In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn import metrics
from apyori import apriori
```

In [2]:

```
df = pd.read_csv('store_data.csv')
df.head()
```

Out[2]:

	shrimp	almonds	avocado	vegetables mix	green grapes	whole weat flour	yams	cottage cheese	energy drink	tomato juice	low fat yogurt	green tea	honey	salad	mineral water	salmon	a
0	burgers	meatballs	eggs	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	chutney	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	turkey	avocado	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	mineral water	milk	energy bar	whole wheat rice	green tea	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	low fat yogurt	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

In [3]:
```
df = pd.read_csv('store_data.csv',header = None)
df.head()
```

Out[3]:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
0	shrimp	almonds	avocado	vegetables mix	green grapes	whole weat flour	yams	cottage cheese	energy drink	tomato juice	low fat yogurt	green tea	honey	salad	mineral water	salmon	antioxyc ju
1	burgers	meatballs	eggs	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
2	chutney	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
3	turkey	avocado	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
4	mineral water	milk	energy bar	whole wheat rice	green tea	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N

In [4]:
```
df.shape
```

Out[4]: (7501, 20)

In [5]:
```
items = []
for i in range(0,df.shape[0]):
    items.append([str(df.values[i,j]) for j in range(0,20)])
```

In [6]:
```
items
```

Out[6]:
```
[['shrimp',
  'almonds',
  'avocado',
  'vegetables mix',
  'green grapes',
  'whole weat flour',
  'yams',
  'cottage cheese',
  'energy drink',
  'tomato juice',
  'low fat yogurt',
  'green tea',
  'honey',
  'salad',
  'mineral water',
  'salmon',
  'antioxydant juice',
  'frozen smoothie',
  'spinach',
  'olive oil']
```

In [7]:
```
final_items = []
for k in items:
    dummy = [x for x in k if x!='nan']
    final_items.append(dummy)
```

In [13]:
```
#final_items
```

In [9]:
```
rules = apriori(final_items,min_support = 0.0045,min_confidence = 0.2,min_lift = 3)
rules
```

Out[9]:
```
<generator object apriori at 0x7fd894abecf0>
```

In [10]:
```
list_rules = list(rules)
print(len(list_rules))
```

```
24
```

In [11]:
```
list_rules[0]
```

Out[11]:
```
RelationRecord(items=frozenset({'chicken', 'light cream'}), support=0.004532728969470737, ordered_statistics=[OrderedStatistic(items_base=frozenset({'light cream'}), items_add=frozenset({'chicken'}), confidence=0.29059829059829057, lift=4.84395061728395)])
```

In [12]:

```
for i in list_rules:
    pair = i[0]
    items = [x for x in pair]
    print("Rule : " + items[0] + "-->" +items[1])
    print("Support : "+ str(i[1]))
    print("confidence : " + str(i[2][0][2]))
    print("lift : "+str(i[2][0][3]))
    print("=====================================")
```

```
Rule : chicken-->light cream
Support : 0.004532728969470737
confidence : 0.29059829059829057
lift : 4.84395061728395
=====================================
Rule : mushroom cream sauce-->escalope
Support : 0.005732568990801226
confidence : 0.3006993006993007
lift : 3.790832696715049
=====================================
Rule : pasta-->escalope
Support : 0.005865884548726837
confidence : 0.3728813559322034
lift : 4.700811850163794
=====================================
Rule : ground beef-->herb & pepper
Support : 0.015997866951073192
confidence : 0.3234501347708895
lift : 3.2919938411349285
```

In [ ]:

```
# for single rule

pair = list_rules[0][0]
items = [x for x in pair]
print("Rule : " + items[0] + "-->" +items[1])
print("Support : "+ str(list_rules[0][1]))
print("confidence : " + str(list_rules[0][2][0][2]))
print("lift : "+str(list_rules[0][2][0][3]) )
```

In [ ]:

```

```

In [ ]:

```

```