1. Understanding the Dataset

The dataset contains the following columns:

- **Invoice Number**: Identifies a transaction.
- **Stock Code**: Product ID.
- **Description**: Product description.
- **Quantity**: Number of items purchased.
- **Invoice Date**: Date of the transaction.
- **Unit Price**: Price of one product.
- **Customer ID**: Identifies the customer.
- **Country**: Country where the transaction occurred.

We will use this data to recommend products based on customer behavior, such as frequent purchases or associations between products.

2. Project Overview

The goal is to build a recommendation system. Here are the two main types of recommendation systems:

- **Collaborative Filtering**: Based on user-item interactions (e.g., customers who bought X also bought Y).
- **Content-Based Filtering**: Based on product attributes (e.g., similar descriptions).

For this project, we'll likely focus on **Collaborative Filtering** since we're working with transaction data.

3. Loading the Data

You'll begin by importing the necessary libraries and loading the dataset.

[1]:
```
!pip install pandas
```

```
Requirement already satisfied: pandas in c:\users\bosss\appdata\local\programs\python\python313\lib\site-packages (2.2.3)
Requirement already satisfied: numpy>=1.26.0 in c:\users\bosss\appdata\local\programs\python\python313\lib\site-packages (from pandas) (2.1.2)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\bosss\appdata\local\programs\python\python313\lib\site-packages (from pandas) (2.9.0.p
ost0)
Requirement already satisfied: pytz>=2020.1 in c:\users\bosss\appdata\local\programs\python\python313\lib\site-packages (from pandas) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in c:\users\bosss\appdata\local\programs\python\python313\lib\site-packages (from pandas) (2024.2)
Requirement already satisfied: six>=1.5 in c:\users\bosss\appdata\local\programs\python\python313\lib\site-packages (from python-dateutil>=2.8.2->panda
s) (1.16.0)
```

[3]:
```
import pandas as pd
```

```
data = pd.read_excel('gu.xlsx')
```

```
print(data.head(6))
```

```
  InvoiceNo StockCode                          Description  Quantity  \
0    536365    85123A   WHITE HANGING HEART T-LIGHT HOLDER         6
1    536365     71053                  WHITE METAL LANTERN         6
2    536365    84406B       CREAM CUPID HEARTS COAT HANGER         8
3    536365    84029G  KNITTED UNION FLAG HOT WATER BOTTLE         6
4    536365    84029E       RED WOOLLY HOTTIE WHITE HEART.         6
5    536365     22752         SET 7 BABUSHKA NESTING BOXES         2

          InvoiceDate  UnitPrice  CustomerID         Country
0 2010-12-01 08:26:00       2.55     17850.0  United Kingdom
1 2010-12-01 08:26:00       3.39     17850.0  United Kingdom
2 2010-12-01 08:26:00       2.75     17850.0  United Kingdom
3 2010-12-01 08:26:00       3.39     17850.0  United Kingdom
4 2010-12-01 08:26:00       3.39     17850.0  United Kingdom
5 2010-12-01 08:26:00       7.65     17850.0  United Kingdom
```

```
data
```

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	2011-12-09 12:50:00	0.85	12680.0	France
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	2011-12-09 12:50:00	2.10	12680.0	France
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	2011-12-09 12:50:00	4.15	12680.0	France
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	2011-12-09 12:50:00	4.15	12680.0	France
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	2011-12-09 12:50:00	4.95	12680.0	France

541909 rows × 8 columns

4. Data Preprocessing

- **Handling Missing Values**: Check for any missing values and handle them accordingly.
- **Data Formatting**: Ensure the data is in the correct format for analysis.
- **Duplicate Removal**: You may also want to remove duplicate entries, if any.

```
print(data.isnull().sum())
```

```
InvoiceNo           0
StockCode           0
Description      1454
Quantity            0
InvoiceDate         0
UnitPrice           0
CustomerID     135080
Country             0
dtype: int64
```

```
data = data.dropna(subset=['CustomerID'])
```

```
data['InvoiceDate'] = pd.to_datetime(data['InvoiceDate'])
```

5. Exploratory Data Analysis (EDA)

Before building the recommendation system, it's essential to understand the data. Analyze aspects such as:

- Most popular products.
- Products frequently bought together.

```
# Most frequently purchased products
popular_products = data.groupby('Description')['Quantity'].sum().sort_values(ascending=False).head(10)
print(popular_products)
```

```
Description
WORLD WAR 2 GLIDERS ASSTD DESIGNS     53215
JUMBO BAG RED RETROSPOT               45066
ASSORTED COLOUR BIRD ORNAMENT         35314
WHITE HANGING HEART T-LIGHT HOLDER    34147
PACK OF 72 RETROSPOT CAKE CASES       33409
POPCORN HOLDER                        30504
RABBIT NIGHT LIGHT                    27094
MINI PAINT SET VINTAGE                25880
PACK OF 12 LONDON TISSUES             25321
PACK OF 60 PINK PAISLEY CAKE CASES    24163
Name: Quantity, dtype: int64
```

6. Building the Recommendation System

We can now move on to the recommendation system. Here's how to build it:

Step 1: Create a User-Item Matrix

This matrix will show the quantity of each product purchased by each customer.

```
# Create the user-item matrix
user_item_matrix = data.pivot_table(index='CustomerID', columns='Description', values='Quantity', aggfunc='sum', fill_val
print(user_item_matrix.head())
```

```
Description  10 COLOUR SPACEBOY PEN  12 COLOURED PARTY BALLOONS  \
CustomerID
12346.0                           0                           0
12347.0                           0                           0
12348.0                           0                           0
12349.0                           0                           0
12350.0                           0                           0

Description  12 DAISY PEGS IN WOOD BOX  12 EGG HOUSE PAINTED WOOD  \
CustomerID
12346.0                              0                          0
12347.0                              0                          0
12348.0                              0                          0
12349.0                              0                          0
12350.0                              0                          0

Description  12 HANGING EGGS HAND PAINTED  12 IVORY ROSE PEG PLACE SETTINGS  \
CustomerID
12346.0                                 0                                 0
12347.0                                 0                                 0
12348.0                                 0                                 0
12349.0                                 0                                 0
12350.0                                 0                                 0

Description  12 MESSAGE CARDS WITH ENVELOPES  12 PENCIL SMALL TUBE WOODLAND  \
CustomerID
12346.0                                    0                              0
12347.0                                    0                              0
12348.0                                    0                              0
12349.0                                    0                              0
12350.0                                    0                              0

Description  12 PENCILS SMALL TUBE RED RETROSPOT  12 PENCILS SMALL TUBE SKULL  \
CustomerID
12346.0                                        0                            0
12347.0                                        0                            0
12348.0                                        0                            0
12349.0                                        0                            0
12350.0                                        0                            0

Description  ...  ZINC STAR T-LIGHT HOLDER  ZINC SWEETHEART SOAP DISH  \
CustomerID   ...
12346.0      ...                         0                          0
12347.0      ...                         0                          0
12348.0      ...                         0                          0
12349.0      ...                         0                          0
12350.0      ...                         0                          0
```

Step 2: Apply Collaborative Filtering

We'll use **Cosine Similarity** or another metric to measure the similarity between customers.

```
from sklearn.metrics.pairwise import cosine_similarity

# Compute the cosine similarity between customers
customer_similarity = cosine_similarity(user_item_matrix)

# Convert the matrix to a DataFrame
customer_similarity_df = pd.DataFrame(customer_similarity, index=user_item_matrix.index, columns=user_item_matrix.index)
print(customer_similarity_df.head())
```

```
CustomerID  12346.0   12347.0   12348.0   12349.0   12350.0   12352.0  \
CustomerID
12346.0         0.0  0.000000  0.000000  0.000000  0.000000  0.000000
12347.0         0.0  1.000000  0.148879  0.020750  0.014435  0.034833
12348.0         0.0  0.148879  1.000000  0.000169  0.000315  0.001578
12349.0         0.0  0.020750  0.000169  1.000000  0.030121  0.072258
12350.0         0.0  0.014435  0.000315  0.030121  1.000000  0.001938

CustomerID  12353.0   12354.0   12355.0   12356.0  ...  18273.0  18274.0  \
CustomerID                                         ...
12346.0         0.0  0.000000  0.000000  0.000000  ...      0.0      0.0
12347.0         0.0  0.022843  0.506252  0.186107  ...      0.0      0.0
12348.0         0.0  0.010634  0.286226  0.226244  ...      0.0      0.0
12349.0         0.0  0.004931  0.000180  0.150819  ...      0.0      0.0
12350.0         0.0  0.000000  0.000000  0.001179  ...      0.0      0.0

CustomerID   18276.0   18277.0   18278.0   18280.0  18281.0   18282.0  \
CustomerID
12346.0     0.000000  0.000000  0.000000  0.000000      0.0  0.000000
12347.0     0.407060 -0.001245  0.015133  0.037236      0.0  0.011921
12348.0     0.168758  0.000000  0.000000  0.000000      0.0  0.000000
12349.0     0.000000 -0.000344  0.015680  0.000000      0.0  0.014689
12350.0     0.000000  0.000000  0.000000  0.000000      0.0  0.000000

CustomerID   18283.0   18287.0
CustomerID
12346.0     0.000000  0.000000
12347.0     0.075476  0.108942
12348.0     0.177440  0.110096
12349.0     0.038343  0.005644
12350.0     0.021421  0.000000

[5 rows x 4372 columns]
```

Step 3: Recommend Products

For a given customer, we can recommend products based on what similar customers have purchased.

```
def recommend_products(customer_id, user_item_matrix, customer_similarity_df, n_recommendations=5):
    # Get the similar customers
    similar_customers = customer_similarity_df[customer_id].sort_values(ascending=False).index[1:]

    # Get products bought by similar customers
    similar_customer_products = user_item_matrix.loc[similar_customers].sum(axis=0)

    # Recommend products not already purchased by the target customer
    products_already_bought = user_item_matrix.loc[customer_id][user_item_matrix.loc[customer_id] > 0].index
    recommendations = similar_customer_products.drop(products_already_bought).sort_values(ascending=False).head(n_recommendations)

    return recommendations

# Example: Recommend products for Customer ID 12345
recommendations = recommend_products(12567, user_item_matrix, customer_similarity_df)
print(recommendations)
```

```
Description
WORLD WAR 2 GLIDERS ASSTD DESIGNS     53215
WHITE HANGING HEART T-LIGHT HOLDER    34147
POPCORN HOLDER                        30504
PACK OF 12 LONDON TISSUES             25321
PACK OF 60 PINK PAISLEY CAKE CASES    24163
dtype: int64
```

7. Further Improvements

- **Optimization**: Use more advanced collaborative filtering techniques, like matrix factorization (e.g., SVD).
- **Hybrid Approach**: Combine content-based and collaborative filtering for better accuracy.

8. Final Steps

- Test the system with various customer IDs.
- Save the project code and results in a PDF format as per the submission requirements.

Once you've completed the coding part, generate a report summarizing your approach, key findings, and results.