```python

import numpy as np

import pandas as pd

from mlxtend.frequent_patterns import apriori, association_rules
```

```python
# Loading the Data

data = pd.read_excel('/content/Online Retail.xlsx')
data.head()
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_
  and should_run_async(code)
```

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39

```python
# Exploring the columns of the data
data.columns
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_
  and should_run_async(code)
Index(['InvoiceNo', 'StockCode', 'Description', 'Quantity', 'InvoiceDate',
       'UnitPrice', 'CustomerID', 'Country'],
      dtype='object')
```

```python
# Exploring the different regions of transactions
data.Country.unique()
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_
  and should_run_async(code)
array(['United Kingdom', 'France', 'Australia', 'Netherlands', 'Germany',
       'Norway', 'EIRE', 'Switzerland', 'Spain', 'Poland', 'Portugal',
       'Italy', 'Belgium', 'Lithuania', 'Japan', 'Iceland',
       'Channel Islands', 'Denmark', 'Cyprus', 'Sweden', 'Austria',
       'Israel', 'Finland', 'Bahrain', 'Greece', 'Hong Kong', 'Singapore',
       'Lebanon', 'United Arab Emirates', 'Saudi Arabia',
       'Czech Republic', 'Canada', 'Unspecified', 'Brazil', 'USA',
       'European Community', 'Malta', 'RSA'], dtype=object)
```

```
# Stripping extra spaces in the description

data['Description'] = data['Description'].str.strip()


# Dropping the rows without any invoice number

data.dropna(axis = 0, subset =['InvoiceNo'], inplace = True)

data['InvoiceNo'] = data['InvoiceNo'].astype('str')


# Dropping all transactions which were done on credit

data = data[~data['InvoiceNo'].str.contains('C')]
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipykernel.py:283: DeprecationWarning: `should_
  and should_run_async(code)
```

```
# Transactions done in France

basket_France = (data[data['Country'] =="France"]

          .groupby(['InvoiceNo', 'Description'])['Quantity']

          .sum().unstack().reset_index().fillna(0)

          .set_index('InvoiceNo'))


# Transactions done in the United Kingdom

basket_UK = (data[data['Country'] =="United Kingdom"]

          .groupby(['InvoiceNo', 'Description'])['Quantity']

          .sum().unstack().reset_index().fillna(0)

          .set_index('InvoiceNo'))


# Transactions done in Portugal

basket_Por = (data[data['Country'] =="Portugal"]

          .groupby(['InvoiceNo', 'Description'])['Quantity']

          .sum().unstack().reset_index().fillna(0)

          .set_index('InvoiceNo'))



basket_Sweden = (data[data['Country'] =="Sweden"]
basket_Sweden = (data[data['Country'] =="Sweden"]

          .groupby(['InvoiceNo', 'Description'])['Quantity']

          .sum().unstack().reset_index().fillna(0)

          .set_index('InvoiceNo'))
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should
  and should_run_async(code)
```

```
# Defining the hot encoding function to make the data suitable
# for the concerned libraries

def hot_encode(x):

    if(x<= 0):

        return 0

    if(x>= 1):

        return 1


# Encoding the datasets

basket_encoded = basket_France.applymap(hot_encode)

basket_France = basket_encoded



basket_encoded = basket_UK.applymap(hot_encode)

basket_UK = basket_encoded



basket_encoded = basket_Por.applymap(hot_encode)

basket_Por = basket_encoded



basket_encoded = basket_Sweden.applymap(hot_encode)

basket_Sweden = basket_encoded
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should
  and should_run_async(code)
```

```
# Building the model

frq_items = apriori(basket_France, min_support = 0.05, use_colnames = True)


# Collecting the inferred rules in a dataframe

rules = association_rules(frq_items, metric ="lift", min_threshold = 1)

rules = rules.sort_values(['confidence', 'lift'], ascending =[False, False])

print(rules.head())
```

```
                                          antecedents  \
44                         (JUMBO BAG WOODLAND ANIMALS)
258  (PLASTERS IN TIN CIRCUS PARADE, RED TOADSTOOL ...
270  (RED TOADSTOOL LED NIGHT LIGHT, PLASTERS IN TI...
301  (SET/20 RED RETROSPOT PAPER NAPKINS, SET/6 RED...
300  (SET/20 RED RETROSPOT PAPER NAPKINS, SET/6 RED...

                       consequents  antecedent support  consequent support  \
44                       (POSTAGE)            0.076531            0.765306
258                      (POSTAGE)            0.051020            0.765306
270                      (POSTAGE)            0.053571            0.765306
301  (SET/6 RED SPOTTY PAPER PLATES)            0.102041            0.127551
300    (SET/6 RED SPOTTY PAPER CUPS)            0.102041            0.137755

      support  confidence      lift  leverage  conviction  zhangs_metric
44   0.076531       1.000  1.306667  0.017961         inf       0.254144
258  0.051020       1.000  1.306667  0.011974         inf       0.247312
270  0.053571       1.000  1.306667  0.012573         inf       0.247978
301  0.099490       0.975  7.644000  0.086474   34.897959       0.967949
300  0.099490       0.975  7.077778  0.085433   34.489796       0.956294
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_
  and should_run_async(code)
/usr/local/lib/python3.10/dist-packages/mlxtend/frequent_patterns/fpcommon.py:110: Deprecation
  warnings.warn(
```

```
frq_items = apriori(basket_UK, min_support = 0.01, use_colnames = True)

rules = association_rules(frq_items, metric ="lift", min_threshold = 1)

rules = rules.sort_values(['confidence', 'lift'], ascending =[False, False])

print(rules.head())
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_
  and should_run_async(code)
/usr/local/lib/python3.10/dist-packages/mlxtend/frequent_patterns/fpcommon.py:110: Deprecation
  warnings.warn(
```

```
frq_items = apriori(basket_Por, min_support = 0.05, use_colnames = True)

rules = association_rules(frq_items, metric ="lift", min_threshold = 1)

rules = rules.sort_values(['confidence', 'lift'], ascending =[False, False])

print(rules.head())
```

```
frq_items = apriori(basket_Sweden, min_support = 0.05, use_colnames = True)

rules = association_rules(frq_items, metric ="lift", min_threshold = 1)

rules = rules.sort_values(['confidence', 'lift'], ascending =[False, False])

print(rules.head())
```