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 transistional data set known as online retail.The company primarily offers one-of-a-kind gifts gifts for every occasion.The company has a large number of wholesales 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 [1]:
```
import pandas as pd
from matplotlib import pyplot as plt
%matplotlib inline
```

In [3]:
```
df=pd.read_csv(r"C:\Users\Sushma sree\Downloads\OnlineRetail.csv")
df
```

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

541909 rows × 8 columns

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

Out[4]:

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

Out[5]:

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

In [6]:
```
1 df['InvoiceNo'].value_counts()
```

Out[6]:
```
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 [7]:
```
1 df['CustomerID'].value_counts()
```

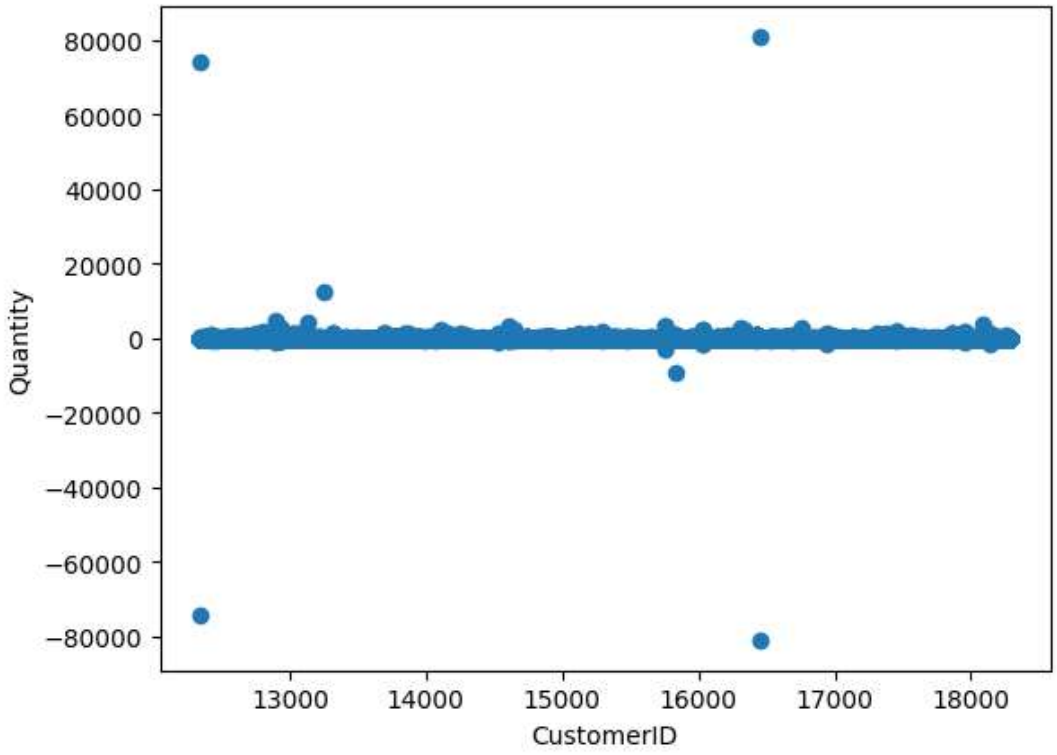
Out[7]:
```
CustomerID
17841.0    7983
14911.0    5903
14096.0    5128
12748.0    4642
14606.0    2782
           ...
15070.0       1
15753.0       1
17065.0       1
16881.0       1
16995.0       1
Name: count, Length: 4372, dtype: int64
```

In [8]:
```
1 df['Quantity'].value_counts()
```

Out[8]:
```
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 [9]:
```
1 plt.scatter(df['CustomerID'],df['Quantity'])
2 plt.xlabel("CustomerID")
3 plt.ylabel("Quantity")
```

Out[9]:
```
Text(0, 0.5, 'Quantity')
```



In [10]:
```
1 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 [11]:
```
1 df.isnull().sum()
```

Out[11]:
```
InvoiceNo          0
StockCode          0
Description     1454
Quantity           0
InvoiceDate        0
UnitPrice          0
CustomerID     135080
Country            0
dtype: int64
```

In [12]:
```
df.dropna(inplace=True)
```

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

Out[13]:
```
InvoiceNo      0
StockCode      0
Description    0
Quantity       0
InvoiceDate    0
UnitPrice      0
CustomerID     0
Country        0
dtype: int64
```

In [14]:
```
from sklearn.cluster import KMeans
km=KMeans()
km
```

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

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

```
C:\Users\Sushma sree\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[15]:
```
array([0, 0, 0, ..., 1, 1, 1])
```

In [16]:
```
df['Cluster']=y_pred
df.head()
```

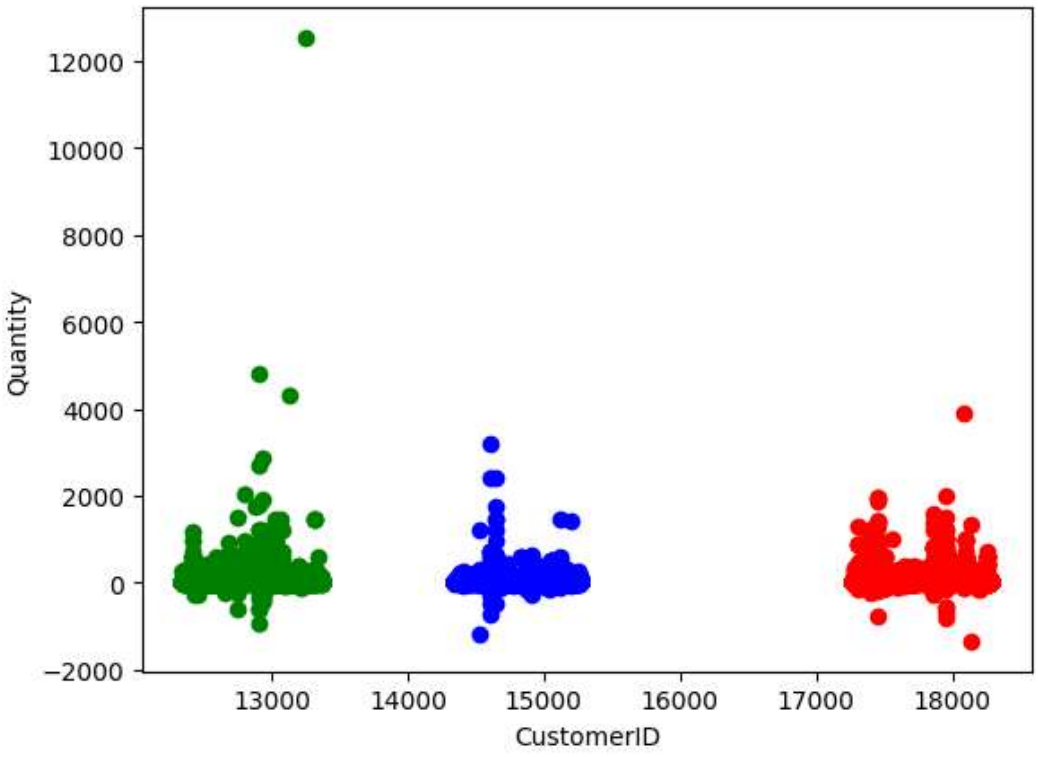
Out[16]:

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

In [18]:
```
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[18]: Text(0, 0.5, 'Quantity')



In [19]:
```
from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler()
scaler.fit(df[['Quantity']])
df['Quantity']=scaler.transform(df[['Quantity']])
```

In [20]:
```
scaler.fit(df[['CustomerID']])
df['CustomerID']=scaler.transform(df[['CustomerID']])
df.head()
```

Out[20]:

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

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

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

```
C:\Users\Sushma sree\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([0, 0, 0, ..., 5, 5, 5])
```

In [24]:
```
df['New Cluster']=y_pred
df.head()
```

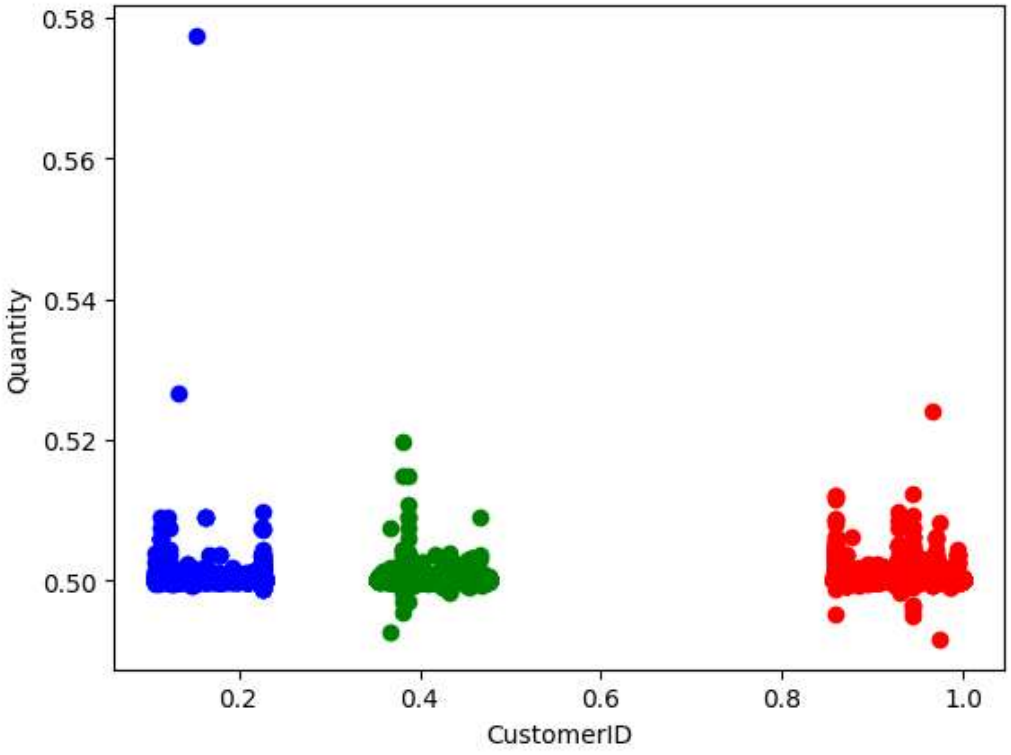
Out[24]:

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

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.xlabel('CustomerID')
plt.ylabel('Quantity')

```

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



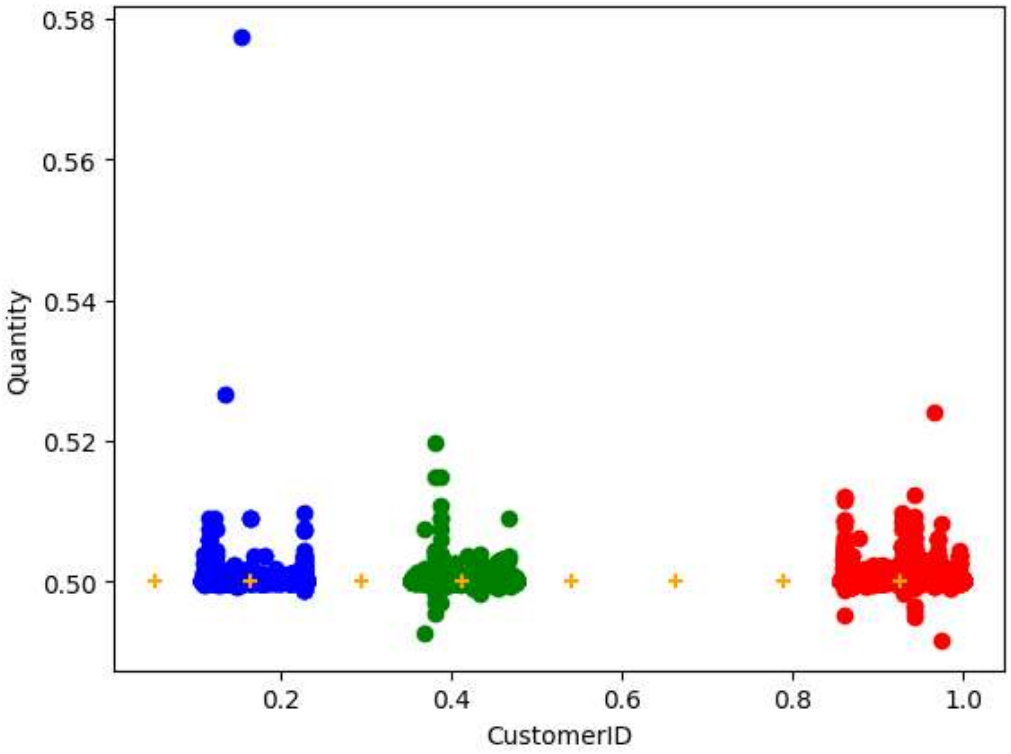
In [27]:
```
km.cluster_centers_
```

Out[27]:
```
array([[0.92462477, 0.5000745 ],
       [0.41117769, 0.50007085],
       [0.16316284, 0.50008248],
       [0.66180743, 0.50007346],
       [0.54048273, 0.50006341],
       [0.05076558, 0.50009106],
       [0.78776778, 0.50006619],
       [0.29395517, 0.50007685]])
```

In [28]:
```
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',marker='+')
plt.xlabel('CustomerID')
plt.ylabel('Quantity')

```

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



In [29]:
```
k_rng=range(1,10)
sse=[]
```

In [30]:

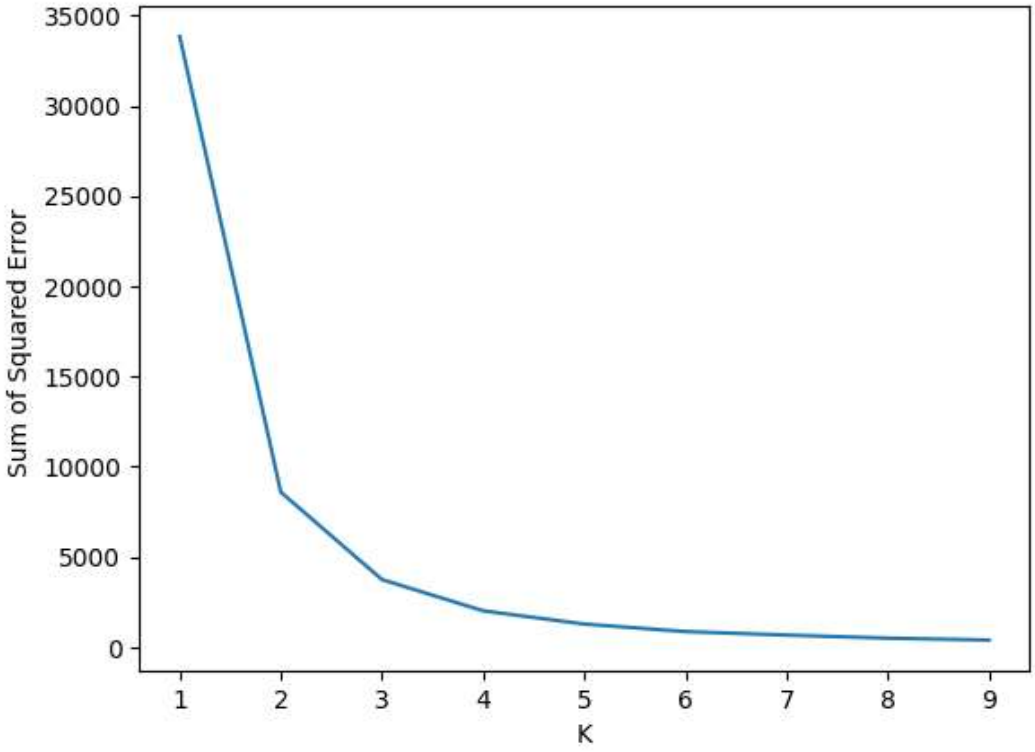
```python
for k in k_rng:
    km=KMeans(n_clusters=k)
    km.fit(df[['CustomerID','Quantity']])
    sse.append(km.inertia_)
print(sse)
plt.plot(k_rng,sse)
plt.xlabel('K')
plt.ylabel('Sum of Squared Error ')
```

```
C:\Users\Sushma sree\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(
C:\Users\Sushma sree\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(
C:\Users\Sushma sree\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(
C:\Users\Sushma sree\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(
C:\Users\Sushma sree\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(
C:\Users\Sushma sree\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(
C:\Users\Sushma sree\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(
C:\Users\Sushma sree\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(
C:\Users\Sushma sree\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(

[33847.22708730174, 8593.145395827489, 3751.8202675804705, 2018.3398426654294, 1286.8456695022774, 868.931811965003, 675.6357968309295, 503.7744959805072, 398.2260637878505]
```

Out[30]:

```
Text(0, 0.5, 'Sum of Squared Error ')
```



Conclusion

For the given dataset we use KMeans Clustering to Preict the best fit.when the K-value has low error rate has high accuracy.So,KMeans Clustering is the Best fit..

In [ ]:
```
1
```