

```
# Chip Purchase Behavior Analysis
```

```
# Date: April 2025
```

```
# Analyst: [Your Name]
```

```
# -----
```

```
# Step 1: Import Libraries and Upload CSVs
```

```
# -----
```

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from google.colab import files
import re
```

```
sns.set(style="whitegrid")
```

```
# Upload CSV files
```

```
uploaded = files.upload()
```

```
# -----
```

```
# Step 2: Load and Explore the Data
```

```
# -----
```

```
transactions = pd.read_csv('transactions.csv')
customers = pd.read_csv('customers.csv')
```

```
print("Transactions Sample:")
```

```
print(transactions.head())
```

```
print("\nCustomers Sample:")
```

```
print(customers.head())
```

```
# -----
```

```
# Step 3: Clean and Merge Data
```

```
# -----
```

```
transactions['DATE'] = pd.to_datetime(transactions['DATE'], dayfirst=True)
transactions.rename(columns={'PROD_QTY': 'Quantity'}, inplace=True)
merged = pd.merge(transactions, customers, on='LYLTY_CARD_NBR', how='inner')
merged['Segment'] = merged['LIFESTAGE'] + ' - ' + merged['PREMIUM_CUSTOMER']
```

```
# -----
```

```
# Step 4: Feature Engineering - Pack Size & Brand
```

```
# -----
```

```
merged['Pack_Size'] = merged['PROD_NAME'].str.extract(r'(\d+)[gG]').astype(float)
merged['Brand'] = merged['PROD_NAME'].str.split().str[0]
```

```
# -----
```

```
# Step 5: Segment-Level Sales Analysis
```

```
# -----
```

```
segment_sales =
merged.groupby('Segment')['TOT_SALES'].sum().sort_values(ascending=False).reset_index()
```

```
plt.figure(figsize=(12,6))
```

```
sns.barplot(data=segment_sales, x='TOT_SALES', y='Segment', palette='Blues_d')
```

```
plt.title('Total Sales by Customer Segment')
```

```
plt.xlabel('Total Sales ($)')
```

```
plt.ylabel('Customer Segment')
```

```
plt.tight_layout()
plt.show()
```

```
# -----
```

```
# Step 6: Top-Selling Products for Best Segment
```

```
# -----
```

```
best_segment_name = segment_sales.iloc[0]['Segment']
best_segment = merged[merged['Segment'] == best_segment_name]
```

```
top_products =
best_segment.groupby('PROD_NAME')['TOT_SALES'].sum().sort_values(ascending=False).head(10).reset_index()
```

```
plt.figure(figsize=(10,6))
sns.barplot(data=top_products, x='TOT_SALES', y='PROD_NAME', palette='magma')
plt.title(f"Top Products for {best_segment_name}")
plt.xlabel('Total Sales')
plt.ylabel('Product')
plt.tight_layout()
plt.show()
```

```
# -----
```

```
# Step 7: Monthly Sales Trend by Segment
```

```
# -----
```

```
merged['MONTH'] = merged['DATE'].dt.to_period('M').dt.to_timestamp()
monthly_sales = merged.groupby(['MONTH', 'Segment'])['TOT_SALES'].sum().reset_index()
```

```
plt.figure(figsize=(14,7))
sns.lineplot(data=monthly_sales, x='MONTH', y='TOT_SALES', hue='Segment', marker='o')
plt.title('Monthly Sales Trend by Segment')
plt.xlabel('Month')
plt.ylabel('Total Sales ($)')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

```
# -----
```

```
# Step 8: Average Spend per Transaction by Segment
```

```
# -----
```

```
avg_txn = merged.groupby(['TXN_ID', 'Segment'])['TOT_SALES'].sum().reset_index()
avg_per_segment =
avg_txn.groupby('Segment')['TOT_SALES'].mean().reset_index().sort_values(by='TOT_SALES',
ascending=False)
```

```
plt.figure(figsize=(12,6))
sns.barplot(data=avg_per_segment, y='Segment', x='TOT_SALES', palette='viridis')
plt.title('Avg Spend per Transaction by Segment')
plt.xlabel('Avg Spend ($)')
plt.ylabel('Segment')
plt.tight_layout()
plt.show()
```

```
# -----
```

```
# Step 9: Avg Pack Size by Segment
```

```
# -----
```

```

avg_pack = merged.groupby('Segment')['Pack_Size'].mean().reset_index().sort_values(by='Pack_Size',
ascending=False)

plt.figure(figsize=(12,6))
sns.barplot(data=avg_pack, x='Pack_Size', y='Segment', palette='cubehelix')
plt.title('Average Pack Size by Segment')
plt.xlabel('Avg Pack Size (g)')
plt.ylabel('Segment')
plt.tight_layout()
plt.show()

# -----
#   Step 10: Top Brands by Segment
# -----
top_brands = merged.groupby(['Segment', 'Brand'])['TOT_SALES'].sum().reset_index()
top_brands = top_brands.sort_values(['Segment', 'TOT_SALES'], ascending=[True, False])
top_5_brands = top_brands.groupby('Segment').head(5)

# Just preview the top 5 brands per segment
print("\nTop 5 Brands per Segment:")
print(top_5_brands)

# -----
#   Step 11: Strategic Recommendations
# -----
print("\nRecommendations:")
print("- Focus marketing on high-spending segments like 'Older Families - Budget'")
print("- Promote bulk/value chip products in top-performing stores")
print("- Schedule promotions around mid-year seasonal sales spikes")
print("- Consider larger pack sizes for older families and retirees")
print("- Prioritize high-performing brands in target segments")

```