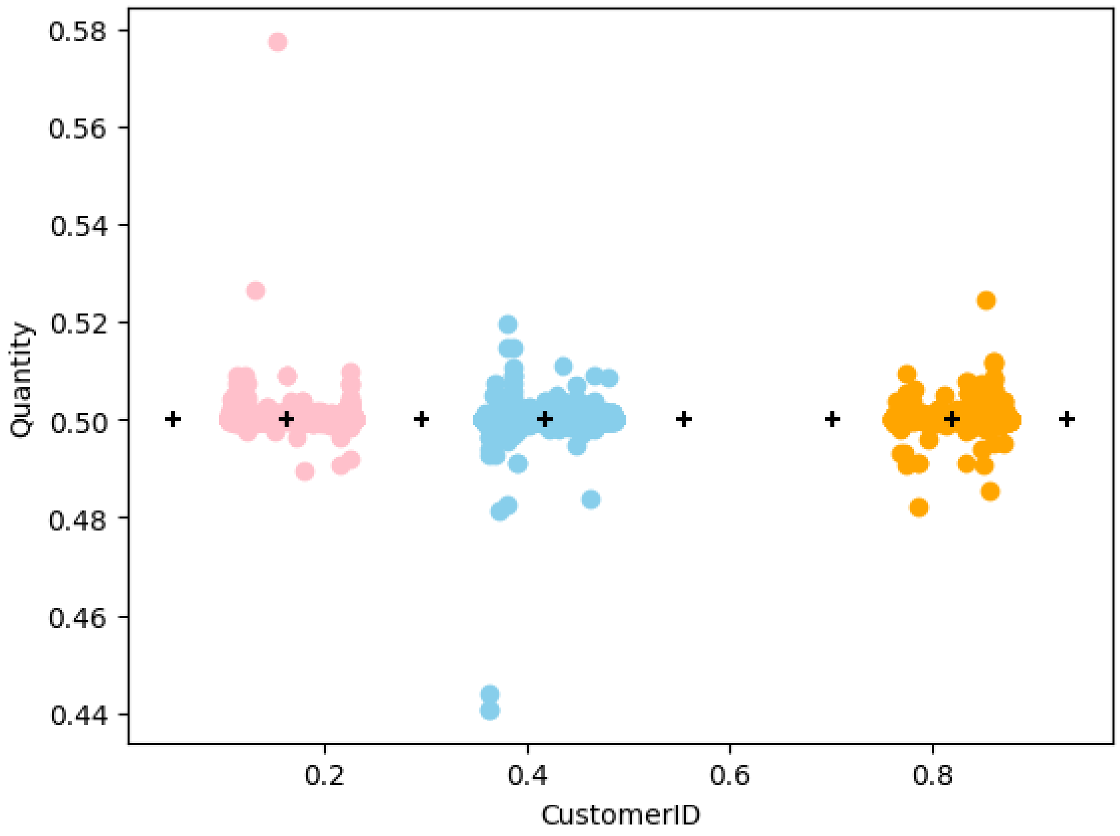
```
array([[0.81846964, 0.50006032],
       [0.16326876, 0.50006159],
       [0.41826663, 0.50006089],
       [0.55502897, 0.50005357],
       [0.93301823, 0.50005097],
       [0.2960104 , 0.50006014],
       [0.70121934, 0.50005792],
       [0.05128805, 0.50006687]])
```

In [42]:

```
df1=df[df["New cluster"]==0]
df2=df[df["New cluster"]==1]
df3=df[df["New cluster"]==2]
plt.scatter(df1["CustomerID"],df1["Quantity"],color="orange")
plt.scatter(df2["CustomerID"],df2["Quantity"],color="pink")
plt.scatter(df3["CustomerID"],df3["Quantity"],color="skyblue")
plt.scatter(km.cluster_centers_[:,0],km.cluster_centers_[:,1],color="black",marker="+")
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[42]:

```
Text(0, 0.5, 'Quantity')
```



In [44]:

```
k_rng=range(1,10)
sse=[]
for k in k_rng:
    km=KMeans(n_clusters=k)
    km.fit(df[["CustomerID","Quantity"]])
    sse.append(km.inertia_)
    sse
```

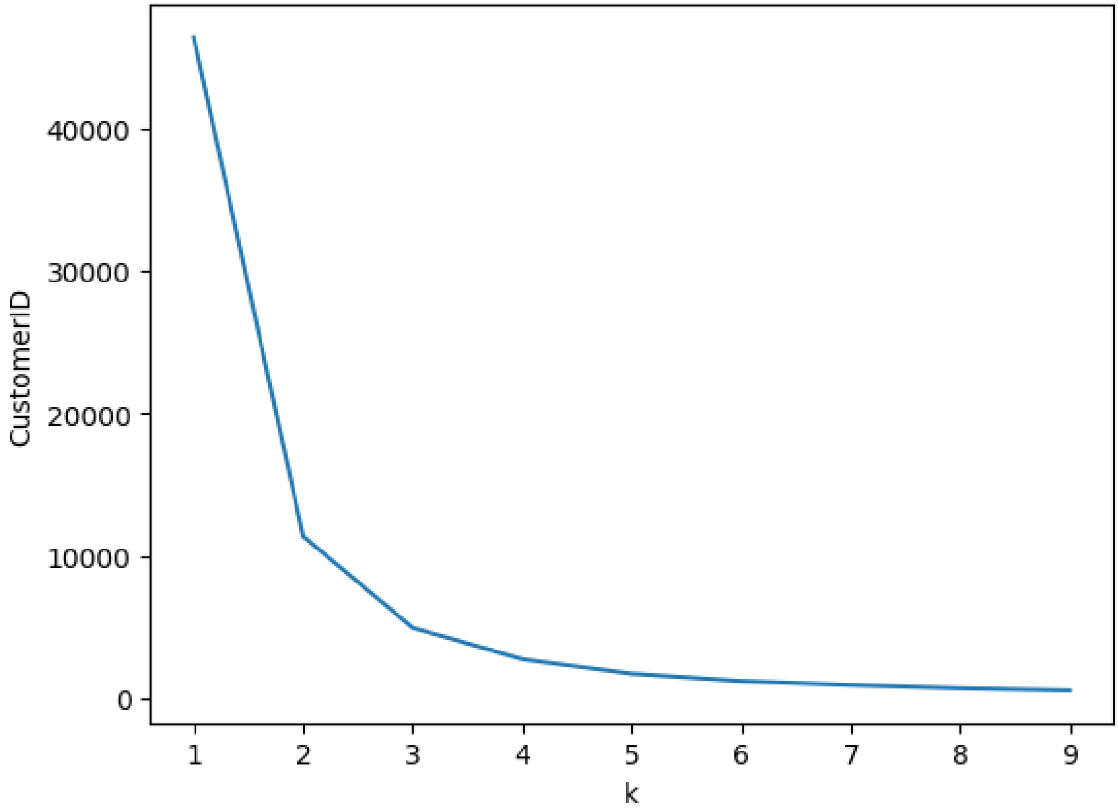
```
C:\Users\RAMADEVI SURIPAKA\AppData\Local\Programs\Python\Python310\lib\sit
e-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default valu
e of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
it` explicitly to suppress the warning
  warnings.warn(
C:\Users\RAMADEVI SURIPAKA\AppData\Local\Programs\Python\Python310\lib\sit
e-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default valu
e of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
it` explicitly to suppress the warning
  warnings.warn(
C:\Users\RAMADEVI SURIPAKA\AppData\Local\Programs\Python\Python310\lib\sit
e-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default valu
e of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
it` explicitly to suppress the warning
  warnings.warn(
C:\Users\RAMADEVI SURIPAKA\AppData\Local\Programs\Python\Python310\lib\sit
e-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default valu
e of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
it` explicitly to suppress the warning
  warnings.warn(
C:\Users\RAMADEVI SURIPAKA\AppData\Local\Programs\Python\Python310\lib\sit
e-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default valu
e of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
it` explicitly to suppress the warning
  warnings.warn(
C:\Users\RAMADEVI SURIPAKA\AppData\Local\Programs\Python\Python310\lib\sit
e-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default valu
e of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
it` explicitly to suppress the warning
  warnings.warn(
C:\Users\RAMADEVI SURIPAKA\AppData\Local\Programs\Python\Python310\lib\sit
e-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default valu
e of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
it` explicitly to suppress the warning
  warnings.warn(
C:\Users\RAMADEVI SURIPAKA\AppData\Local\Programs\Python\Python310\lib\sit
e-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default valu
e of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
it` explicitly to suppress the warning
  warnings.warn(
C:\Users\RAMADEVI SURIPAKA\AppData\Local\Programs\Python\Python310\lib\sit
e-packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default valu
e of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
it` explicitly to suppress the warning
  warnings.warn(
```

In [45]:

```
plt.plot(k_rng,sse)
plt.xlabel("k")
plt.ylabel("CustomerID")
```

Out[45]:

```
Text(0, 0.5, 'CustomerID')
```



Conclusion:

For the given dataset we use K-means Clustering and done the grouping based on the given data.In theabove dataset we will take customer id and quantity based on that we make the clusters. When the K-value islow error rate is more and the K-value is high error rate is very high. So,finally we can Conclude the abovedataset is bestfit for K-Means.A fundamental step for any unsupervised algorithm is to determine the optimal number of clusters into which the data may be clustered. The Elbow Method is one of the most popular methods to determine this optimal value of k.

In [ ]:

```

```

In [ ]: