

Project - 5 (DATASET: Online Retail) The transactions made by a UK-based, registered, non-store online retailer between December 1, 2010, and December 9, 2011, are all included in the transnational data set known as online retail. The company primarily offers one-of-a-kind gifts for every occasion. The company has a large number of wholesalers as clients. **Company Objective** Using the global online retail dataset, we will design a clustering model and select the ideal group of clients for the business to target.

```
In [2]: ▶ import pandas as pd
        from matplotlib import pyplot as plt
        %matplotlib inline
```

```
In [3]: df=pd.read_csv(r"C:\Users\chinta pavani\Documents\OnlineRetail1.csv")
df
```

Out[3]:

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

541909 rows × 8 columns

```
In [4]: df.head()
```

```
Out[4]:
```

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

```
In [5]: df.tail()
```

```
Out[5]:
```

| | 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 [6]: df.shape
```

```
Out[6]: (541909, 8)
```

In [7]: `df.describe`

```
Out[7]: <bound method NDFrame.describe 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 [8]: `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 [9]: df.isnull().sum()
```

```
Out[9]: InvoiceNo      0
        StockCode     0
        Description  1454
        Quantity     0
        InvoiceDate    0
        UnitPrice     0
        CustomerID   135080
        Country       0
        dtype: int64
```

```
In [14]: df.fillna(method='ffill',inplace=True)
```

```
In [15]: df.isnull().sum()
```

```
Out[15]: InvoiceNo      0
        StockCode     0
        Description    0
        Quantity      0
        InvoiceDate    0
        UnitPrice     0
        CustomerID    0
        Country       0
        dtype: int64
```

```
In [11]: df['InvoiceNo'].value_counts()
```

```
Out[11]: 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 [10]: df['CustomerID'].value_counts()
```

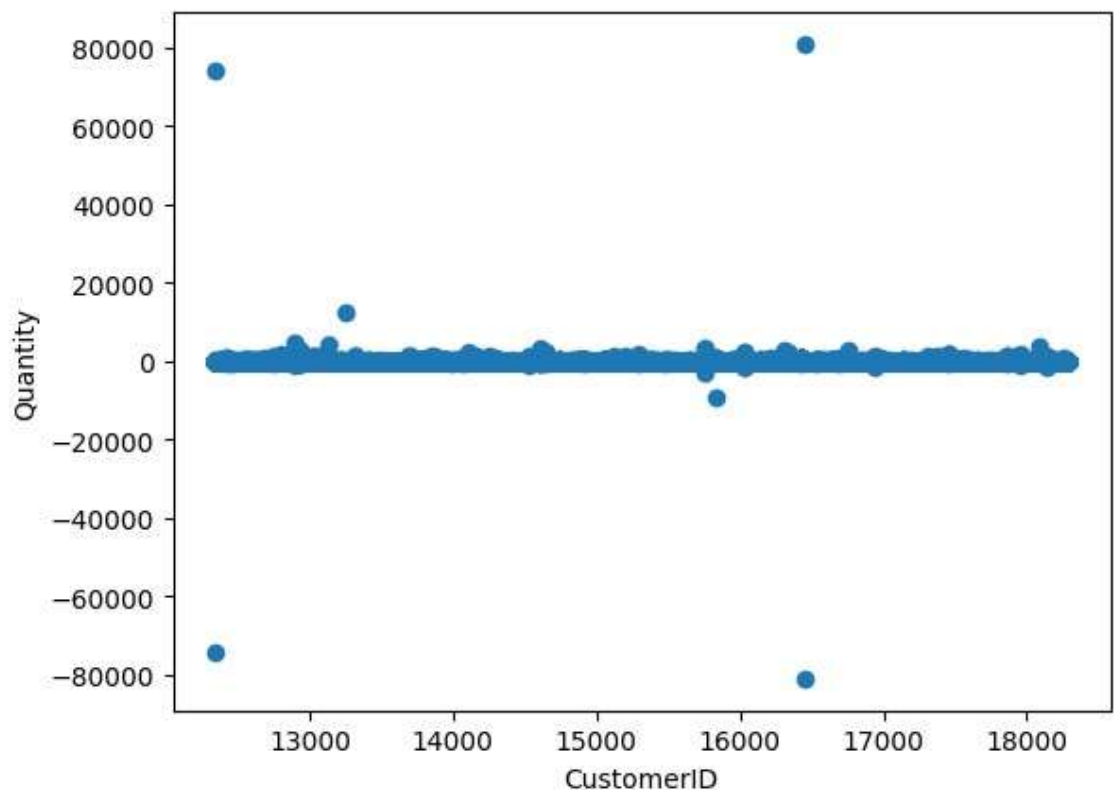
```
Out[10]: CustomerID
17841.0    7983
14911.0    5903
14096.0    5128
12748.0    4642
14606.0    2782
...
15070.0     1
15753.0     1
17065.0     1
16881.0     1
16995.0     1
Name: count, Length: 4372, dtype: int64
```

```
In [12]: df['Quantity'].value_counts()
```

```
Out[12]: 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 [13]: ▶ plt.scatter(df["CustomerID"],df["Quantity"])
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[13]: Text(0, 0.5, 'Quantity')



```
In [16]: ▶ from sklearn.cluster import KMeans
km=KMeans()
km
```

Out[16]:

| |
|----------|
| ▼ KMeans |
| KMeans() |

```
In [17]: ▶ y_predicted=km.fit_predict(df[["CustomerID","Quantity"]])
y_predicted
```

C:\Users\chinta pavani\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning
warnings.warn(

Out[17]: array([1, 1, 1, ..., 6, 6, 6])

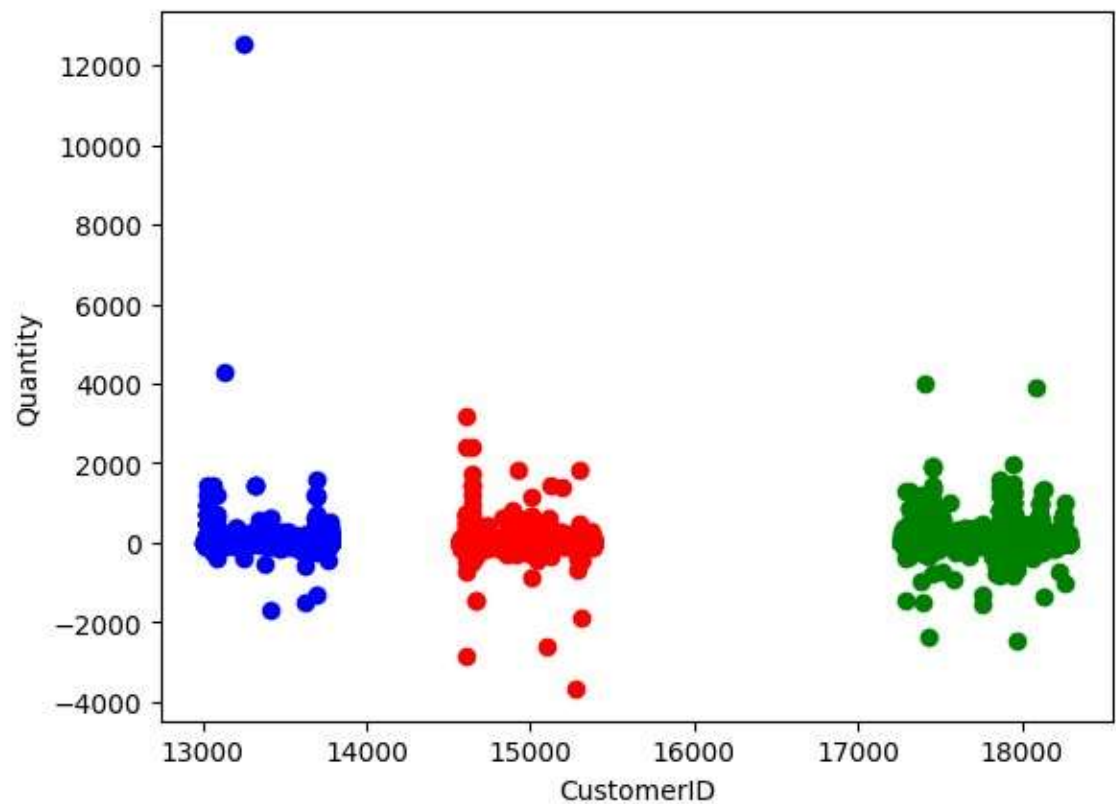
```
In [18]: df["cluster"]=y_predicted  
df.head()
```

Out[18]:

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


```
In [19]: ▶ df1=df[df.cluster==0]
df2=df[df.cluster==1]
df3=df[df.cluster==2]
plt.scatter(df1["CustomerID"],df1["Quantity"],color="red")
plt.scatter(df2["CustomerID"],df2["Quantity"],color="green")
plt.scatter(df3["CustomerID"],df3["Quantity"],color="blue")
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[19]: Text(0, 0.5, 'Quantity')



```
In [20]: from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler()
scaler.fit(df[["Quantity"]])
df["Quantity"]=scaler.transform(df[["Quantity"]])
df.head()
```

Out[20]:

| | 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 | 17850.0 | United Kingdom |
| 1 | 536365 | 71053 | WHITE METAL LANTERN | 0.500037 | 01-12-2010 08:26 | 3.39 | 17850.0 | United Kingdom |
| 2 | 536365 | 84406B | CREAM CUPID HEARTS COAT HANGER | 0.500049 | 01-12-2010 08:26 | 2.75 | 17850.0 | United Kingdom |
| 3 | 536365 | 84029G | KNITTED UNION FLAG HOT WATER BOTTLE | 0.500037 | 01-12-2010 08:26 | 3.39 | 17850.0 | United Kingdom |
| 4 | 536365 | 84029E | RED WOOLLY HOTTIE WHITE HEART. | 0.500037 | 01-12-2010 08:26 | 3.39 | 17850.0 | United Kingdom |

```
In [21]: > scaler.fit(df[["CustomerID"]])
df["CustomerID"]=scaler.transform(df[["CustomerID"]])
df.head()
```

Out[21]:

| | 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 | United Kingdom |
| 1 | 536365 | 71053 | WHITE METAL LANTERN | 0.500037 | 01-12-2010 08:26 | 3.39 | 0.926443 | United Kingdom |
| 2 | 536365 | 84406B | CREAM CUPID HEARTS COAT HANGER | 0.500049 | 01-12-2010 08:26 | 2.75 | 0.926443 | United Kingdom |
| 3 | 536365 | 84029G | KNITTED UNION FLAG HOT WATER BOTTLE | 0.500037 | 01-12-2010 08:26 | 3.39 | 0.926443 | United Kingdom |
| 4 | 536365 | 84029E | RED WOOLLY HOTTIE WHITE HEART. | 0.500037 | 01-12-2010 08:26 | 3.39 | 0.926443 | United Kingdom |

K-Means Clustering

```
In [22]: > km=KMeans()
```

```
In [23]: > y_predicted=km.fit_predict(df[["CustomerID","Quantity"]])
y_predicted
```

C:\Users\chinta pavani\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning
warnings.warn(

Out[23]: array([3, 3, 3, ..., 6, 6, 6])

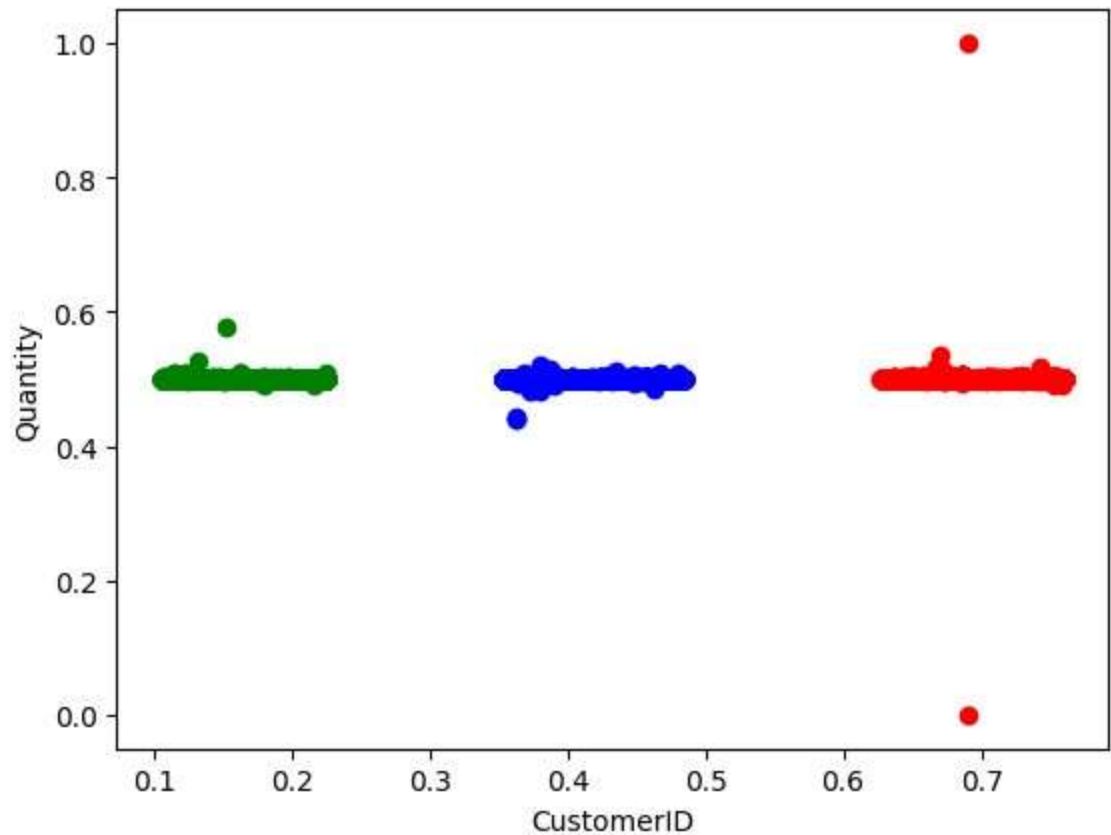
```
In [24]: df["New Cluster"]=y_predicted  
df.head()
```

Out[24]:

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

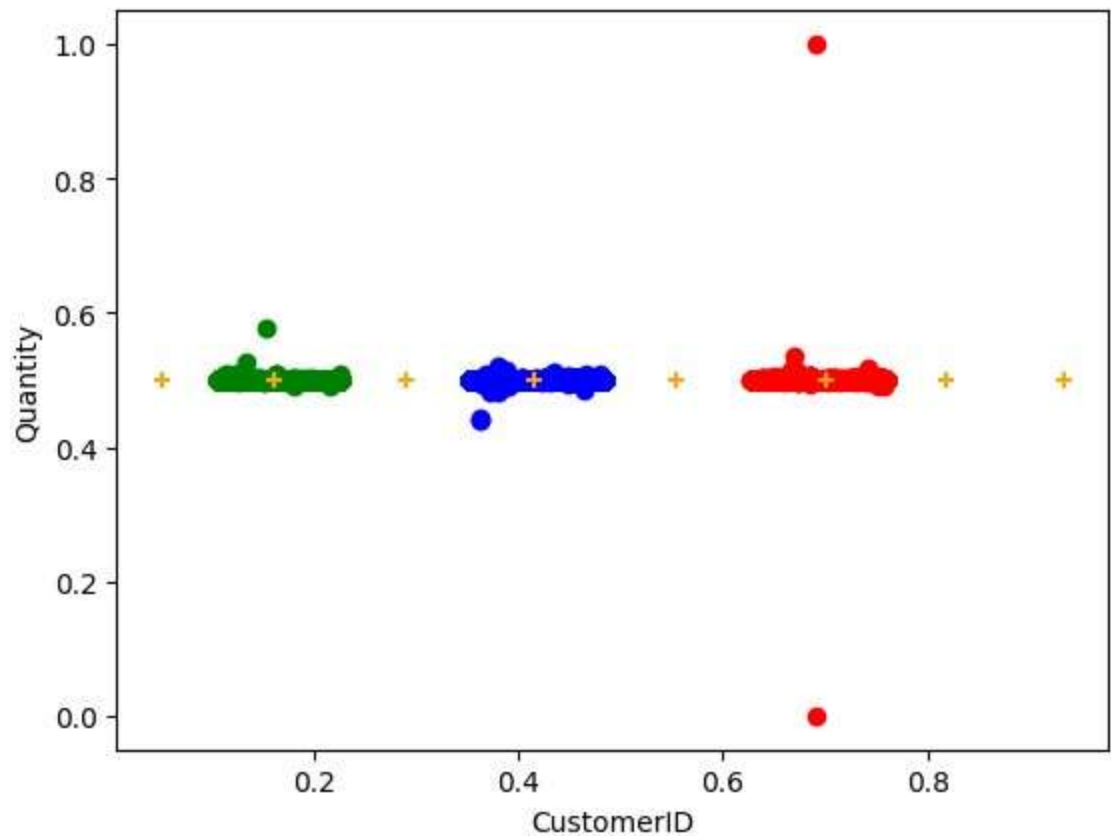
```
In [25]: ▶ df1=df[df["New Cluster"]==0]
df2=df[df["New Cluster"]==1]
df3=df[df["New Cluster"]==2]
plt.scatter(df1["CustomerID"],df1["Quantity"],color="red")
plt.scatter(df2["CustomerID"],df2["Quantity"],color="green")
plt.scatter(df3["CustomerID"],df3["Quantity"],color="blue")
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[25]: Text(0, 0.5, 'Quantity')



```
In [26]: ▶ 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="red")
plt.scatter(df2["CustomerID"],df2["Quantity"],color="green")
plt.scatter(df3["CustomerID"],df3["Quantity"],color="blue")
plt.scatter(km.cluster_centers_[0],km.cluster_centers_[1],color="orange")
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[26]: Text(0, 0.5, 'Quantity')



```
In [27]: ▶ k_rng=range(1,10)
sse=[]
```

```
In [28]: ► for k in k_rng:
            km=KMeans(n_clusters=k)
            km.fit(df[["CustomerID","Quantity"]])
            sse.append(km.inertia_)
            #km.inertia_ will give you the value of sum of square error
            print(sse)
            plt.plot(k_rng,sse)
            plt.xlabel("K")
            plt.ylabel("Sum of Squared Error")
```

C:\Users\chinta pavani\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning

warnings.warn(

C:\Users\chinta pavani\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning

warnings.warn(

C:\Users\chinta pavani\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning

warnings.warn(

C:\Users\chinta pavani\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning

warnings.warn(

C:\Users\chinta pavani\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning

warnings.warn(

C:\Users\chinta pavani\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning

warnings.warn(

C:\Users\chinta pavani\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning

warnings.warn(

C:\Users\chinta pavani\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning

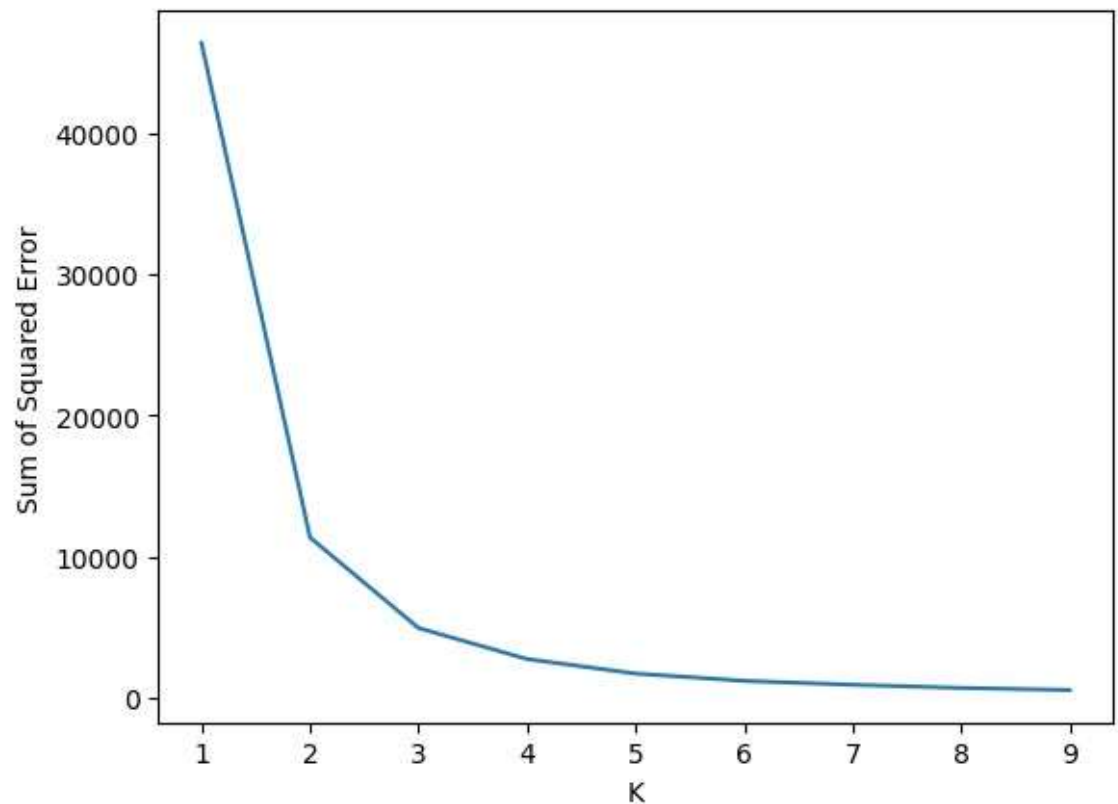
warnings.warn(

C:\Users\chinta pavani\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning

warnings.warn(

```
[46374.84553398485, 11336.065820168866, 4921.06838404593, 2723.5053916362754, 1695.0392229312758, 1178.6080833990227, 902.5787504264805, 677.3046506461086, 528.8251995247878]
```

```
Out[28]: Text(0, 0.5, 'Sum of Squared Error')
```



CONCLUSION

For the given dataset we use K-means Clustering and done the grouping based on the given data. In the above dataset we will take customer id and quantity based on that we make the clusters. When the K-value is above dataset we will take customer id and quantity based on that we make the clusters dataset is bestfit for K-Means.

```
In [ ]: ▶
```