In [1]:
```
import pandas as pd

customers = pd.read_csv("Customers.csv")
products = pd.read_csv("Products.csv")
transactions = pd.read_csv("Transactions.csv")
```

In [2]:
```
# Merge Transactions with Customers
customer_transactions = transactions.merge(customers, on="CustomerID")

# Merge again with Products to add category and price information
customer_data = customer_transactions.merge(products, on="ProductID")

# Display the first few rows to see the combined dataset
print(customer_data.head())
```

```
  TransactionID CustomerID ProductID      TransactionDate  Quantity  \
0        T00001      C0199      P067  2024-08-25 12:38:23         1
1        T00112      C0146      P067  2024-05-27 22:23:54         1
2        T00166      C0127      P067  2024-04-25 07:38:55         1
3        T00272      C0087      P067  2024-03-26 22:55:37         2
4        T00363      C0070      P067  2024-03-21 15:10:10         3

   TotalValue  Price_x     CustomerName         Region  SignupDate  \
0      300.68   300.68   Andrea Jenkins         Europe  2022-12-03
1      300.68   300.68  Brittany Harvey           Asia  2024-09-04
2      300.68   300.68  Kathryn Stevens         Europe  2024-04-04
3      601.36   300.68  Travis Campbell  South America  2024-04-11
4      902.04   300.68    Timothy Perez         Europe  2022-03-15

                      ProductName     Category  Price_y
0  ComfortLiving Bluetooth Speaker  Electronics   300.68
1  ComfortLiving Bluetooth Speaker  Electronics   300.68
2  ComfortLiving Bluetooth Speaker  Electronics   300.68
3  ComfortLiving Bluetooth Speaker  Electronics   300.68
4  ComfortLiving Bluetooth Speaker  Electronics   300.68
```

In [3]:
```
# Calculate total spending per customer
customer_spending = customer_data.groupby("CustomerID")["TotalValue"].sum()

# Count products purchased per category
category_counts = customer_data.pivot_table(
    index="CustomerID", columns="Category", values="Quantity", aggfunc="sum
)

# Add region information
region_info = customers.set_index("CustomerID")["Region"]

# Combine all into one DataFrame
customer_profile = pd.concat([customer_spending, category_counts, region_in
customer_profile.columns = ["TotalSpend"] + list(category_counts.columns) +

# Convert 'Region' into numbers (for similarity calculations)
customer_profile["Region"] = customer_profile["Region"].astype("category").

# Display the customer profile
print(customer_profile.head())
```

CustomerID	TotalSpend	Books	Clothing	Electronics	Home Decor	Region
C0001	3354.52	2.0	0.0	7.0	3.0	3
C0002	1862.74	0.0	4.0	0.0	6.0	0
C0003	2725.38	0.0	4.0	4.0	6.0	3
C0004	5354.88	8.0	0.0	6.0	9.0	3
C0005	2034.24	0.0	0.0	4.0	3.0	0

In [4]:
```
from sklearn.metrics.pairwise import cosine_similarity

# Calculate similarity scores between customers
similarity_matrix = cosine_similarity(customer_profile.fillna(0))

# Create a DataFrame for similarity matrix
similarity_df = pd.DataFrame(similarity_matrix, index=customer_profile.inde

# Function to get top 3 similar customers for a given customer
def get_top_3_similar(customer_id):
    similar_scores = similarity_df[customer_id].sort_values(ascending=False
    return list(zip(similar_scores.index, similar_scores.values))
```

In [5]:
```python
# Create a dictionary for the top 3 lookalikes
lookalike_recommendations = {
    customer_id: get_top_3_similar(customer_id) for customer_id in customer
}

# Convert to a DataFrame for saving
lookalike_df = pd.DataFrame(
    [
        {"CustomerID": cust, "Lookalikes": str(lookalike_recommendations[cu
        for cust in lookalike_recommendations
    ]
)

# Save as CSV
lookalike_df.to_csv("Rounak_Kumar_Lookalike.csv", index=False)
```

In [ ]: