Customer Segmentation for E-commerce & Retail 🛍️

**Rasesh Bhalala MSc Data Science **

```
import pandas as pd
import numpy as np
```

```
df = pd.read_excel("Online Retail.xlsx")
df.info()  # Check data types and missing values
df.dropna(inplace=True)  # Remove missing values
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 8 columns):
 #   Column       Non-Null Count   Dtype
---  ------       --------------   -----
 0   InvoiceNo    541909 non-null  object
 1   StockCode    541909 non-null  object
 2   Description  540455 non-null  object
 3   Quantity     541909 non-null  int64
 4   InvoiceDate  541909 non-null  datetime64[ns]
 5   UnitPrice    541909 non-null  float64
 6   CustomerID   406829 non-null  float64
 7   Country      541909 non-null  object
dtypes: datetime64[ns](1), float64(2), int64(1), object(4)
memory usage: 33.1+ MB
```

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

```
df['TotalPrice'] = df['Quantity'] * df['UnitPrice']
```

```
latest_date = df['InvoiceDate'].max()
```

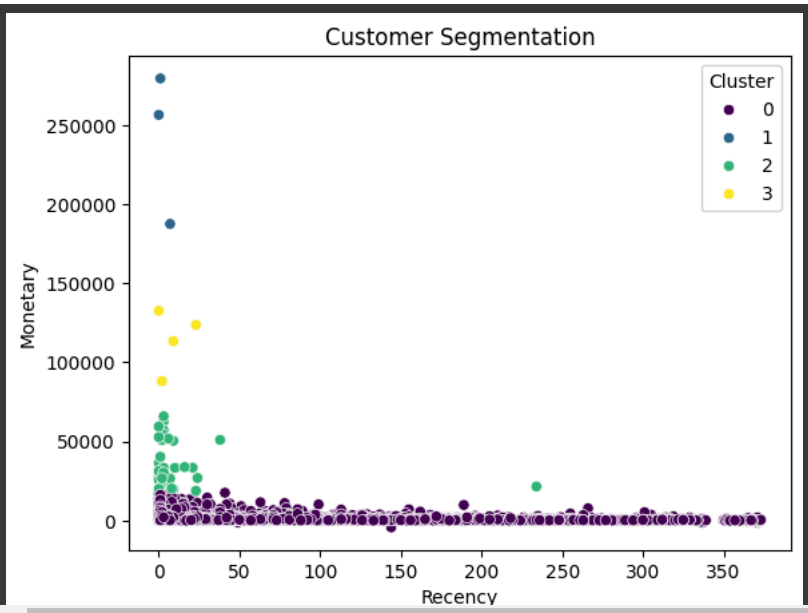
```
rfm_df = df.groupby('CustomerID').agg({
    'InvoiceDate': lambda x: (latest_date - x.max()).days,  # Recency
    'InvoiceNo': 'count',  # Frequency
    'TotalPrice': 'sum'  # Monetary
}).rename(columns={'InvoiceDate': 'Recency', 'InvoiceNo': 'Frequency', 'TotalPrice': 'Monetary'})
```

```
print(rfm_df.head())
```

```
            Recency  Frequency  Monetary
CustomerID
12346.0         325          2      0.00
12347.0           1        182   4310.00
12348.0          74         31   1797.24
12349.0          18         73   1757.55
12350.0         309         17    334.40
```

```
# Apply K-Means Clustering
kmeans = KMeans(n_clusters=4, random_state=42)
rfm_df['Cluster'] = kmeans.fit_predict(rfm_df)
```

```
sns.scatterplot(x='Recency', y='Monetary', hue='Cluster', data=rfm_df, palette='viridis')
plt.title("Customer Segmentation")
plt.show()
```




```
Start coding or generate with AI.
```

```
print("\nBusiness Insights & Applications:")
print(" 🔹 Loyal Customers: High frequency, high spending → Offer loyalty rewards.")
print(" 🔹 At-Risk Customers: High recency, low spending → Send discount offers.")
print(" 🔹 Potential Customers: Moderate frequency and spending → Encourage repeat purchases.")
```

```
Business Insights & Applications:
 🔹 Loyal Customers: High frequency, high spending → Offer loyalty rewards.
 🔹 At-Risk Customers: High recency, low spending → Send discount offers.
 🔹 Potential Customers: Moderate frequency and spending → Encourage repeat purchases.
```

```
Start coding or generate with AI.
```

```
Start coding or generate with AI.
```