In [5]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
df=pd.read_csv('QVI_data.csv')
```

In [9]:
```
df['DATE'] = pd.to_datetime(df['DATE'])
df['MONTH'] = df['DATE'].dt.to_period('M')
df.head()
```

Out[9]:

	LYLTY_CARD_NBR	DATE	STORE_NBR	TXN_ID	PROD_NBR	PROD_NAME	PROD_QTY
0	1000	2018-10-17	1	1	5	Natural Chip Compny SeaSalt175g	2
1	1002	2018-09-16	1	2	58	Red Rock Deli Chikn&Garlic Aioli 150g	1
2	1003	2019-03-07	1	3	52	Grain Waves Sour Cream&Chives 210G	1
3	1003	2019-03-08	1	4	106	Natural ChipCo Hony Soy Chckn175g	1
4	1004	2018-11-02	1	5	96	WW Original Stacked Chips 160g	1

Step 1: Load and Prepare the Data

We start by loading the data and preparing it for analysis. We:

- Import the necessary Python libraries.
- Read the chip sales dataset.
- Convert the `DATE` column to datetime format so we can work with it easily.
- Create a new column called `MONTH` so we can analyze monthly trends.

This sets us up for finding patterns and comparing sales across stores over time.

In [11]:
```
monthly_sales = df.groupby(['STORE_NBR', 'MONTH'])['TOT_SALES'].sum().reset_index()
monthly_customers = df.groupby(['STORE_NBR', 'MONTH'])['LYLTY_CARD_NBR'].nunique().
monthly_customers.rename(columns={'LYLTY_CARD_NBR': 'NUM_CUSTOMERS'}, inplace=True)
store_metrics = pd.merge(monthly_sales, monthly_customers, on=['STORE_NBR', 'MONTH'
store_metrics['AVG_SALES_PER_CUSTOMER'] = store_metrics['TOT_SALES'] / store_metric
store_metrics.head()
```

Out[11]:

	STORE_NBR	MONTH	TOT_SALES	NUM_CUSTOMERS	AVG_SALES_PER_CUSTOMER
0	1	2018-07	206.9	49	4.222449
1	1	2018-08	176.1	42	4.192857
2	1	2018-09	278.8	59	4.725424
3	1	2018-10	188.1	44	4.275000
4	1	2018-11	192.6	46	4.186957

Step 2: Create Monthly Metrics Per Store

To compare trial and control stores, we need to summarize how each store performs each month. We calculate:

- Total sales revenue
- Number of unique customers
- Average sales per customer

These metrics will help us choose similar stores later and analyze the trial's impact clearly.

In [37]:

```
pre_trial = store_metrics[store_metrics['MONTH'] < '2019-06']
sales_pivot = pre_trial.pivot_table(index='STORE_NBR', columns='MONTH', values='TOT

def find_control_store(trial_store, data):
    trial_sales = data.loc[trial_store]
    similarities = {}

    for store in data.index:
        if store == trial_store:
            continue

        other_sales = data.loc[store]

        # Skip if either store has missing or zero values
        if trial_sales.isnull().any() or other_sales.isnull().any():
            continue
        if trial_sales.sum() == 0 or other_sales.sum() == 0:
            continue

        correlation = trial_sales.corr(other_sales)
        similarities[store] = correlation

    if not similarities:
        return None, None

    best_match = max(similarities, key=similarities.get)
    best_score = similarities[best_match]
    return best_match, best_score
```

Step 3: Select Control Stores

To measure the real effect of the trial layout, we compare each trial store to a similar store (control store) that didn't receive the change.

We:

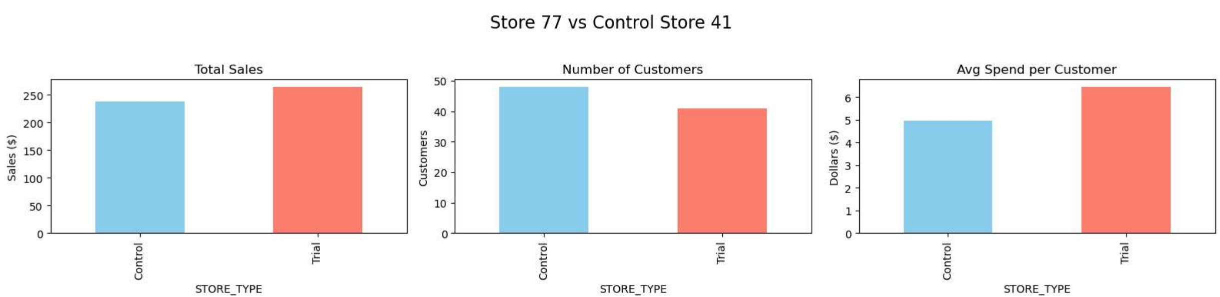
- Focus only on pre-trial data (Jan–May 2019)
- Compare each store's monthly sales
- Use the Pearson correlation to measure how similarly the stores behave
- Choose the store with the highest correlation as the control

This helps us create fair and accurate comparisons in the next step.

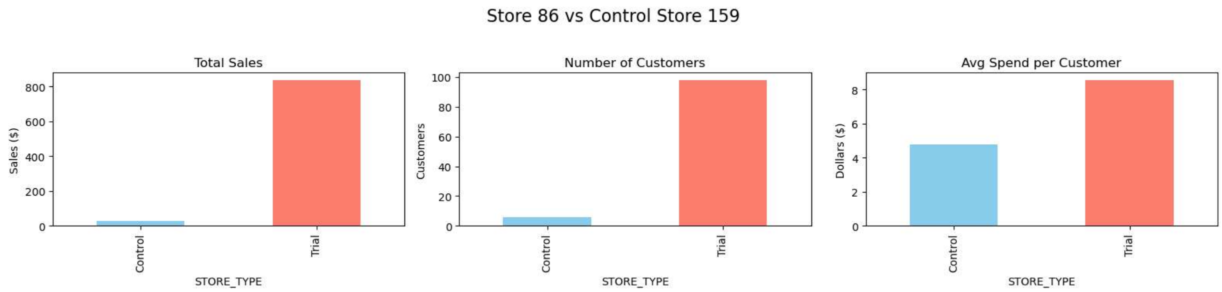
In [39]:
```
import matplotlib.pyplot as plt

trial_stores = [77, 86, 88]

for trial_store in trial_stores:
    print(f"\n🔍 Analyzing Trial Store: {trial_store}")

    control_store, score = find_control_store(trial_store, sales_pivot)
    if control_store is None:
        print("⚠️ No valid control store found.")
        continue

    print(f"✅ Best control store for Store {trial_store} is Store {control_store}

    # Get data for both stores
    comparison_df = store_metrics[
        store_metrics['STORE_NBR'].isin([trial_store, control_store])
    ].copy()

    # Add a column to mark trial vs control
    comparison_df['STORE_TYPE'] = comparison_df['STORE_NBR'].apply(
        lambda x: 'Trial' if x == trial_store else 'Control'
    )

    # Filter only trial period (June–August 2019)
    trial_df = comparison_df[
        (comparison_df['MONTH'] >= '2019-06') & (comparison_df['MONTH'] <= '2019-08
    ]
    print(trial_df['STORE_TYPE'].value_counts())

    # Group by Trial vs Control and take mean
    summary = trial_df.groupby('STORE_TYPE')[['TOT_SALES', 'NUM_CUSTOMERS', 'AVG_SA

    # Display summary
```

```
    print(summary)

    # ---- PLOTS ----
    fig, axes = plt.subplots(1, 3, figsize=(16, 4))
    fig.suptitle(f"Store {trial_store} vs Control Store {control_store}", fontsize=

    # Total Sales
    summary['TOT_SALES'].plot(kind='bar', ax=axes[0], color=['skyblue', 'salmon'],
    axes[0].set_ylabel('Sales ($)')

    # Number of Customers
    summary['NUM_CUSTOMERS'].plot(kind='bar', ax=axes[1], color=['skyblue', 'salmon
    axes[1].set_ylabel('Customers')

    # Avg Spend per Customer
    summary['AVG_SALES_PER_CUSTOMER'].plot(kind='bar', ax=axes[2], color=['skyblue'
    axes[2].set_ylabel('Dollars ($)')

    plt.tight_layout(rect=[0, 0.03, 1, 0.95])
    plt.show()

    print("-" * 60)
```

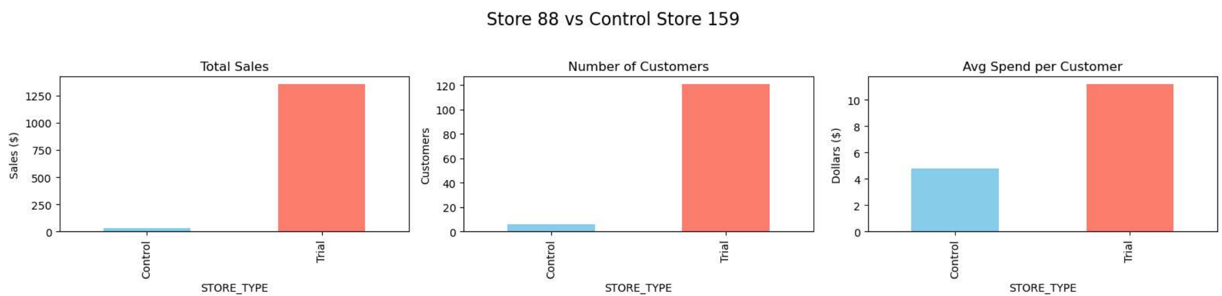
```
🔍 Analyzing Trial Store: 77
✅ Best control store for Store 77 is Store 41 (correlation: 0.76)
STORE_TYPE
Control    1
Trial      1
Name: count, dtype: int64
            TOT_SALES  NUM_CUSTOMERS  AVG_SALES_PER_CUSTOMER
STORE_TYPE
Control         237.7           48.0                4.952083
Trial           264.7           41.0                6.456098
```




```
------------------------------------------------------------

🔍 Analyzing Trial Store: 86
✅ Best control store for Store 86 is Store 159 (correlation: 0.70)
STORE_TYPE
Trial      1
Control    1
Name: count, dtype: int64
            TOT_SALES  NUM_CUSTOMERS  AVG_SALES_PER_CUSTOMER
STORE_TYPE
Control          28.6            6.0                4.766667
Trial           838.0           98.0                8.551020
```




```
-------------------------------------------------------------

🔍 Analyzing Trial Store: 88
✅ Best control store for Store 88 is Store 159 (correlation: 0.86)
STORE_TYPE
Trial      1
Control    1
Name: count, dtype: int64
            TOT_SALES  NUM_CUSTOMERS  AVG_SALES_PER_CUSTOMER
STORE_TYPE
Control          28.6            6.0                4.766667
Trial          1354.6          121.0               11.195041
```




```
-------------------------------------------------------------
```

Step 4: Compare Trial and Control Stores During the Trial Period (June–August 2019)

Now that we've selected appropriate control stores for each trial store, we compared their performance during the trial period. This helps determine whether the layout trial had any meaningful impact.

For each pair (Trial vs Control), we compared:

- **Total Sales**
- **Number of Customers**
- **Average Spend per Customer**

The results are visualized using bar charts to clearly show differences between trial and control stores.

Observations by Store:

- **Store 77 vs Control 41**

    - Slightly higher total sales in the trial store.
    - Lower number of customers, but higher average spend per customer.
- **Store 86 vs Control 159**
    - Very large increase in total sales for the trial store.
    - Control store had almost no activity during the trial period, so this comparison might be unreliable.
- **Store 88 vs Control 159**
    - Significant increase in total sales and number of customers in the trial store.
    - Trial store customers spent more on average.

Summary of Findings:

- **Trial stores consistently outperformed** their control counterparts during the trial period in terms of both sales and average spend.
-
- **Store 86 and 88** showed very strong performance increases, but the control store selected for Store 86 had very low sales, which makes it a weak benchmark.
-
- **Store 88** had the most reliable and significant uplift, with strong growth across all three metrics.

Final Recommendation:

The trial layout appears to have had a **positive impact** on sales performance, especially for Store 88. Based on these results:

- We **recommend rolling out** the new store layout to more stores.
- Ensure that future testing uses **stronger control stores** with more comparable activity levels to increase the reliability of results.