

```
In [1]: import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

```
In [2]: qvi_data = pd.read_csv("QVI_data.csv")

qvi_data['DATE'] = pd.to_datetime(qvi_data['DATE'])

qvi_data['MONTH'] = qvi_data['DATE'].dt.to_period('M')

print(qvi_data.info())
print(qvi_data.head())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 264834 entries, 0 to 264833
Data columns (total 13 columns):
#   Column                Non-Null Count  Dtype
---  -
0   LYLTY_CARD_NBR        264834 non-null int64
1   DATE                  264834 non-null datetime64[ns]
2   STORE_NBR             264834 non-null int64
3   TXN_ID                264834 non-null int64
4   PROD_NBR              264834 non-null int64
5   PROD_NAME             264834 non-null object
6   PROD_QTY              264834 non-null int64
7   TOT_SALES             264834 non-null float64
8   PACK_SIZE            264834 non-null int64
9   BRAND                 264834 non-null object
10  LIFESTAGE             264834 non-null object
11  PREMIUM_CUSTOMER      264834 non-null object
12  MONTH                 264834 non-null period[M]
dtypes: datetime64[ns](1), float64(1), int64(6), object(4), period[M](1)
memory usage: 26.3+ MB
None
```

	LYLTY_CARD_NBR	DATE	STORE_NBR	TXN_ID	PROD_NBR	\
0	1000	2018-10-17	1	1	5	
1	1002	2018-09-16	1	2	58	
2	1003	2019-03-07	1	3	52	
3	1003	2019-03-08	1	4	106	
4	1004	2018-11-02	1	5	96	

	PROD_NAME	PROD_QTY	TOT_SALES	PACK_SIZE	\
0	Natural Chip Compny SeaSalt	175g	2	6.0	175
1	Red Rock Deli Chikn&Garlic Aioli	150g	1	2.7	150
2	Grain Waves Sour Cream&Chives	210G	1	3.6	210
3	Natural ChipCo Hony Soy Chckn	175g	1	3.0	175
4	WW Original Stacked Chips	160g	1	1.9	160

	BRAND	LIFESTAGE	PREMIUM_CUSTOMER	MONTH
0	NATURAL	YOUNG SINGLES/COUPLES	Premium	2018-10
1	RRD	YOUNG SINGLES/COUPLES	Mainstream	2018-09
2	GRNWVES	YOUNG FAMILIES	Budget	2019-03
3	NATURAL	YOUNG FAMILIES	Budget	2019-03
4	WOOLWORTHS	OLDER SINGLES/COUPLES	Mainstream	2018-11

```
In [3]: monthly_metrics = qvi_data.groupby(['STORE_NBR', 'MONTH']).agg(
        total_sales=('TOT_SALES', 'sum'),
        total_customers=('LYLTY_CARD_NBR', pd.Series.nunique),
        total_transactions=('TXN_ID', pd.Series.nunique)
    ).reset_index()

monthly_metrics['transactions_per_customer'] = (
    monthly_metrics['total_transactions'] / monthly_metrics['total_customers']
)

monthly_metrics.head()
```

```
Out[3]:
```

	STORE_NBR	MONTH	total_sales	total_customers	total_transactions	transactions_per
0	1	2018-07	206.9	49	52	
1	1	2018-08	176.1	42	43	
2	1	2018-09	278.8	59	62	
3	1	2018-10	188.1	44	45	
4	1	2018-11	192.6	46	47	

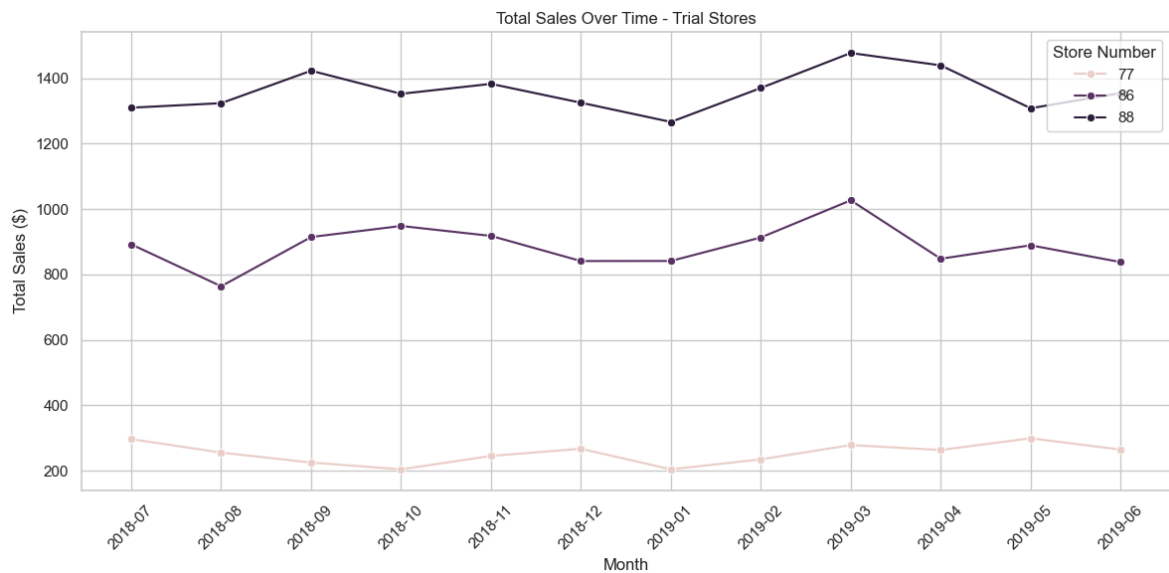
```
In [5]: trial_data['MONTH'] = trial_data['MONTH'].astype(str)
```

C:\Users\ubait\AppData\Local\Temp\ipykernel\_7652\4135628852.py:1: SettingWithCopyWarning:  
Warning:  
A value is trying to be set on a copy of a slice from a DataFrame.  
Try using .loc[row\_indexer,col\_indexer] = value instead  
  
See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)  
trial\_data['MONTH'] = trial\_data['MONTH'].astype(str)

```
In [6]: # Convert MONTH to string for plotting
trial_data['MONTH'] = trial_data['MONTH'].astype(str)

# Total Sales Over Time
plt.figure(figsize=(12, 6))
sns.lineplot(data=trial_data, x='MONTH', y='total_sales', hue='STORE_NBR', marker='o')
plt.title('Total Sales Over Time - Trial Stores')
plt.xlabel('Month')
plt.ylabel('Total Sales ($)')
plt.xticks(rotation=45)
plt.legend(title='Store Number')
plt.tight_layout()
plt.show()
```

C:\Users\ubait\AppData\Local\Temp\ipykernel\_7652\1968235051.py:2: SettingWithCopyWarning:  
Warning:  
A value is trying to be set on a copy of a slice from a DataFrame.  
Try using .loc[row\_indexer,col\_indexer] = value instead  
  
See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)  
trial\_data['MONTH'] = trial\_data['MONTH'].astype(str)



```
In [9]: # Ensure MONTH is string for plotting
trial_data['MONTH'] = trial_data['MONTH'].astype(str)

# Plot: Total Customers Over Time
plt.figure(figsize=(12, 6))
sns.lineplot(data=trial_data, x='MONTH', y='total_customers', hue='STORE_NBR', m
plt.title('Total Customers Over Time - Trial Stores')
plt.xlabel('Month')
plt.ylabel('Number of Customers')
plt.xticks(rotation=45)
plt.legend(title='Store Number')
plt.tight_layout()
plt.show()
```

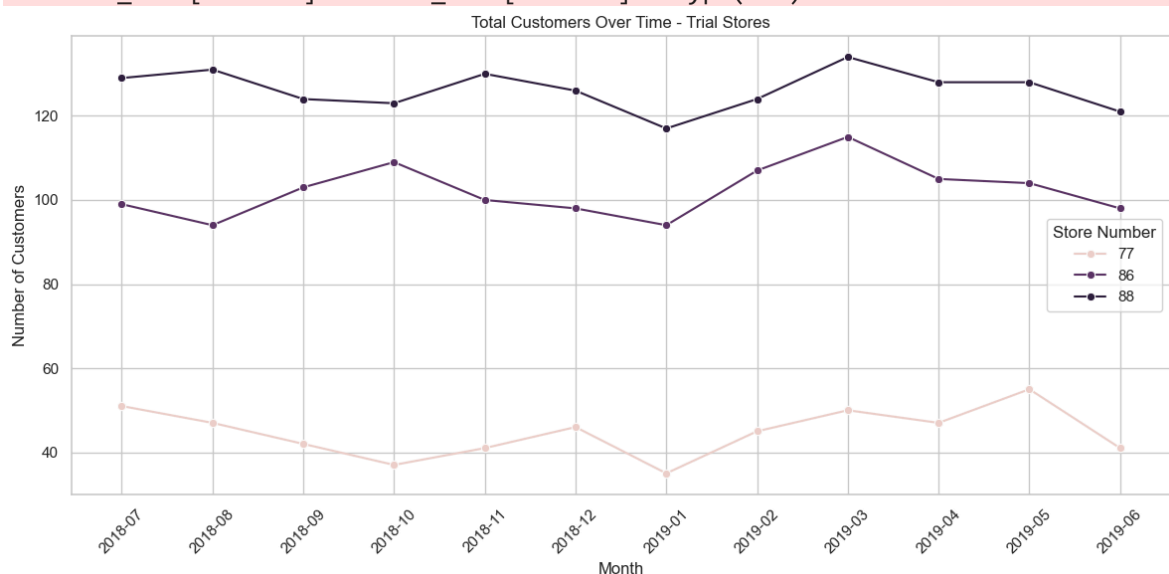
C:\Users\ubait\AppData\Local\Temp\ipykernel\_7652\3771779990.py:2: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.

Try using .loc[row\_indexer,col\_indexer] = value instead

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
trial_data['MONTH'] = trial_data['MONTH'].astype(str)
```



```
In [10]: # Plot: Transactions per Customer Over Time
plt.figure(figsize=(12, 6))
sns.lineplot(data=trial_data, x='MONTH', y='transactions_per_customer', hue='STC')
plt.title('Transactions per Customer Over Time - Trial Stores')
plt.xlabel('Month')
plt.ylabel('Transactions per Customer')
plt.xticks(rotation=45)
plt.legend(title='Store Number')
plt.tight_layout()
plt.show()
```

