```
pip install apyori

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from apyori import apriori

store_data = pd.read_csv("/content/Market Basket_Small dataset.csv",
header=None)
display(store_data.head(15))
print(store_data.shape)

transactions = []
for i in range(0, len(store_data)):
    transactions.append([str(store_data.values[i,j]) for j in range(0,
len(store_data.columns))])

association_rules = apriori(transactions, min_support=0.5,
min_confidence=0.7, min_lift=1.2, min_length=2)
association_results = list(association_rules)

print(len(association_results ))

print(association_results )

print("There are {} Relation
derived.".format(len(association_results)))

for i in range(0, len(association_results)):
    print(association_results[i][0])

# Import the transaction encoder function from mlxtend
from mlxtend.preprocessing import TransactionEncoder

# Instantiate transaction encoder and identify unique items
encoder = TransactionEncoder().fit(transactions)

# One-hot encode transactions
onehot = encoder.transform(transactions)

# Convert one-hot encoded data to DataFrame
onehot = pd.DataFrame(onehot, columns = encoder.columns_).drop('nan',
axis=1)

# Print the one
onehot.head()

# Import the association rules function
from mlxtend.frequent_patterns import apriori, association_rules

# Compute frequent itemsets using the Apriori algorithm
frequent_itemsets = apriori(onehot, min_support = 0.5,
                            max_len = 2, use_colnames = True)

# Compute all association rules using confidence
rules = association_rules(frequent_itemsets,
                            metric = "confidence",
                            min_threshold = 0.7)

# Print association rules
rules.info()

rules.head()
```