In [1]:

```
data =[['milk','bread','rice','book'],
       ['bread','jam','book','pen'],
       ['jam','milk','bread','rice','eggs'],
       ['rice','eggs','pen','book'],
       ['eggs','pen','milk','bread','jam'],
       ['eggs','rice','bread','jam']]
```

In [2]:

```
from mlxtend.preprocessing import TransactionEncoder
te = TransactionEncoder()
te_array = te.fit_transform(data)
```

In [3]:

```
import pandas as pd
df=pd.DataFrame(te_array,columns=te.columns_)
df
```

Out[3]:

	book	bread	eggs	jam	milk	pen	rice
0	True	True	False	False	True	False	True
1	True	True	False	True	False	True	False
2	False	True	True	True	True	False	True
3	True	False	True	False	False	True	True
4	False	True	True	True	True	True	False
5	False	True	True	True	False	False	True

In [4]:

```
from mlxtend.frequent_patterns import apriori
itemset=apriori(df,min_support=0.6,use_colnames=True)
itemset
```

Out[4]:

	support	itemsets
0	0.833333	(bread)
1	0.666667	(eggs)
2	0.666667	(jam)
3	0.666667	(rice)
4	0.666667	(jam, bread)

In [5]:

```
from mlxtend.frequent_patterns import association_rules
res=association_rules(itemset,metric="confidence",min_threshold=0.6)
res
```

Out[5]:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	convic
0	(jam)	(bread)	0.666667	0.833333	0.666667	1.0	1.2	0.111111	
1	(bread)	(jam)	0.833333	0.666667	0.666667	0.8	1.2	0.111111	1.666

In [6]:
```
result = res[['antecedents','consequents','support','confidence','lift']]
result
```

Out[6]:

	antecedents	consequents	support	confidence	lift
0	(jam)	(bread)	0.666667	1.0	1.2
1	(bread)	(jam)	0.666667	0.8	1.2

In [7]:
```
result[result["confidence"]>=1]
```

Out[7]:

	antecedents	consequents	support	confidence	lift
0	(jam)	(bread)	0.666667	1.0	1.2

In [8]:
```
#Filtering rules based on condition
result[(result['lift']>=1) & (result['confidence'] >= 0.6)]
```

Out[8]:

	antecedents	consequents	support	confidence	lift
0	(jam)	(bread)	0.666667	1.0	1.2
1	(bread)	(jam)	0.666667	0.8	1.2