online retail

problem statement

```
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
ObjectiveUsing the global online retail dataset, we will design a clustering
model and
select the ideal group of clients for the business to target.
```

In [3]:
```
import pandas as pd
from matplotlib import pyplot as plt
%matplotlib inline
```

DATA COLLECTION

In [4]:
```
s=pd.read_csv(r"C:\Users\mouni\Downloads\OnlineRetail1.csv")
s
```

Out[4]:

	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 King
1	536365	71053	WHITE METAL LANTERN	6	01-12-2010 08:26	3.39	17850.0	Ur King
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	01-12-2010 08:26	2.75	17850.0	Ur King
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	01-12-2010 08:26	3.39	17850.0	Ur King
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	01-12-2010 08:26	3.39	17850.0	Ur King
...	...	...	...	...	...	...	...	
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

data cleaning

In [5]:
```
s.head(100)
```

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
...	...	...	...	...	...	...	...	...
95	536378	22352	LUNCH BOX WITH CUTLERY RETROSPOT	6	01-12-2010 09:37	2.55	14688.0	United Kingdom
96	536378	21212	PACK OF 72 RETROSPOT CAKE CASES	120	01-12-2010 09:37	0.42	14688.0	United Kingdom
97	536378	21975	PACK OF 60 DINOSAUR CAKE CASES	24	01-12-2010 09:37	0.55	14688.0	United Kingdom
98	536378	21977	PACK OF 60 PINK PAISLEY CAKE CASES	24	01-12-2010 09:37	0.55	14688.0	United Kingdom
99	536378	84991	60 TEATIME FAIRY CAKE CASES	24	01-12-2010 09:37	0.55	14688.0	United Kingdom

100 rows × 8 columns

In [6]:
```
s.tail()
```

Out[6]:

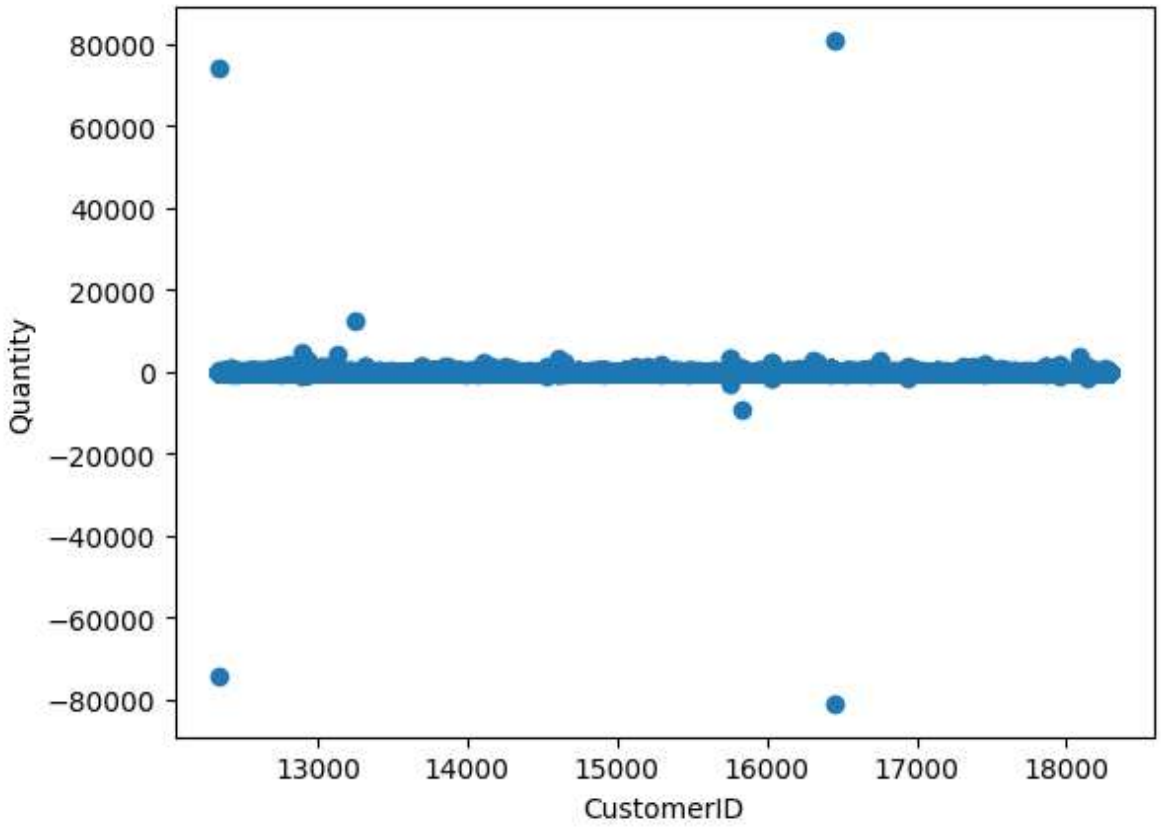
	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Cou
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	09-12-2011 12:50	0.85	12680.0	Fra
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	09-12-2011 12:50	2.10	12680.0	Fra
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	09-12-2011 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]:
```
s['Description'].value_counts()
```

Out[7]:
```
Description
WHITE HANGING HEART T-LIGHT HOLDER     2369
REGENCY CAKESTAND 3 TIER               2200
JUMBO BAG RED RETROSPOT                2159
PARTY BUNTING                          1727
LUNCH BAG RED RETROSPOT                1638
                                       ...
amazon sales                              1
sold as set/6 by dotcom                   1
damages/dotcom?                           1
historic computer difference?....se       1
Show Samples                              1
Name: count, Length: 4219, dtype: int64
```

In [8]:
```
plt.scatter(s["CustomerID"],s["Quantity"])
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[8]:
```
Text(0, 0.5, 'Quantity')
```



In [9]:
```
s.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 [10]:
```
s.isnull().sum()
```

Out[10]:
```
InvoiceNo           0
StockCode           0
Description      1454
Quantity            0
InvoiceDate         0
UnitPrice           0
CustomerID     135080
Country             0
dtype: int64
```

In [11]:
```
s.fillna(method='ffill',inplace=True)
```

In [12]:
```
s.isnull().sum()
```

Out[12]:
```
InvoiceNo      0
StockCode      0
Description    0
Quantity       0
InvoiceDate    0
UnitPrice      0
CustomerID     0
Country        0
dtype: int64
```

In [13]:
```
from sklearn.cluster import KMeans
```

In [14]:
```
km=KMeans()
km
```

Out[14]:
```
KMeans()
```

In [15]:
```
y_pred=km.fit_predict(s[["CustomerID","Quantity"]])
y_pred
```

```
C:\Users\mouni\AppData\Local\Programs\Python\Python310\lib\site-packages\skl
earn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` wi
ll change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to
suppress the warning
  warnings.warn(
```

Out[15]:
```
array([2, 2, 2, ..., 3, 3, 3])
```

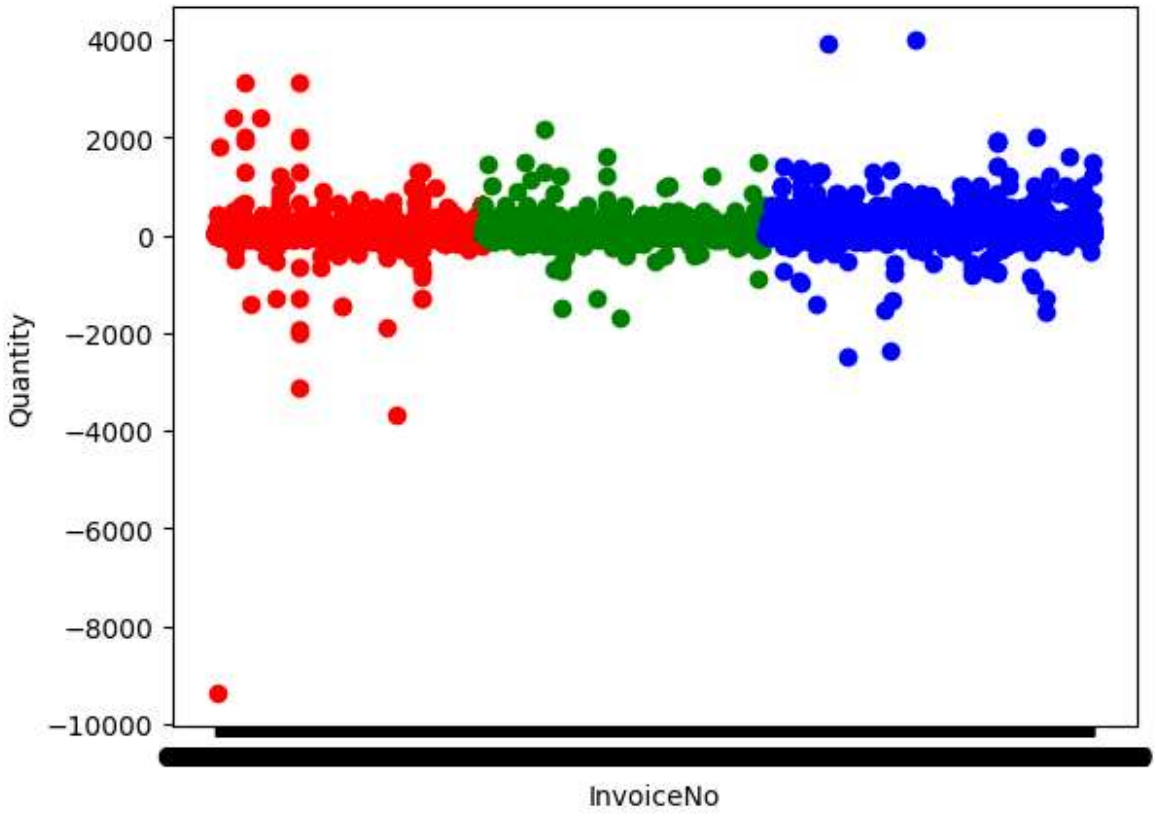
In [16]:
```
s["cluster"]=y_pred
s.head()
```

Out[16]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	c
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 [17]:
```
s1=s[s.cluster==0]
s2=s[s.cluster==1]
s3=s[s.cluster==2]
plt.scatter(s1["InvoiceNo"],s1["Quantity"],color="red")
plt.scatter(s2["InvoiceNo"],s2["Quantity"],color="green")
plt.scatter(s3["InvoiceNo"],s3["Quantity"],color="blue")
plt.xlabel("InvoiceNo")
plt.ylabel("Quantity")
```

Out[17]:
```
Text(0, 0.5, 'Quantity')
```



In [18]:
```
from sklearn.preprocessing import MinMaxScaler
```

In [19]:
```
scaler=MinMaxScaler()
scaler.fit(s[["Quantity"]])
s["Quantity"]=scaler.transform(s[["Quantity"]])
s.head()
```

Out[19]:

	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 [20]:
```
scaler.fit(s[["CustomerID"]])
s["CustomerID"]=scaler.transform(s[["CustomerID"]])
s.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	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	

kmean cluster

In [21]:
```
km=KMeans()
```

In [22]:
```
y_pred=km.fit_predict(s[["CustomerID","Quantity"]])
y_pred
```

```
C:\Users\mouni\AppData\Local\Programs\Python\Python310\lib\site-packages\skl
earn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` wi
ll change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to
suppress the warning
  warnings.warn(
```

Out[22]:
```
array([1, 1, 1, ..., 3, 3, 3])
```

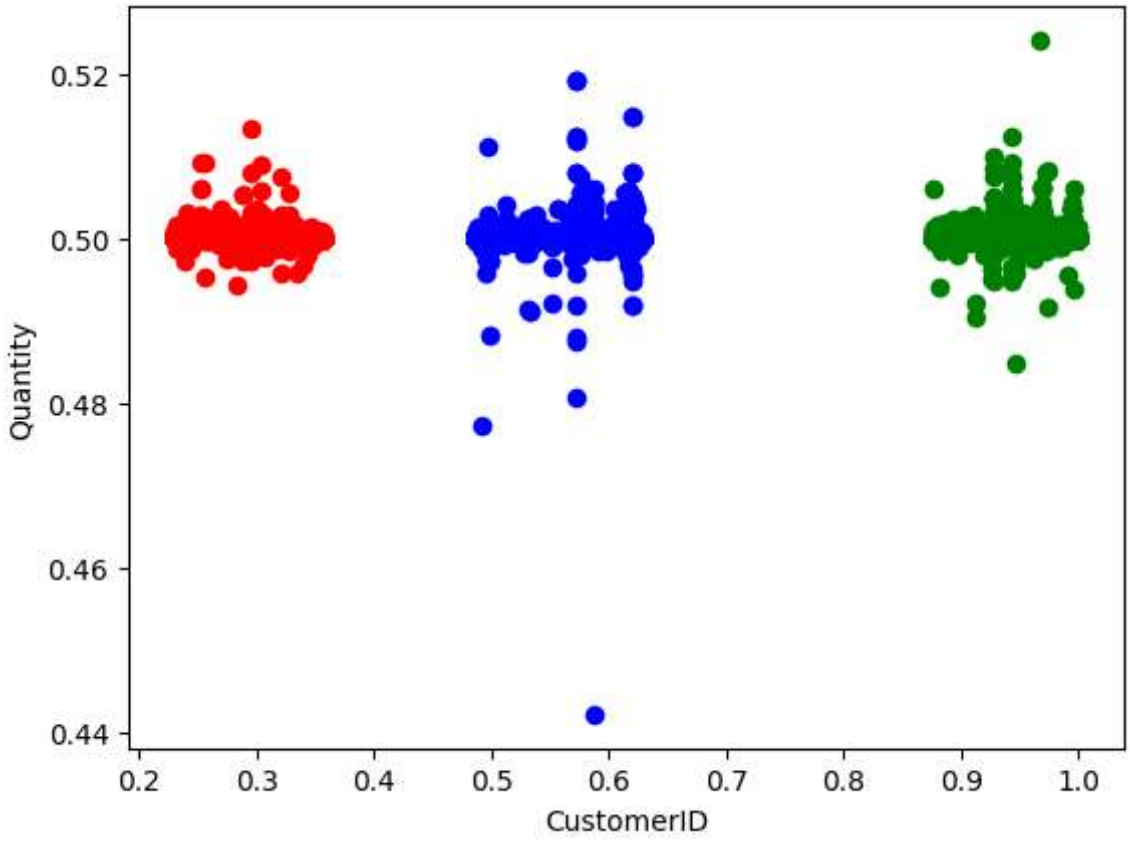
In [23]:
```
s["New Cluster"]=y_pred
s.head()
```

Out[23]:

	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 [28]:
```
s1=s[s["New Cluster"]==0]
s2=s[s["New Cluster"]==1]
s3=s[s["New Cluster"]==2]
plt.scatter(s1["CustomerID"],s1["Quantity"],color="red")
plt.scatter(s2["CustomerID"],s2["Quantity"],color="green")
plt.scatter(s3["CustomerID"],s3["Quantity"],color="blue")
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[28]:
```
Text(0, 0.5, 'Quantity')
```

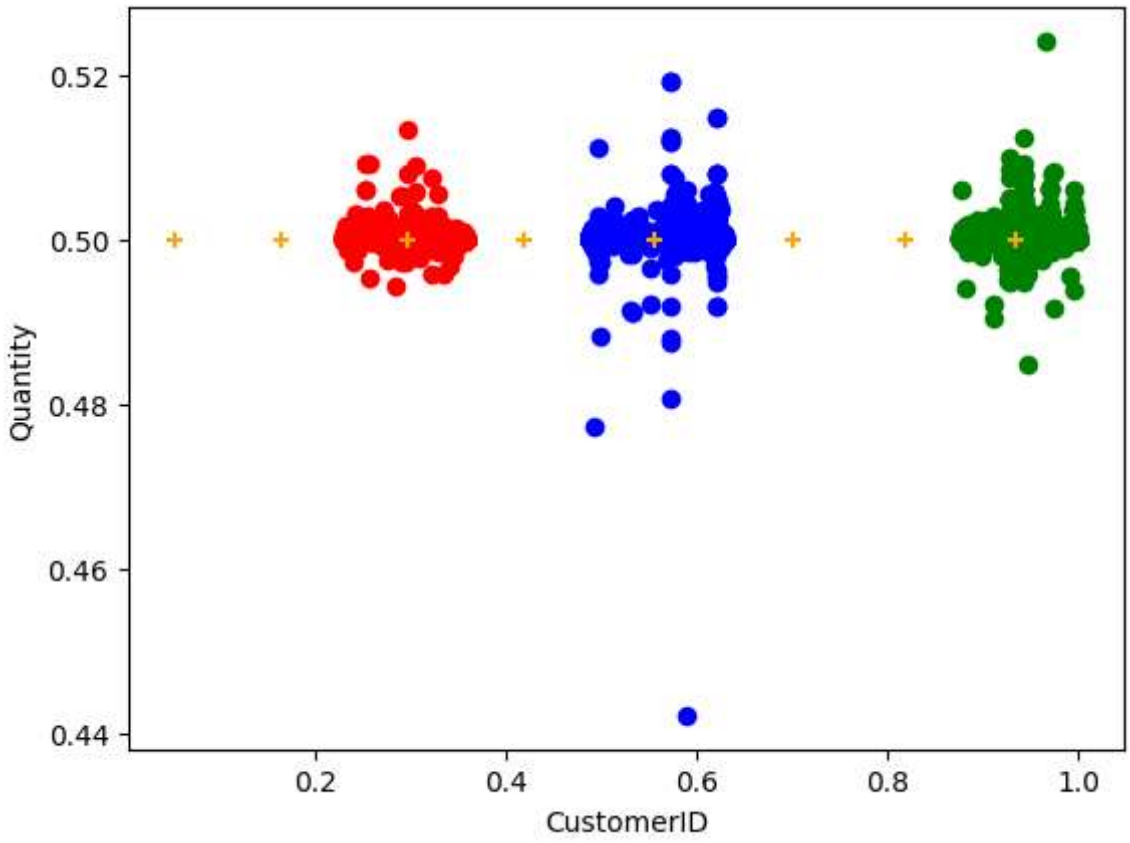


In [29]:
```
km.cluster_centers_
```

Out[29]:
```
array([[0.2960104 , 0.50006014],
       [0.93301823, 0.50005097],
       [0.55502897, 0.50005357],
       [0.05128805, 0.50006687],
       [0.41826663, 0.50006089],
       [0.81846964, 0.50006032],
       [0.70121934, 0.50005792],
       [0.16326876, 0.50006159]])
```

In [30]:
```
s1=s[s["New Cluster"]==0]
s2=s[s["New Cluster"]==1]
s3=s[s["New Cluster"]==2]
plt.scatter(s1["CustomerID"],s1["Quantity"],color="red")
plt.scatter(s2["CustomerID"],s2["Quantity"],color="green")
plt.scatter(s3["CustomerID"],s3["Quantity"],color="blue")
plt.scatter(km.cluster_centers_[:,0],km.cluster_centers_[:,1],color="orange",r
plt.xlabel("CustomerID")
plt.ylabel("Quantity")
```

Out[30]:
```
Text(0, 0.5, 'Quantity')
```



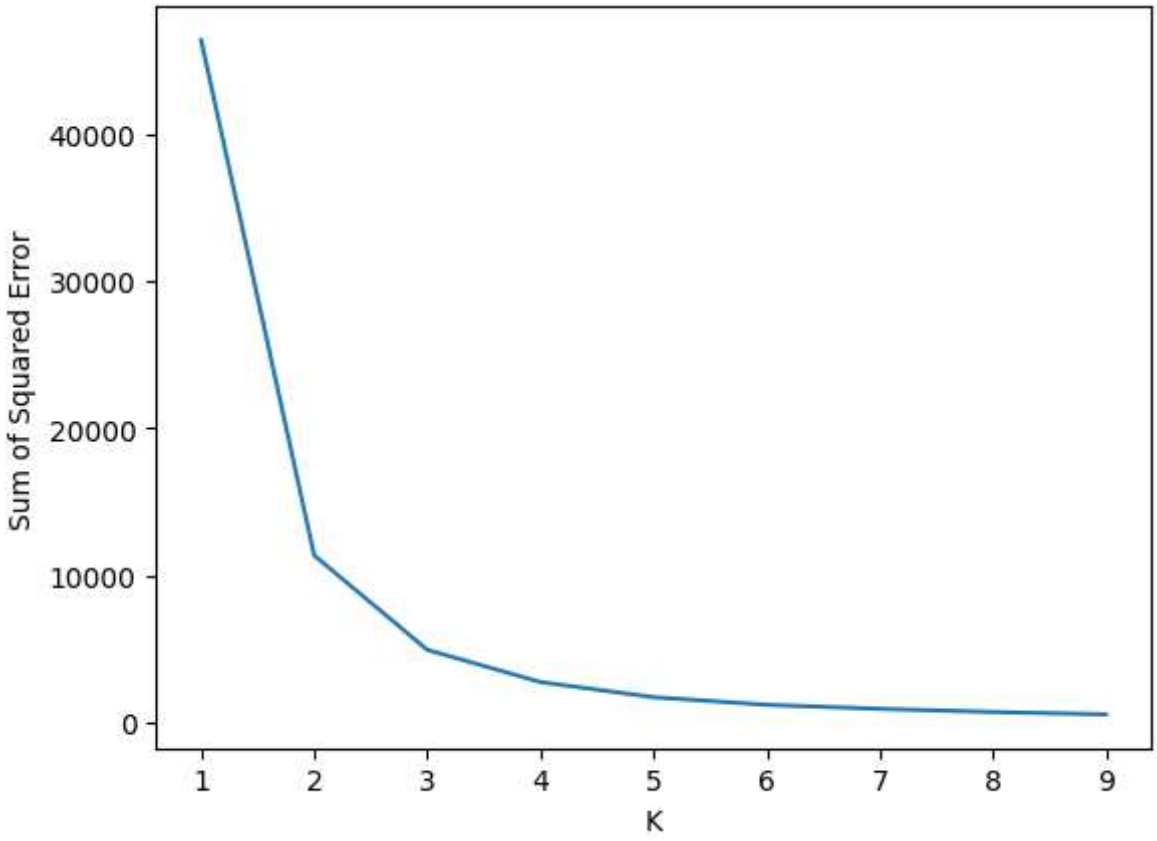
In [37]:
```
k_rng=range(1,10)
sse=[]
```

In [39]:
```
for k in k_rng:
    km=KMeans(n_clusters=k)
    km.fit(s[["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\mouni\AppData\Local\Programs\Python\Python310\lib\site-packages\skl
earn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` wi
ll change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to
suppress the warning
  warnings.warn(
C:\Users\mouni\AppData\Local\Programs\Python\Python310\lib\site-packages\skl
earn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` wi
ll change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to
suppress the warning
  warnings.warn(
C:\Users\mouni\AppData\Local\Programs\Python\Python310\lib\site-packages\skl
earn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` wi
ll change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to
suppress the warning
  warnings.warn(
C:\Users\mouni\AppData\Local\Programs\Python\Python310\lib\site-packages\skl
earn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` wi
ll change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to
suppress the warning
  warnings.warn(
C:\Users\mouni\AppData\Local\Programs\Python\Python310\lib\site-packages\skl
earn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` wi
ll change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to
suppress the warning
  warnings.warn(
C:\Users\mouni\AppData\Local\Programs\Python\Python310\lib\site-packages\skl
earn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` wi
ll change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to
suppress the warning
  warnings.warn(
C:\Users\mouni\AppData\Local\Programs\Python\Python310\lib\site-packages\skl
earn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` wi
ll change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to
suppress the warning
  warnings.warn(
C:\Users\mouni\AppData\Local\Programs\Python\Python310\lib\site-packages\skl
earn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` wi
ll change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to
suppress the warning
  warnings.warn(
C:\Users\mouni\AppData\Local\Programs\Python\Python310\lib\site-packages\skl
earn\cluster\_kmeans.py:870: FutureWarning: The default value of `n_init` wi
ll change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to
suppress the warning
  warnings.warn(
```

```
[46374.84553398371, 11336.065305485055, 4915.844333336665, 2723.51910518956
4, 1695.1451771972916, 1178.5718587781669, 902.5218165498411, 688.9201989697
517, 528.8093107707914]
```

Out[39]: 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
low error rate is more and the K-value is high error rate is very high.
So,finally we can Conclude the above
dataset is bestfit for K-Means.

In [ ]: