

```

In [8]: # Step 1: Aggregate transaction data
customer_transactions = transactions.groupby('CustomerID').agg(
    TotalSpend=('TotalValue', 'sum'),
    TotalQuantity=('Quantity', 'sum'),
    TransactionCount=('TransactionID', 'count')
).reset_index()

# Step 2: Merge with customer profiles
customer_profiles = customers.merge(customer_transactions, on='CustomerID', how='left')

# One-hot encode the Region column
customer_profiles = pd.get_dummies(customer_profiles, columns=['Region'], drop_first=True)

# Drop unnecessary columns
customer_profiles_features = customer_profiles.drop(columns=['CustomerID', 'CustomerName'])

from sklearn.metrics.pairwise import cosine_similarity
import numpy as np

# Compute cosine similarity
similarity_matrix = cosine_similarity(customer_profiles_features)

# Create a DataFrame for similarity matrix
similarity_df = pd.DataFrame(similarity_matrix, index=customer_profiles['CustomerID'])

# Generate top 3 recommendations for the first 20 customers
lookalike_dict = {}

for cust_id in customer_profiles['CustomerID'][:20]:
    # Sort similarity scores in descending order, exclude the customer themselves
    similar_customers = similarity_df[cust_id].sort_values(ascending=False)[1:]
    lookalike_dict[cust_id] = list(zip(similar_customers.index, similar_customers.values))

# Convert the Lookalike dictionary into a DataFrame
lookalike_df = pd.DataFrame.from_dict(lookalike_dict, orient='index', columns=['LookalikeID', 'Similarity'])
lookalike_df.to_csv("Lookalike.csv")

print(lookalike_df.head())

```

	Top1	Top2 \
C0001	(C0011, 0.9999999840608153)	(C0012, 0.9999999811673553)
C0002	(C0034, 0.9999999147211551)	(C0043, 0.9999998706917115)
C0003	(C0136, 0.9999998983359859)	(C0190, 0.9999998793526346)
C0004	(C0195, 0.9999999946388577)	(C0039, 0.9999999944704254)
C0005	(C0145, 0.9999999957807079)	(C0173, 0.9999999952585269)

	Top3
C0001	(C0191, 0.9999999794383162)
C0002	(C0030, 0.9999998693464531)
C0003	(C0142, 0.9999998567679897)
C0004	(C0169, 0.9999999781031476)
C0005	(C0178, 0.9999999929629075)