In [1]:
```python
import pandas as pd
import numpy as np
```

Objective:

From the transaction dataset, recommend which items can be sold/recommended together in a combo.

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

In [3]:
```python
df.shape
```

Out[3]:
```
(541909, 8)
```

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

Out[4]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	01/12/10 8:26	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	01/12/10 8:26	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	01/12/10 8:26	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	01/12/10 8:26	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	01/12/10 8:26	3.39	17850.0	United Kingdom

In [5]:
```python
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 [6]:
```
## exclude the return products by excluding the negative unitprice
```

In [7]:
```
df[df.UnitPrice<0]
```

Out[7]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
299983	A563186	B	Adjust bad debt	1	12/08/11 14:51	-11062.06	NaN	United Kingdom
299984	A563187	B	Adjust bad debt	1	12/08/11 14:52	-11062.06	NaN	United Kingdom

In [8]:
```
df=df[df['UnitPrice']>=0]
```

In [9]:
```
df.describe()
```

Out[9]:

	Quantity	UnitPrice	CustomerID
count	541907.000000	541907.000000	406829.000000
mean	9.552281	4.651957	15287.690570
std	218.081560	94.395447	1713.600303
min	-80995.000000	0.000000	12346.000000
25%	1.000000	1.250000	13953.000000
50%	3.000000	2.080000	15152.000000
75%	10.000000	4.130000	16791.000000
max	80995.000000	38970.000000	18287.000000

In [10]:

```
df.Country.value_counts()
```

```
Out[10]: Country
United Kingdom          495476
Germany                   9495
France                    8557
EIRE                      8196
Spain                     2533
Netherlands               2371
Belgium                   2069
Switzerland               2002
Portugal                  1519
Australia                 1259
Norway                    1086
Italy                      803
Channel Islands            758
Finland                    695
Cyprus                     622
Sweden                     462
Unspecified                446
Austria                    401
Denmark                    389
Japan                      358
Poland                     341
Israel                     297
USA                        291
Hong Kong                  288
Singapore                  229
Iceland                    182
Canada                     151
Greece                     146
Malta                      127
United Arab Emirates        68
European Community          61
RSA                         58
Lebanon                     45
Lithuania                   35
Brazil                      32
Czech Republic              30
Bahrain                     19
Saudi Arabia                10
Name: count, dtype: int64
```

In [11]:
```python
## Main data is from UK, lets keep one country data for analysis
df=df[df.Country=='United Kingdom']
df.Country.value_counts()
```

Out[11]:
```
Country
United Kingdom    495476
Name: count, dtype: int64
```

In [12]:
```python
data=df.groupby(['InvoiceNo','Description'])['Quantity'].sum().unstack().fillna(0)
data.head()
```

Out[12]:

Description	4 PURPLE FLOCK DINNER CANDLES	50'S CHRISTMAS GIFT BAG LARGE	DOLLY GIRL BEAKER	I LOVE LONDON MINI BACKPACK	NINE DRAWER OFFICE TIDY	OVAL WALL MIRROR DIAMANTE	RED SPOT GIFT BAG LARGE	SET 2 TEA TOWELS I LOVE LONDON	SPACEBOY BABY GIFT SET	TOADSTOOL BEDSIDE LIGHT	...	wrongly coded 20713
InvoiceNo												
536365	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0
536366	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0
536367	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0
536368	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0
536369	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0

5 rows × 4202 columns

In [13]:
```python
data=(data>0).astype('int')
```

In [14]:
```python
data=data.astype('bool')
```

In [15]:
```python
### 22038 invoices and 4202 items
```

In [16]:
```
## Apriori algorithm
```

In [17]:
```
pip install mlxtend
```

```
Requirement already satisfied: mlxtend in c:\users\kanwar\anaconda3\lib\site-packages (0.23.4)
Requirement already satisfied: scipy>=1.2.1 in c:\users\kanwar\anaconda3\lib\site-packages (from mlxtend) (1.15.3)
Requirement already satisfied: numpy>=1.16.2 in c:\users\kanwar\anaconda3\lib\site-packages (from mlxtend) (2.1.3)
Requirement already satisfied: pandas>=0.24.2 in c:\users\kanwar\anaconda3\lib\site-packages (from mlxtend) (2.2.3)
Requirement already satisfied: scikit-learn>=1.3.1 in c:\users\kanwar\anaconda3\lib\site-packages (from mlxtend) (1.6.1)
Requirement already satisfied: matplotlib>=3.0.0 in c:\users\kanwar\anaconda3\lib\site-packages (from mlxtend) (3.10.0)
Requirement already satisfied: joblib>=0.13.2 in c:\users\kanwar\anaconda3\lib\site-packages (from mlxtend) (1.4.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\kanwar\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxten
d) (1.3.1)
Requirement already satisfied: cycler>=0.10 in c:\users\kanwar\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend)
(0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\kanwar\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxten
d) (4.55.3)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\kanwar\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxten
d) (1.4.8)
Requirement already satisfied: packaging>=20.0 in c:\users\kanwar\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend)
(24.2)
Requirement already satisfied: pillow>=8 in c:\users\kanwar\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (11.
1.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\kanwar\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlxten
d) (3.2.0)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\kanwar\anaconda3\lib\site-packages (from matplotlib>=3.0.0->mlx
tend) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\kanwar\anaconda3\lib\site-packages (from pandas>=0.24.2->mlxtend) (202
4.1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\kanwar\anaconda3\lib\site-packages (from pandas>=0.24.2->mlxtend) (20
25.2)
Requirement already satisfied: six>=1.5 in c:\users\kanwar\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib>=
3.0.0->mlxtend) (1.17.0)
Requirement already satisfied: threadpoolctl>=3.1.0 in c:\users\kanwar\anaconda3\lib\site-packages (from scikit-learn>=1.3.1->m
lxtend) (3.5.0)
Note: you may need to restart the kernel to use updated packages.
```

In [18]:
```
from mlxtend.frequent_patterns import apriori
```

In [19]:
```python
## min_support 2.28 % means the item should be present in atleast 3% of the dataset
frequent_itemsets_plus=apriori(data,min_support=0.02,use_colnames=True).sort_values('support',ascending=False).reset_index(drc
```

In [20]:
```python
frequent_itemsets_plus['len']=frequent_itemsets_plus['itemsets'].apply(lambda x:len(x))
```

In [21]:
```python
frequent_itemsets_plus
```

Out[21]:

	support	itemsets	len
0	0.098285	(WHITE HANGING HEART T-LIGHT HOLDER)	1
1	0.087939	(JUMBO BAG RED RETROSPOT)	1
2	0.076459	(REGENCY CAKESTAND 3 TIER)	1
3	0.072330	(PARTY BUNTING)	1
4	0.063164	(LUNCH BAG RED RETROSPOT)	1
...	...	...	...
237	0.020329	(KNITTED UNION FLAG HOT WATER BOTTLE)	1
238	0.020329	(SMALL HEART MEASURING SPOONS)	1
239	0.020192	(CHOCOLATE THIS WAY METAL SIGN)	1
240	0.020147	(LUNCH BAG SUKI DESIGN , LUNCH BAG CARS BLUE)	2
241	0.020011	(PACK OF 60 SPACEBOY CAKE CASES)	1

242 rows × 3 columns

In [22]:
```python
frequent_itemsets_plus[frequent_itemsets_plus['len']>2]
```

Out[22]:

	support	itemsets	len
182	0.02237	(GREEN REGENCY TEACUP AND SAUCER, ROSES REGENC...	3

In [30]:
```
## what two items we can buy together or reccommend
from mlxtend.frequent_patterns import association_rules
```

In [31]:
```
data.shape
```

Out[31]:
```
(22038, 4202)
```

In [32]:
```
ass_rules=association_rules(frequent_itemsets_plus,metric='lift',min_threshold=1,num_itemsets=22038).sort_values('lift',ascend
```

In [33]:
```
ass_rules
```

Out[33]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	representativity	leverage	conviction	zhangs_m
0	(PINK REGENCY TEACUP AND SAUCER)	(GREEN REGENCY TEACUP AND SAUCER, ROSES REGENC...	0.031899	0.031809	0.022370	0.701280	22.046810	1.0	0.021356	3.241136	0.98
1	(GREEN REGENCY TEACUP AND SAUCER, ROSES REGENC...	(PINK REGENCY TEACUP AND SAUCER)	0.031809	0.031899	0.022370	0.703281	22.046810	1.0	0.021356	3.262685	0.98
2	(GREEN REGENCY TEACUP AND SAUCER)	(ROSES REGENCY TEACUP AND SAUCER , PINK REGENC...	0.042381	0.024775	0.022370	0.527837	21.304904	1.0	0.021320	2.065442	0.99
3	(ROSES REGENCY TEACUP AND SAUCER , PINK REGENC...	(GREEN REGENCY TEACUP AND SAUCER)	0.024775	0.042381	0.022370	0.902930	21.304904	1.0	0.021320	9.865279	0.97
4	(ROSES REGENCY TEACUP AND SAUCER)	(GREEN REGENCY TEACUP AND SAUCER,	0.043425	0.026182	0.022370	0.515152	19.675752	1.0	0.021233	2.008500	0.99

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	representativity	leverage	conviction	zhangs_m
		PINK REGENCY...									
...	...	...	...	...	...	...	...	...	...	...	
83	(ROSES REGENCY TEACUP AND SAUCER)	(REGENCY CAKESTAND 3 TIER)	0.043425	0.076459	0.020601	0.474399	6.204634	1.0	0.017281	1.757115	0.87
84	(PARTY BUNTING)	(SPOTTY BUNTING)	0.072330	0.047237	0.020510	0.283563	6.003045	1.0	0.017093	1.329864	0.89
85	(SPOTTY BUNTING)	(PARTY BUNTING)	0.047237	0.072330	0.020510	0.434198	6.003045	1.0	0.017093	1.639567	0.87
86	(JUMBO BAG RED RETROSPOT)	(LUNCH BAG RED RETROSPOT)	0.087939	0.063164	0.024095	0.273994	4.337842	1.0	0.018540	1.290397	0.84
87	(LUNCH BAG RED RETROSPOT)	(JUMBO BAG RED RETROSPOT)	0.063164	0.087939	0.024095	0.381466	4.337842	1.0	0.018540	1.474552	0.82

88 rows × 14 columns

In [36]:
```
ass_rules.iloc[7]
```

```
Out[36]: antecedents            (PINK REGENCY TEACUP AND SAUCER)
         consequents           (GREEN REGENCY TEACUP AND SAUCER)
         antecedent support                            0.031899
         consequent support                            0.042381
         support                                       0.026182
         confidence                                    0.820768
         lift                                         19.366261
         representativity                                   1.0
         leverage                                       0.02483
         conviction                                    5.342904
         zhangs_metric                                 0.979613
         jaccard                                        0.54434
         certainty                                     0.812836
         kulczynski                                    0.719271
         Name: 7, dtype: object
```

```
In [43]: ass_rules[ass_rules['antecedents']=={'PINK REGENCY TEACUP AND SAUCER'}]
```

Out[43]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	representativity	leverage	conviction	zhangs_met
0	(PINK REGENCY TEACUP AND SAUCER)	(GREEN REGENCY TEACUP AND SAUCER, ROSES REGENC...	0.031899	0.031809	0.022370	0.701280	22.046810	1.0	0.021356	3.241136	0.986(
7	(PINK REGENCY TEACUP AND SAUCER)	(GREEN REGENCY TEACUP AND SAUCER)	0.031899	0.042381	0.026182	0.820768	19.366261	1.0	0.024830	5.342904	0.979(
8	(PINK REGENCY TEACUP AND SAUCER)	(ROSES REGENCY TEACUP AND SAUCER)	0.031899	0.043425	0.024775	0.776671	17.885355	1.0	0.023390	4.283263	0.975'

In [ ]: