PRACTICAL-3

Run Apriori algorithm to find frequent item sets and association rules on 2 real datasets and use appropriate evaluation measures to compute correctness of obtained patterns:

a) Use minimum support as 50% and minimum confidence as 75%

b) Use minimum support as 60% and minimum confidence as 60 %

```
import pandas as pd
from mlxtend.preprocessing import TransactionEncoder
from mlxtend.frequent_patterns import fpgrowth, association_rules

groc = pd.read_csv("/groceries - groceries.csv", header=None)
MB = pd.read_csv("/Market_Basket_Optimisation.csv", header=None)

groc_list = groc.apply(lambda row: [str(i) for i in row.dropna()], axis=1).tolist()
MB_list = MB[0].apply(lambda x: [i.strip() for i in str(x).split(",")]).tolist()

te = TransactionEncoder()

groc_df = pd.DataFrame(te.fit(groc_list).transform(groc_list), columns=te.columns_)
MB_df = pd.DataFrame(te.fit(MB_list).transform(MB_list), columns=te.columns_)


freq_groc = fpgrowth(groc_df, min_support=0.1, use_colnames=True)
freq_MB = fpgrowth(MB_df, min_support=0.03, use_colnames=True) # Adjusted min_support

rules_groc = association_rules(freq_groc, metric="confidence", min_threshold=0.6)
rules_MB = association_rules(freq_MB, metric="confidence", min_threshold=0.6)

print("\nGroceries Frequent Itemsets:")
```

```
print("\nGroceries Frequent Itemsets:")
print(freq_groc.head())

print("\nMarket Basket Frequent Itemsets:")
print(freq_MB.head())

print("\nGroceries Rules:")
print(rules_groc.head())

print("\nMarket Basket Rules:")
print(rules_MB.head())
```

```
Groceries Frequent Itemsets:
    support         itemsets
0  0.102176              (4)
1  0.139488         (yogurt)
2  0.132066              (3)
3  0.104921  (tropical fruit)
4  0.255490      (whole milk)

Market Basket Frequent Itemsets:
    support            itemsets
0  0.043328            (shrimp)
1  0.076790           (burgers)
2  0.061059            (turkey)
3  0.076923      (mineral water)
4  0.049727  (frozen vegetables)

Groceries Rules:
Empty DataFrame
Columns: [antecedents, consequents, antecedent support, consequent support, support, confidence, lift, representativity, leverage, convicti
Index: []

Market Basket Rules:
Empty DataFrame
Columns: [antecedents, consequents, antecedent support, consequent support, support, confidence, lift, representativity, leverage, convicti
Index: []
```