```
import numpy as np
import pandas as pd
pd.set_option('display.max_rows', None)
from matplotlib import pyplot as plt
%matplotlib inline
import seaborn as sns
sns.set_style('whitegrid')
from mlxtend.preprocessing import TransactionEncoder
from mlxtend.frequent_patterns import apriori, association_rules


market_basket_df = pd.read_csv('./Market_Basket_DayTwoTransactions.csv'
, header=None)
market_basket_df.head()
market_basket_df.shape


basket_items = []
for index, row in market_basket_df.iterrows():
   cleansed_items = [item for item in row if str(item)!='nan']
   basket_items.append(cleansed_items)
basket_items


tran_encod = TransactionEncoder()
tran_encod_list = tran_encod.fit(basket_items).transform(basket_items)
transaction_df = pd.DataFrame(tran_encod_list, columns=tran_encod.colum
ns_)
#transaction_df.shape
transaction_df.head()


item_count = {}
for col in transaction_df.columns:
    item_count[col] = transaction_df[col].sum()
item_freq_df = pd.DataFrame(data=list(item_count.values()), index=list(
item_count.keys()), columns=['frequency']).sort_values(by='frequency',
ascending=False)
item_freq_df.shape, item_freq_df.head(10)


plt.figure(figsize=(16,7))
sns.barplot(y=item_freq_df.index[:10], x=item_freq_df.frequency[:10])
plt.xticks(rotation=90)


apriori(transaction_df, min_support=0.1, use_colnames=True)


print(f'freq>200: {item_freq_df[item_freq_df.frequency>200].shape[0]} i
tems')
```

```
print(f'freq>100: {item_freq_df[item_freq_df.frequency>100].shape[0]} items')
print(f'freq>50: {item_freq_df[item_freq_df.frequency>50].shape[0]} items')


freq_itemset_support = apriori(transaction_df, min_support=0.05, use_colnames=True)
freq_itemset_support


overal_association_rules = association_rules(freq_itemset_support, metric="confidence", min_threshold=0.2)
overal_association_rules
```