Name: Shahnwaz Khan Roll: AC-1272 PracticalQ: 04 Subject: Data Mining

In [ ]:
```
# Import the required libraries
import pandas as pd
from mlxtend.frequent_patterns import apriori
from mlxtend.frequent_patterns import association_rules
from mlxtend.preprocessing import TransactionEncoder
```

In [ ]:
```
# import the dataset
df=pd.read_csv('GroceryData.csv')
display(df)
```

	CustomerID	Bathing Soap	Beverages	Processed Food	Detergent	Utensils	Hair Colour	Skincare Products	Dish Soap
0	1001	0	0	1	1	0	0	1	0
1	1002	1	0	1	1	0	1	1	1
2	1003	1	0	1	0	0	1	1	1
3	1004	0	1	1	0	0	0	1	1
4	1005	0	0	1	0	0	0	1	0

In [ ]:
```
# we do not need customer ID. So drop it.
df.drop(columns='CustomerID',inplace=True)
```

In [ ]:
```
# calculate minimum support
freq=apriori(df,min_support=0.5,use_colnames=True)
freq
```

```
c:\Python310\lib\site-packages\mlxtend\frequent_patterns\fpcommon.py:111: DeprecationWarning: DataFrames with non-bool types result in worse computationalperformance and their support might be discontinued in the future.Please use a DataFrame with bool type
  warnings.warn(
```

Out[ ]:

	support	itemsets
0	1.0	(Processed Food)
1	1.0	(Skincare Products)
2	0.6	(Dish Soap)
3	1.0	(Processed Food, Skincare Products)
4	0.6	(Processed Food, Dish Soap)
5	0.6	(Skincare Products, Dish Soap)
6	0.6	(Processed Food, Skincare Products, Dish Soap)

In [ ]:
```
association_rules(freq,metric='confidence',min_threshold=0.25)
```

Out[ ]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conv

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conv
0	(Processed Food)	(Skincare Products)	1.0	1.0	1.0	1.0	1.0	0.0	
1	(Skincare Products)	(Processed Food)	1.0	1.0	1.0	1.0	1.0	0.0	
2	(Processed Food)	(Dish Soap)	1.0	0.6	0.6	0.6	1.0	0.0	
3	(Dish Soap)	(Processed Food)	0.6	1.0	0.6	1.0	1.0	0.0	
4	(Skincare Products)	(Dish Soap)	1.0	0.6	0.6	0.6	1.0	0.0	
5	(Dish Soap)	(Skincare Products)	0.6	1.0	0.6	1.0	1.0	0.0	
6	(Processed Food, Skincare Products)	(Dish Soap)	1.0	0.6	0.6	0.6	1.0	0.0	
7	(Processed Food, Dish Soap)	(Skincare Products)	0.6	1.0	0.6	1.0	1.0	0.0	
8	(Skincare Products, Dish Soap)	(Processed Food)	0.6	1.0	0.6	1.0	1.0	0.0	
9	(Processed Food)	(Skincare Products, Dish Soap)	1.0	0.6	0.6	0.6	1.0	0.0	
10	(Skincare Products)	(Processed Food, Dish Soap)	1.0	0.6	0.6	0.6	1.0	0.0	
11	(Dish Soap)	(Processed Food,	0.6	1.0	0.6	1.0	1.0	0.0	

In [ ]:
```
df=pd.read_csv('bread basket.csv')
df.head()
```

Out[ ]:

	Transaction	Item	date_time	period_day	weekday_weekend
0	1	Bread	30-10-2016 09:58	morning	weekend
1	2	Scandinavian	30-10-2016 10:05	morning	weekend
2	2	Scandinavian	30-10-2016 10:05	morning	weekend
3	3	Hot chocolate	30-10-2016 10:07	morning	weekend
4	3	Jam	30-10-2016 10:07	morning	weekend

In [ ]:
```
df=df.groupby('Transaction')['Item'].apply(list).reset_index(name='Item List')
df.head()
```

Out[ ]:

	Transaction	Item List

	Transaction	Item List
0	1	[Bread]
1	2	[Scandinavian, Scandinavian]
2	3	[Hot chocolate, Jam, Cookies]
3	4	[Muffin]

In [ ]:
```
dummies=TransactionEncoder()
dummi=dummies.fit(df['Item List']).transform(df['Item List'])
```

In [ ]:
```
dummi=pd.DataFrame(dummi,columns=dummies.columns_)
```

In [ ]:
```
dummi
```

Out[ ]:

	Adjustment	Afternoon with the baker	Alfajores	Argentina Night	Art Tray	Bacon	Baguette	Bakewell	Bare Popcorn
0	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...
9460	False	False	False	False	False	False	False	False	False
9461	False	False	False	False	False	False	False	False	False
9462	False	False	False	False	False	False	False	False	False
9463	False	False	False	False	False	False	False	False	False
9464	False	False	False	False	False	False	False	False	False

9465 rows × 94 columns

In [ ]:
```
freq=apriori(dummi,min_support=0.01,use_colnames=True)
freq
```

Out[ ]:

	support	itemsets
0	0.036344	(Alfajores)
1	0.016059	(Baguette)
2	0.327205	(Bread)
3	0.040042	(Brownie)
4	0.103856	(Cake)
...	...	...

	support	itemsets
56	0.023666	(Coffee, Toast)
57	0.014369	(Sandwich, Tea)
58	0.010037	(Cake, Coffee, Bread)
59	0.011199	(Pastry, Coffee, Bread)
60	0.010037	(Cake, Coffee, Tea)

In [ ]:
```
association_rules(freq,metric='confidence',min_threshold=0.5)
```

Out[ ]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage
0	(Alfajores)	(Coffee)	0.036344	0.478394	0.019651	0.540698	1.130235	0.002264
1	(Cake)	(Coffee)	0.103856	0.478394	0.054728	0.526958	1.101515	0.005044
2	(Cookies)	(Coffee)	0.054411	0.478394	0.028209	0.518447	1.083723	0.002179
3	(Hot chocolate)	(Coffee)	0.058320	0.478394	0.029583	0.507246	1.060311	0.001683
4	(Juice)	(Coffee)	0.038563	0.478394	0.020602	0.534247	1.116750	0.002154
5	(Medialuna)	(Coffee)	0.061807	0.478394	0.035182	0.569231	1.189878	0.005614
6	(Pastry)	(Coffee)	0.086107	0.478394	0.047544	0.552147	1.154168	0.006351
7	(Sandwich)	(Coffee)	0.071844	0.478394	0.038246	0.532353	1.112792	0.003877
8	(Scone)	(Coffee)	0.034548	0.478394	0.018067	0.522936	1.093107	0.001539
9	(Spanish Brunch)	(Coffee)	0.018172	0.478394	0.010882	0.598837	1.251766	0.002189
10	(Toast)	(Coffee)	0.033597	0.478394	0.023666	0.704403	1.472431	0.007593