In [1]:
```
import pandas as pd
from matplotlib import pyplot as plt
%matplotlib inline
```

In [3]:
```
df=pd.read_csv(r"C:\Users\sneha\OneDrive\Desktop\OnlineRetail.csv")
df
```

Out[3]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Cou
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	01-12-2010 08:26	2.55	17850.0	Ur Kingc
1	536365	71053	WHITE METAL LANTERN	6	01-12-2010 08:26	3.39	17850.0	Ur Kingc
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	01-12-2010 08:26	2.75	17850.0	Ur Kingc
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	01-12-2010 08:26	3.39	17850.0	Ur Kingc
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	01-12-2010 08:26	3.39	17850.0	Ur Kingc
...	...	...	...	...	...	...	...	
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	09-12-2011 12:50	0.85	12680.0	Fra
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	09-12-2011 12:50	2.10	12680.0	Fra
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	09-12-2011 12:50	4.15	12680.0	Fra
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	09-12-2011 12:50	4.15	12680.0	Fra
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	09-12-2011 12:50	4.95	12680.0	Fra

541909 rows × 8 columns

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

Out[5]:

	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 [6]:
```
df.tail()
```

Out[6]:

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

In [7]:
```
df['InvoiceNo'].value_counts()
```

Out[7]:
```
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 [8]:
```
df['CustomerID'].value_counts()
```

Out[8]:
```
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 [9]:
```
df['Quantity'].value_counts()
```

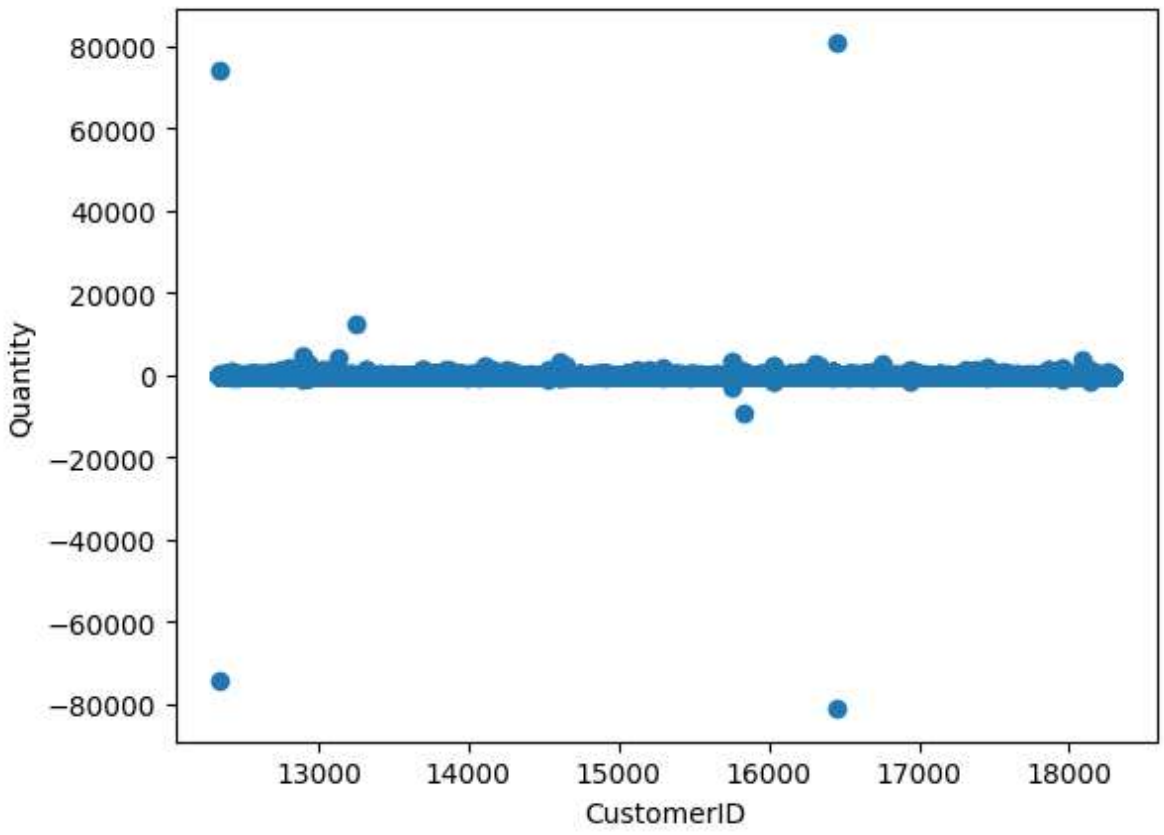
Out[9]:
```
Quantity
 1        148227
 2         81829
 12        61063
 6         40868
 4         38484
           ...
-472           1
-161           1
-1206          1
-272           1
-80995         1
Name: count, Length: 722, dtype: int64
```

In [11]:
```
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 [12]:
```
plt.scatter(df["CustomerID"],df["Quantity"])
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[12]:
```
Text(0, 0.5, 'Quantity')
```



In [13]:
```
df.isnull().sum()
```

Out[13]:
```
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 [16]:
```
from sklearn.cluster import KMeans
km=KMeans()
km
```

Out[16]:
```
KMeans()
```

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

```
C:\Users\sneha\AppData\Local\Programs\Python\Python310\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([3, 3, 3, ..., 1, 1, 1])
```

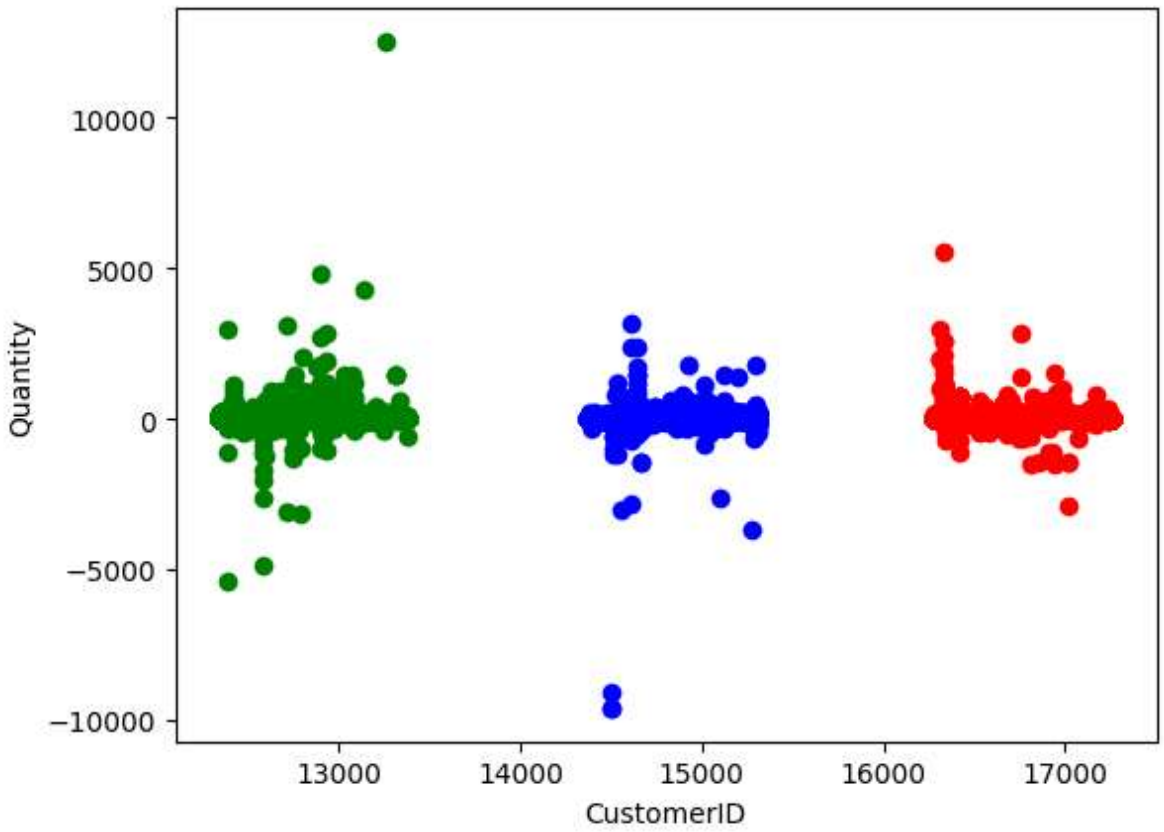
In [18]:
```
df["cluster"]=y_predicted
df.head()
```

Out[18]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	cl
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 [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	c
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	c
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

```
km=KMeans()
```

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

```
C:\Users\sneha\AppData\Local\Programs\Python\Python310\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([6, 6, 6, ..., 1, 1, 1])
```

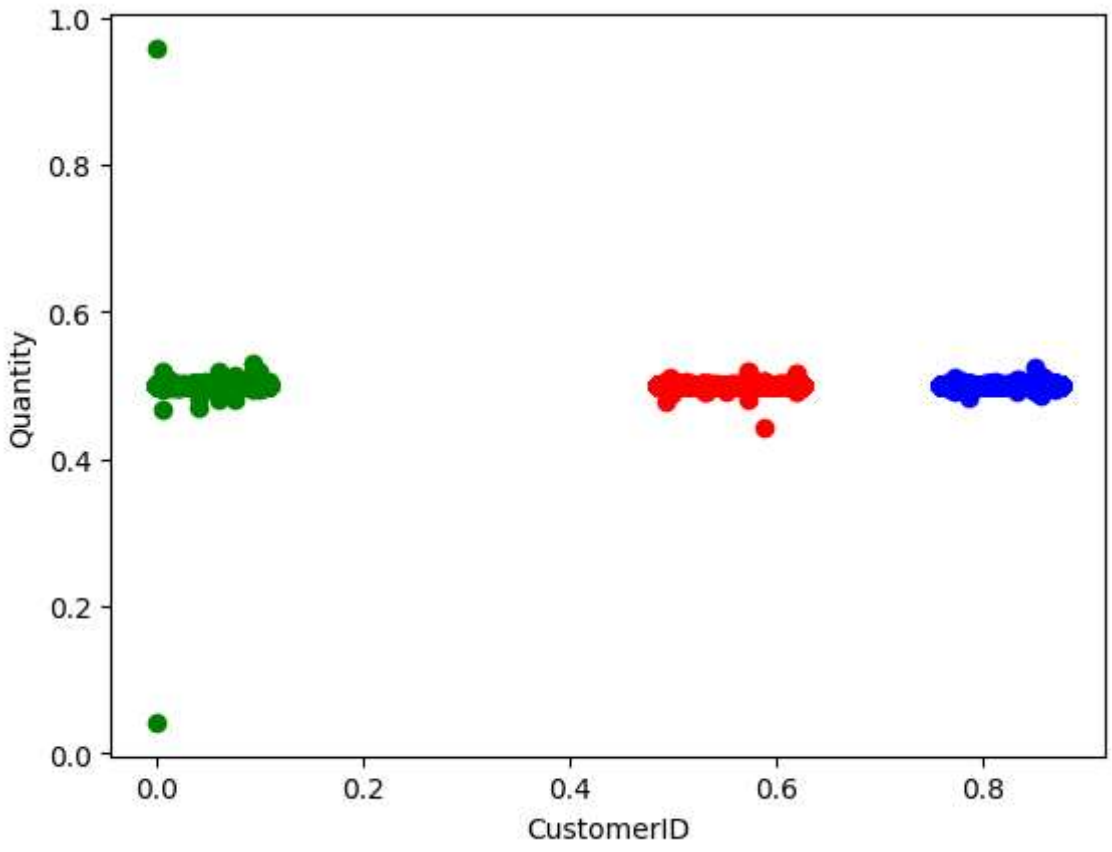
In [24]:
```
df["New Cluster"]=y_predicted
df.head()
```

Out[24]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	c
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	

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')
```

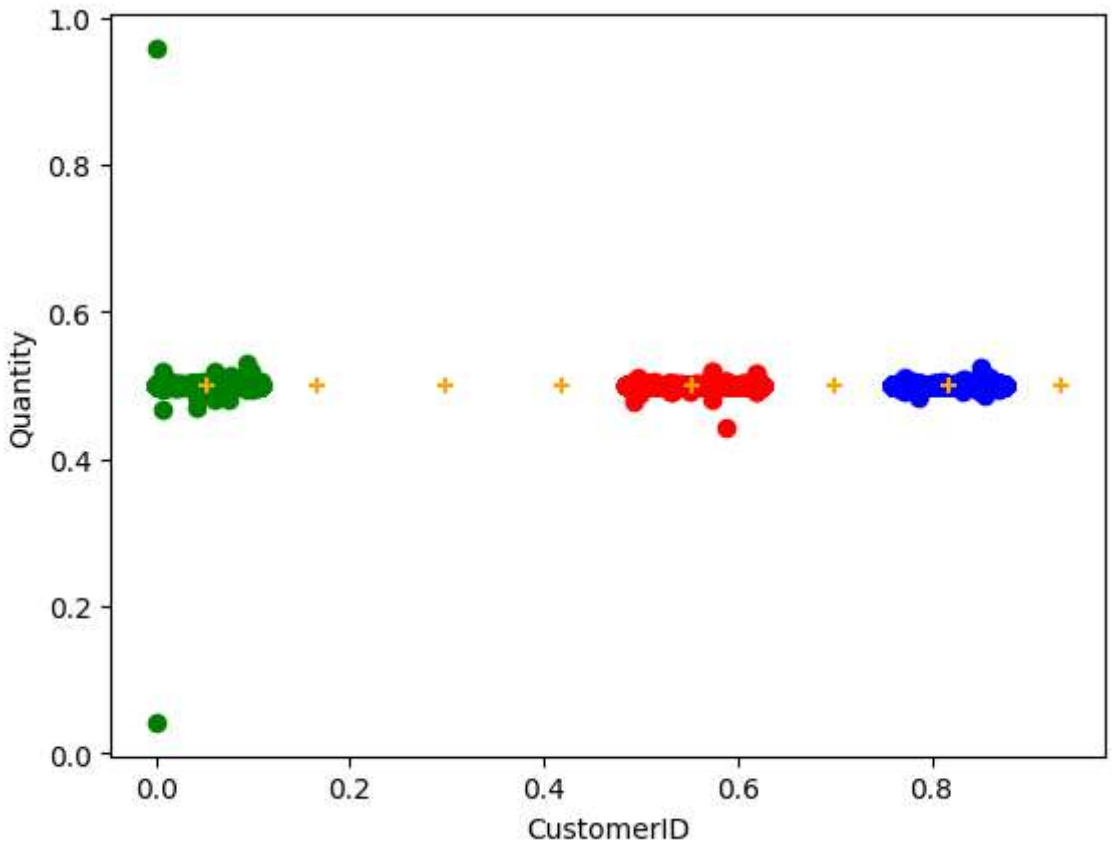


In [26]:
```
km.cluster_centers_
```

Out[26]:
```
array([[0.55322798, 0.50005383],
       [0.0515406 , 0.50006705],
       [0.81756576, 0.50005988],
       [0.2980318 , 0.50006082],
       [0.41758384, 0.50006084],
       [0.69962461, 0.50005827],
       [0.9328779 , 0.50005088],
       [0.16558693, 0.50006063]])
```

In [27]:
```
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",m
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[27]:
```
Text(0, 0.5, 'Quantity')
```



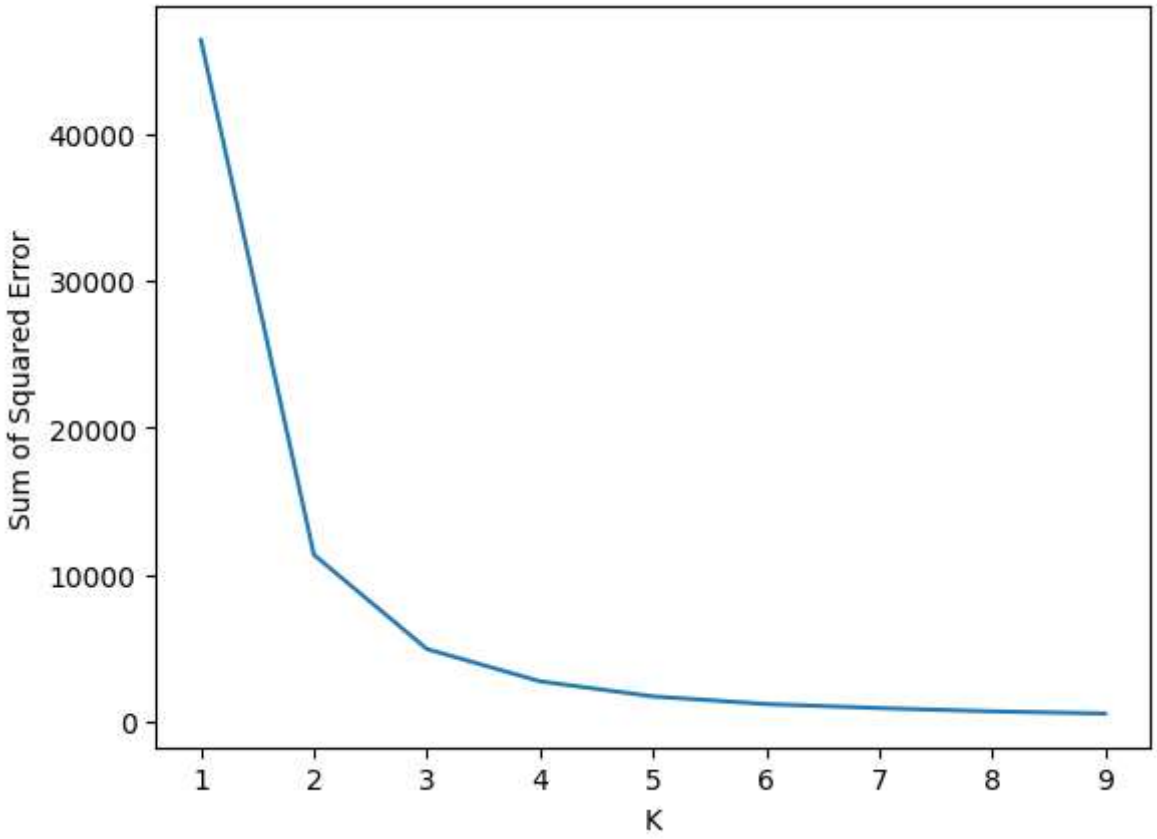
In [28]:
```
k_rng=range(1,10)
sse=[]
```

In [29]:
```
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\sneha\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\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 supp
ress the warning
  warnings.warn(
C:\Users\sneha\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\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 supp
ress the warning
  warnings.warn(
C:\Users\sneha\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\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 supp
ress the warning
  warnings.warn(
C:\Users\sneha\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\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 supp
ress the warning
  warnings.warn(
C:\Users\sneha\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\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 supp
ress the warning
  warnings.warn(
C:\Users\sneha\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\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 supp
ress the warning
  warnings.warn(
C:\Users\sneha\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\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 supp
ress the warning
  warnings.warn(
C:\Users\sneha\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\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 supp
ress the warning
  warnings.warn(
C:\Users\sneha\AppData\Local\Programs\Python\Python310\lib\site-packages\skle
arn\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 supp
ress the warning
  warnings.warn(
```

```
[46374.84553398474, 11336.065820169122, 4921.025469201893, 2723.506695363736
5, 1695.0405858458662, 1178.5646874650968, 903.0609330413083, 677.42937964787
14, 532.2335090345556]
```

Out[29]: Text(0, 0.5, 'Sum of Squared Error')



Conclusion:

For the given dataset we use K Means clustering and done the grou[ing based on the given data. In the dataset based on the customer Id and quantity we made clusters.So, we can conclude that the given dataset is best fit for K Means.

In [ ]: