

**Nama : Taufikul Hakim**

**Nim : G.231.22.0071**

**Penjelasan Coding file Basket\_G231220071.ipynb**

**A. Impor Pustaka**

1. `import pandas as pd`: Mengimpor pustaka pandas yang digunakan untuk manipulasi dan analisis data.
2. `from mlxtend.frequent_patterns import apriori`: Mengimpor fungsi apriori dari pustaka mlxtend yang digunakan untuk menemukan itemset yang sering muncul.
3. `from mlxtend.frequent_patterns import association_rules`: Mengimpor fungsi association\_rules dari pustaka mlxtend yang digunakan untuk menghitung aturan asosiasi dari itemset yang sering muncul.

**B. Membaca Data**

4. `df = pd.read_excel('http://archive.ics.uci.edu/ml/machine-learning-databases/00352/Online%20Retail.xlsx')`: Membaca data dari file Excel yang terletak di URL yang diberikan dan menyimpannya dalam DataFrame df.
5. `df.head()`: Menampilkan lima baris pertama dari DataFrame untuk memeriksa struktur data.

**C. Pembersihan Data**

6. `df['Description'] = df['Description'].str.strip()`: Menghapus spasi di awal dan akhir teks dalam kolom 'Description'.
7. `df.dropna(axis=0, subset=['InvoiceNo'], inplace=True)`: Menghapus baris yang memiliki nilai NaN (kosong) pada kolom 'InvoiceNo'.
8. `df['InvoiceNo'] = df['InvoiceNo'].astype('str')`: Mengubah tipe data kolom 'InvoiceNo' menjadi string.

9. `df = df[~df['InvoiceNo'].str.contains('C')]`: Menghapus baris yang memiliki nilai 'InvoiceNo' yang mengandung huruf 'C'. Ini biasanya digunakan untuk menghapus transaksi yang dikembalikan.

#### D. Menyiapkan DataFrame Basket untuk Prancis

10. `basket = (df[df['Country'] == "France"] ... .set_index('InvoiceNo')):`
- Memfilter data untuk hanya negara Prancis.
  - Mengelompokkan data berdasarkan 'InvoiceNo' dan 'Description', kemudian menjumlahkan kolom 'Quantity'.
  - Mengubah format DataFrame dengan `unstack()` yang mengubah nilai deskripsi produk menjadi kolom, mengisi nilai NaN dengan 0, dan menetapkan 'InvoiceNo' sebagai indeks.

#### E. Mengubah Data ke Format One-Hot Encoding

11. `basket_sets = basket.applymap(lambda x: 1 if x > 0 else 0):` Mengubah nilai kuantitas menjadi 1 jika lebih dari 0, dan 0 jika tidak (one-hot encoding).

#### F. Menerapkan Algoritma Apriori

12. `frequent_itemsets = apriori(basket_sets, min_support=0.07, use_colnames=True):`
- Menjalankan algoritma Apriori pada `basket_sets` dengan `min_support` 7%, dan menyimpan hasilnya dalam `frequent_itemsets`.

#### G. Menghasilkan Aturan Asosiasi

13. `rules = association_rules(frequent_itemsets, metric="lift", min_threshold=1):`
- Menghasilkan aturan asosiasi dari `frequent_itemsets` dengan metrik lift dan threshold minimum 1.
14. `rules.head():` Menampilkan lima aturan pertama.

#### H. Memfilter Aturan Asosiasi

15. `rules_filtered = rules[(rules['lift'] >= 6) & (rules['confidence'] >= 0.8)]`: Memfilter aturan yang memiliki lift  $\geq 6$  dan confidence  $\geq 0.8$ , dan menyimpan hasilnya dalam `rules_filtered`.

I. Menghitung Jumlah Item Tertentu

16. `total_green = basket['ALARM CLOCK BAKELIKE GREEN'].sum()`: Menghitung total kuantitas untuk item 'ALARM CLOCK BAKELIKE GREEN' dan menyimpannya dalam `total_green`.
17. `total_red = basket['ALARM CLOCK BAKELIKE RED'].sum()`: Menghitung total kuantitas untuk item 'ALARM CLOCK BAKELIKE RED' dan menyimpannya dalam `total_red`.

J. Mendefinisikan Fungsi `encode_units`

18. `def encode_units(x): ... return 1 if x >= 1 else 0`: Mendefinisikan fungsi `encode_units` yang mengembalikan 1 jika nilai  $x \geq 1$ , jika tidak mengembalikan 0.

K. Menyiapkan DataFrame Basket untuk Jerman

19. `basket2 = (df[df['Country'] == "Germany"] ... .set_index('InvoiceNo'))`:

- Memfilter data untuk hanya negara Jerman.
- Mengelompokkan data berdasarkan 'InvoiceNo' dan 'Description', kemudian menjumlahkan kolom 'Quantity'.
- Mengubah format DataFrame dengan `unstack()` yang mengubah nilai deskripsi produk menjadi kolom, mengisi nilai NaN dengan 0, dan menetapkan 'InvoiceNo' sebagai indeks.

L. Mengubah Data ke Format One-Hot Encoding dan Menghapus Kolom POSTAGE

20. `basket_sets2 = basket2.applymap(encode_units)`: Menggunakan fungsi `encode_units` untuk mengubah nilai menjadi 1 atau 0 dalam DataFrame `basket_sets2`.

21. `basket_sets2.drop('POSTAGE', inplace=True, axis=1):` Menghapus kolom 'POSTAGE' dari DataFrame basket\_sets2.

M. Menerapkan Algoritma Apriori untuk Data dari Jerman

22. `frequent_itemsets2 = apriori(basket_sets2, min_support=0.05, use_colnames=True):`  
Menjalankan algoritma Apriori pada basket\_sets2 dengan min\_support 5%, dan menyimpan hasilnya dalam frequent\_itemsets2.
23. `rules2 = association_rules(frequent_itemsets2, metric="lift", min_threshold=1):`  
Menghasilkan aturan asosiasi dari frequent\_itemsets2 dengan metrik lift dan threshold minimum 1.

N. Memfilter Aturan Asosiasi untuk Jerman

24. `filtered_rules2 = rules2[(rules2['lift'] >= 4) & (rules2['confidence'] >= 0.5)]:`  
Memfilter aturan yang memiliki lift  $\geq 4$  dan confidence  $\geq 0.5$ , dan menyimpan hasilnya dalam filtered\_rules2.