PROBLEM STATEMENT:TO PREDICT THE BESTFIT MODEL FOR THE GIVEN DATASET

DATA COLLECTION

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

In [2]:
```
df=pd.read_csv(r"C:\Users\pavan\Downloads\OnlineRetail2.csv")
df
```

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	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 [3]:
```
# DATA CLEANING & PREPROCESSING
```

In [4]:
```
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]:
```
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]:
```
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]:
```
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]:
```
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]:
```
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 [10]:
```
df.isnull().sum()
```

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

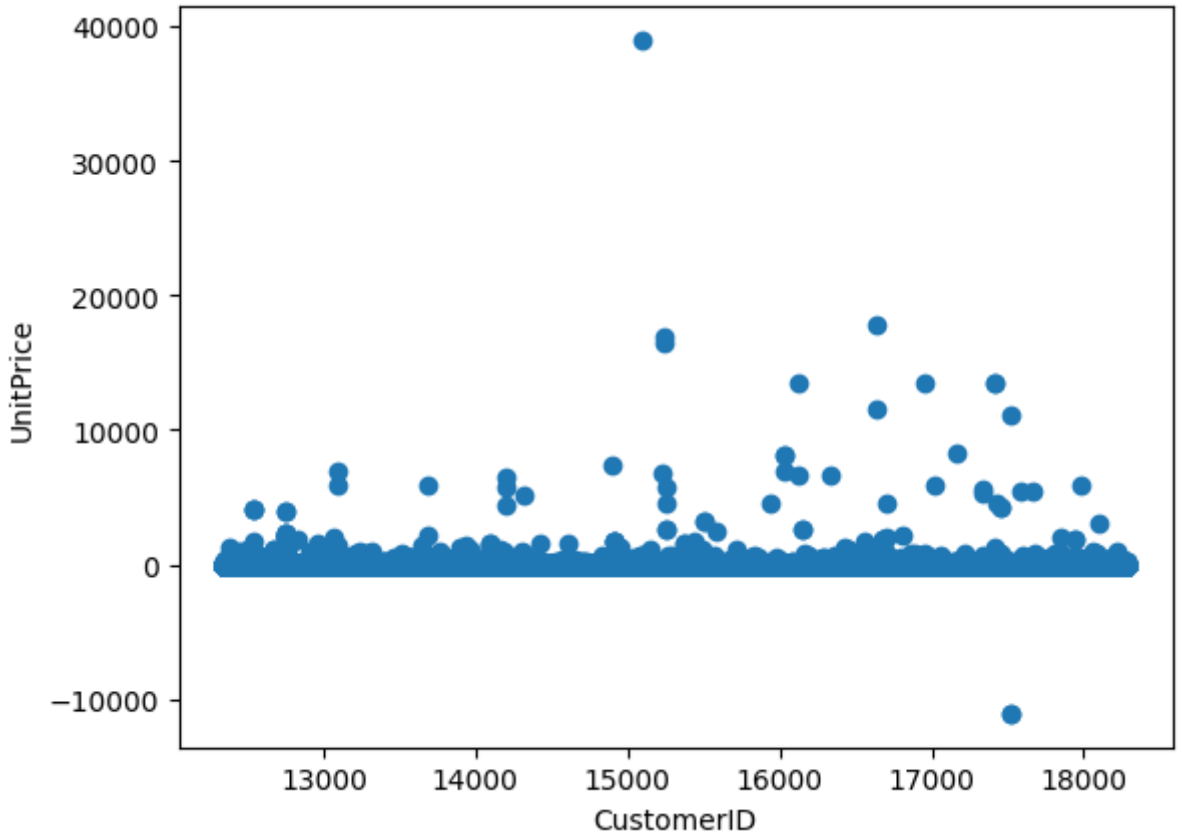
In [11]:
```
df.fillna(method='ffill',inplace=True)
```

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

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

DATA ANALYSIS

In [13]:
```
plt.scatter(df["CustomerID"],df["UnitPrice"])
plt.xlabel("CustomerID")
plt.ylabel("UnitPrice")
plt.show()
```



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

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

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

```
C:\Users\pavan\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster\_kmeans.py:870: FutureWarni
ng: 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(
```

Out[15]:
```
array([3, 3, 3, ..., 0, 0, 0])
```

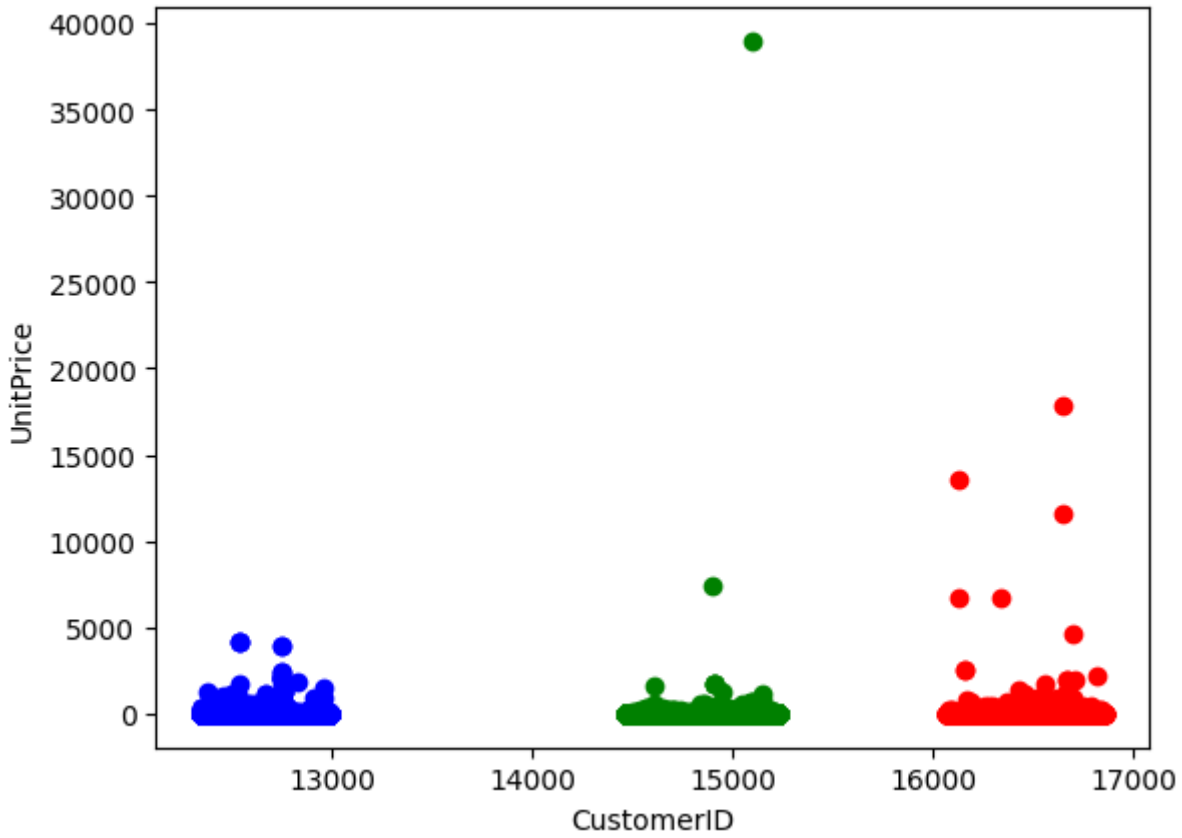
In [16]:
```
df["cluster"]=y_predicted
df.head()
```

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

In [17]:
```
df1=df[df.cluster==0]
df2=df[df.cluster==1]
df3=df[df.cluster==2]
plt.scatter(df1["CustomerID"],df1["UnitPrice"],color='blue')
plt.scatter(df2["CustomerID"],df2["UnitPrice"],color='red')
plt.scatter(df3["CustomerID"],df3["UnitPrice"],color='green')
plt.xlabel("CustomerID")
plt.ylabel("UnitPrice")
```

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



In [18]:
```
from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler()
scaler.fit(df[["CustomerID"]])
df["CustomerID"]=scaler.transform(df[["CustomerID"]])
df.head()
```

Out[18]:

	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	0.926443	United Kingdom	3
1	536365	71053	WHITE METAL LANTERN	6	01-12-2010 08:26	3.39	0.926443	United Kingdom	3
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	01-12-2010 08:26	2.75	0.926443	United Kingdom	3
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	01-12-2010 08:26	3.39	0.926443	United Kingdom	3
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	01-12-2010 08:26	3.39	0.926443	United Kingdom	3

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

Out[19]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	cluster
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	01-12-2010 08:26	0.221150	0.926443	United Kingdom	3
1	536365	71053	WHITE METAL LANTERN	6	01-12-2010 08:26	0.221167	0.926443	United Kingdom	3
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	01-12-2010 08:26	0.221154	0.926443	United Kingdom	3
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	01-12-2010 08:26	0.221167	0.926443	United Kingdom	3
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	01-12-2010 08:26	0.221167	0.926443	United Kingdom	3

In [20]:
```
km=KMeans()
```

In [21]:
```
y_predicted=km.fit_predict(df[["CustomerID","UnitPrice"]])
y_predicted
```

```
C:\Users\pavan\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\cluster\_kmeans.py:870: FutureWarni
ng: 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(
```

Out[21]:
```
array([6, 6, 6, ..., 4, 4, 4])
```

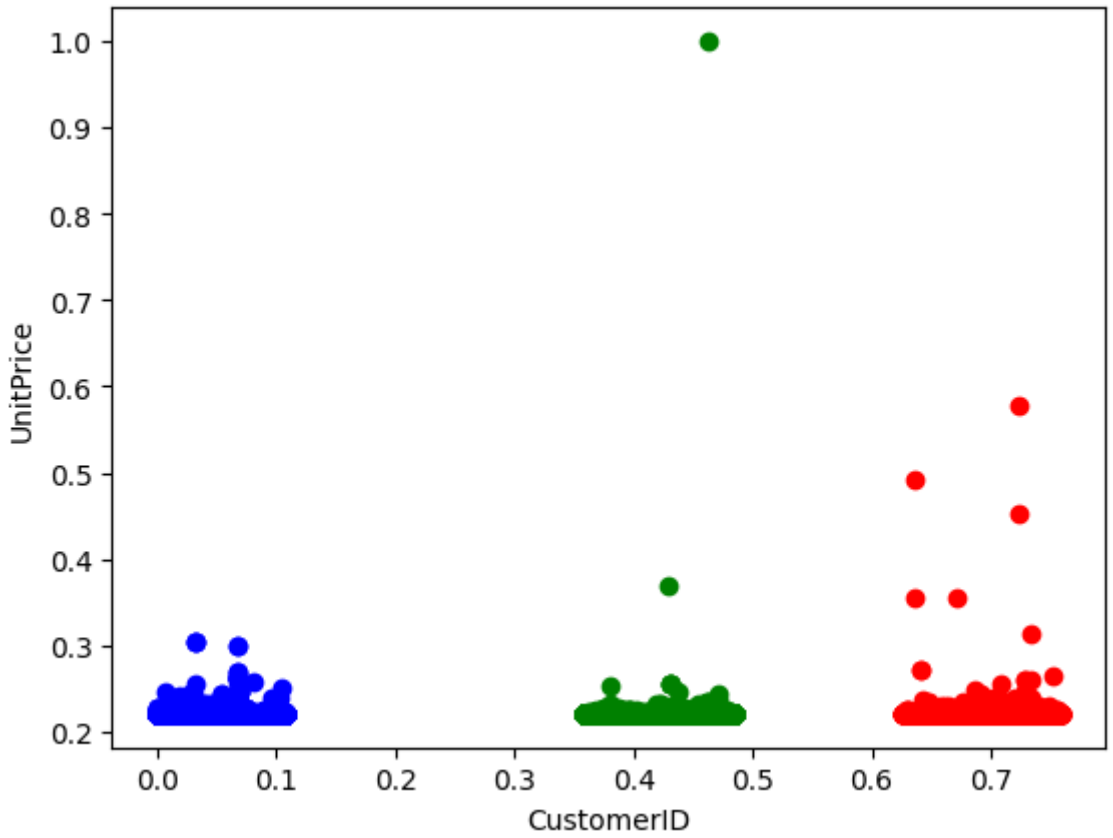
In [22]:
```
df["New Cluster"]=y_predicted
df.head()
```

Out[22]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country	cluster	New Cluster
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	01-12-2010 08:26	0.221150	0.926443	United Kingdom	3	6
1	536365	71053	WHITE METAL LANTERN	6	01-12-2010 08:26	0.221167	0.926443	United Kingdom	3	6
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	01-12-2010 08:26	0.221154	0.926443	United Kingdom	3	6
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	01-12-2010 08:26	0.221167	0.926443	United Kingdom	3	6
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	01-12-2010 08:26	0.221167	0.926443	United Kingdom	3	6

In [23]:
```
df1=df[df.cluster==0]
df2=df[df.cluster==1]
df3=df[df.cluster==2]
plt.scatter(df1["CustomerID"],df1["UnitPrice"],color='blue')
plt.scatter(df2["CustomerID"],df2["UnitPrice"],color='red')
plt.scatter(df3["CustomerID"],df3["UnitPrice"],color='green')
plt.xlabel("CustomerID")
plt.ylabel("UnitPrice")
```

Out[23]:
```
Text(0, 0.5, 'UnitPrice')
```

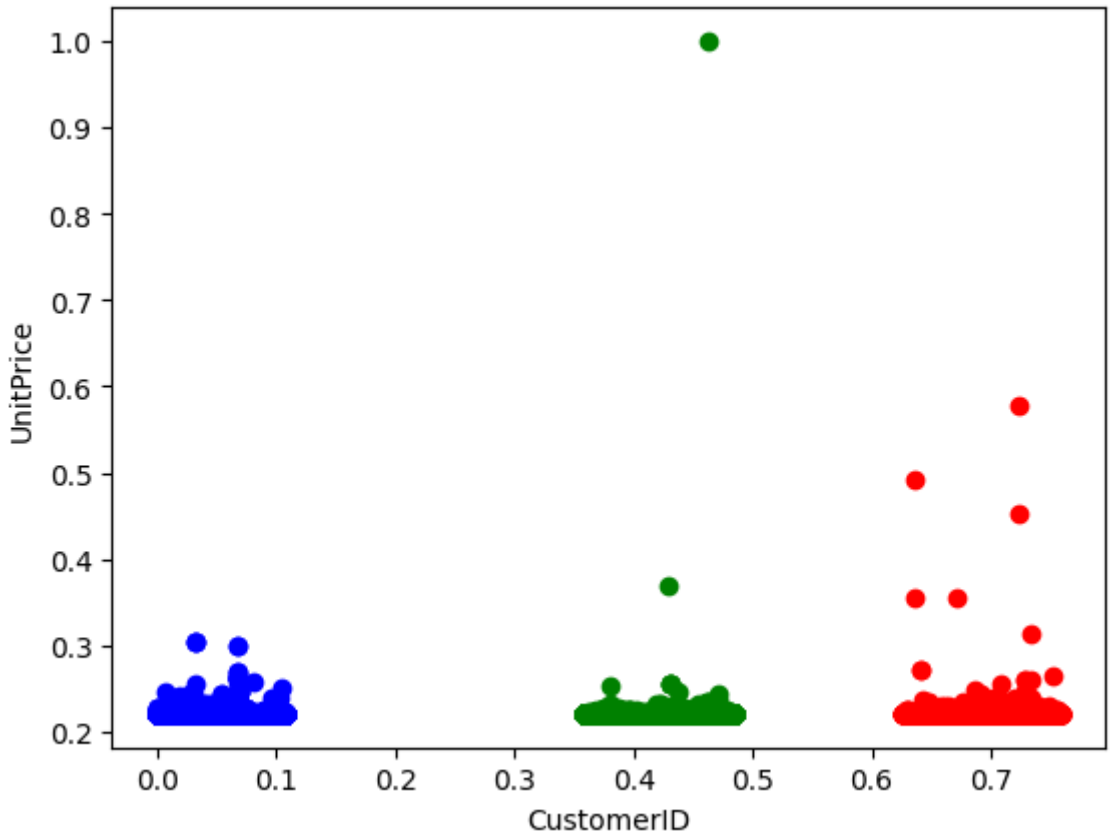


In [24]:
```
km.cluster_centers_
```

Out[24]:
```
array([[0.55342176, 0.22119907],
       [0.8185427 , 0.22119921],
       [0.290733  , 0.22118839],
       [0.41522894, 0.22118452],
       [0.05057095, 0.22120352],
       [0.70059748, 0.22119829],
       [0.93308721, 0.22117835],
       [0.15956932, 0.22118466]])
```

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

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



In [26]:
```
k_rng=range(1,10)
sse=[]
```

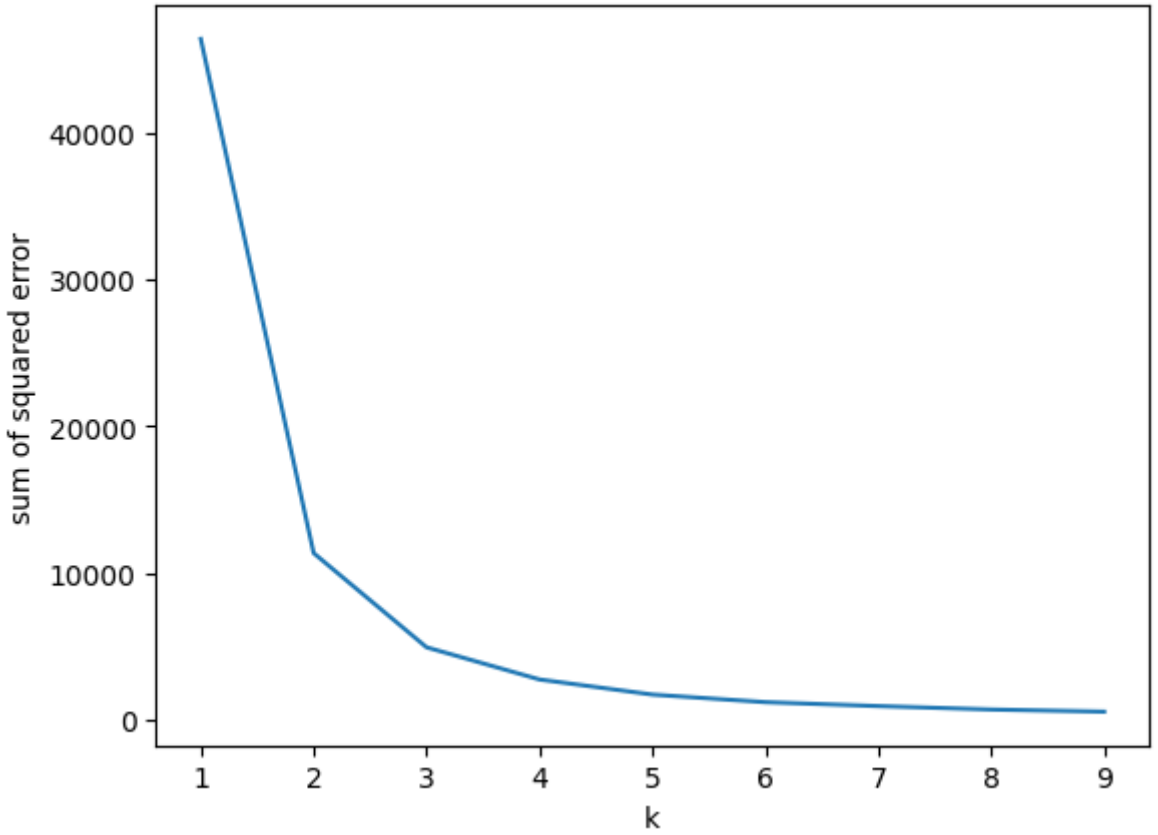
In [27]:
```
for k in k_rng:
    km=KMeans(n_clusters=k)
    km.fit(df[["CustomerID","UnitPrice"]])
    sse.append(km.inertia_)
sse
```

```
C:\Users\pavan\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\pavan\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\pavan\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\pavan\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\pavan\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\pavan\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\pavan\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\pavan\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\pavan\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[27]:
```
[46375.89020547945,
 11337.109981610258,
 4916.975167896996,
 2724.563781877092,
 1696.0847510016088,
 1179.6369942401582,
 912.6142364645152,
 678.2937278822499,
 531.7422029260085]
```

In [28]:
```
plt.plot(k_rng,sse)
plt.xlabel("k")
plt.ylabel("sum of squared error")
```

Out[28]:
```
Text(0, 0.5, 'sum of squared error')
```



CONCLUSION

The given dataset is "Online Retail".For the given dataset KMeans model suits more for the bestfit.

So,the KMeans model is the bestfit for the dataset "Online Retail"

In [ ]: