task 2 : Lookalike Model

To create a Lookalike Model, we'll use a combination of customer and product information to calculate a similarity score between customers. This task can be achieved using techniques such as cosine similarity or distance-based metrics applied to feature vectors representing the customers.

1. Data Preprocessing:

Data Sources:

Customers.csv: Contains customer demographics like Region and SignupDate.

Products.csv: Includes product details such as ProductID, Category, and Price.

Transactions.csv: Tracks customer transactions, including TransactionID, TotalValue, and Quantity.

Data Cleaning: Dates were converted to datetime format, missing values were handled, and datasets were merged to create a unified table.

2. Feature Engineering:

Customer Attributes: Aggregated data to compute features such as:

Total Spending: Sum of all purchases for a customer.

Average Transaction Value: Average spending per transaction.

Transaction Count: Total number of transactions.

Product Preferences: Calculated the proportion of purchases in different product categories for each customer.

Region Encoding: One-hot encoding was used to represent the customer's region.

3. Feature Scaling:

Standardized all numerical features to ensure fair comparison between different units (e.g., spending and transaction count).

4. Similarity Calculation:

Cosine Similarity was used to compute pairwise similarity between customers.

For each customer, the top 3 most similar customers and their similarity scores were identified.

5. Recommendation Generation:

For the first 20 customers (CustomerID: C0001 - C0020), the top 3 similar customers and their scores were saved in a Lookalike.csv file.

got output as:

```
   cust_id                          similar_customers
0    C0001  [{'cust_id': 'C0120', 'score': 0.8560075913567...
1    C0002  [{'cust_id': 'C0178', 'score': 0.8981816126563...
2    C0003  [{'cust_id': 'C0031', 'score': 0.8652900814970...
3    C0004  [{'cust_id': 'C0012', 'score': 0.9310529501109...
4    C0005  [{'cust_id': 'C0007', 'score': 0.9201495226904...
5    C0006  [{'cust_id': 'C0187', 'score': 0.8652500812343...
6    C0007  [{'cust_id': 'C0005', 'score': 0.9201495226904...
7    C0008  [{'cust_id': 'C0109', 'score': 0.8329058602952...
8    C0009  [{'cust_id': 'C0198', 'score': 0.9706349815983...
9    C0010  [{'cust_id': 'C0111', 'score': 0.9037949324444...
10   C0011  [{'cust_id': 'C0107', 'score': 0.9065216391321...
11   C0012  [{'cust_id': 'C0004', 'score': 0.9310529501109...
12   C0013  [{'cust_id': 'C0099', 'score': 0.9375937585255...
13   C0014  [{'cust_id': 'C0060', 'score': 0.9799358106714...
14   C0015  [{'cust_id': 'C0058', 'score': 0.9362438180050...
15   C0016  [{'cust_id': 'C0117', 'score': 0.8937869271314...
16   C0017  [{'cust_id': 'C0075', 'score': 0.9449244161958...
17   C0018  [{'cust_id': 'C0068', 'score': 0.8578723357568...
18   C0019  [{'cust_id': 'C0121', 'score': 0.7853971798940...
19   C0020  [{'cust_id': 'C0050', 'score': 0.8506636429587...
```