Importing Libraries

In [1]:
```
import pandas as pd
import numpy as np
import seaborn as sns
```

In [4]:
```
#importing dataset
file_path = "C:/quantium/"
transaction_data = pd.read_excel(file_path + "QVI_transaction_data.xlsx")
```

In [3]:
```
transaction_data.head()
```

Out[3]:

	DATE	STORE_NBR	LYLTY_CARD_NBR	TXN_ID	PROD_NBR	PROD_NAME	PROD_Q
0	2018-10-17	1	1000	1	5	Natural Chip Compny SeaSalt175g	
1	2019-05-14	1	1307	348	66	CCs Nacho Cheese 175g	
2	2019-05-20	1	1343	383	61	Smiths Crinkle Cut Chips Chicken 170g	
3	2018-08-17	2	2373	974	69	Smiths Chip Thinly S/Cream&Onion 175g	
4	2018-08-18	2	2426	1038	108	Kettle Tortilla ChpsHny&Jlpno Chili 150g	

In [5]:
```
# now the second dataset regarding customer
customer_data = pd.read_csv(file_path + "QVI_purchase_behaviour.csv")
```

In [6]:
```
customer_data.head()
```

Out[6]:

	LYLTY_CARD_NBR	LIFESTAGE	PREMIUM_CUSTOMER
0	1000	YOUNG SINGLES/COUPLES	Premium
1	1002	YOUNG SINGLES/COUPLES	Mainstream
2	1003	YOUNG FAMILIES	Budget
3	1004	OLDER SINGLES/COUPLES	Mainstream
4	1005	MIDAGE SINGLES/COUPLES	Mainstream

Now Summarizing Dataset

In [7]:
```
transaction_data.describe()
```

Out[7]:

	DATE	STORE_NBR	LYLTY_CARD_NBR	TXN_ID	PROD_NBR
count	264836	264836.00000	2.648360e+05	2.648360e+05	264836.000000
mean	2018-12-30 00:52:12.879215616	135.08011	1.355495e+05	1.351583e+05	56.583157
min	2018-07-01 00:00:00	1.00000	1.000000e+03	1.000000e+00	1.000000
25%	2018-09-30 00:00:00	70.00000	7.002100e+04	6.760150e+04	28.000000
50%	2018-12-30 00:00:00	130.00000	1.303575e+05	1.351375e+05	56.000000
75%	2019-03-31 00:00:00	203.00000	2.030942e+05	2.027012e+05	85.000000
max	2019-06-30 00:00:00	272.00000	2.373711e+06	2.415841e+06	114.000000
std	NaN	76.78418	8.057998e+04	7.813303e+04	32.826638

Checking the null cell

In [8]:
```
transaction_data.isnull().sum()
```

Out[8]:
```
DATE              0
STORE_NBR         0
LYLTY_CARD_NBR    0
TXN_ID            0
PROD_NBR          0
PROD_NAME         0
PROD_QTY          0
TOT_SALES         0
dtype: int64
```

In [9]:
```
data_types = transaction_data.dtypes
print(data_types)
```

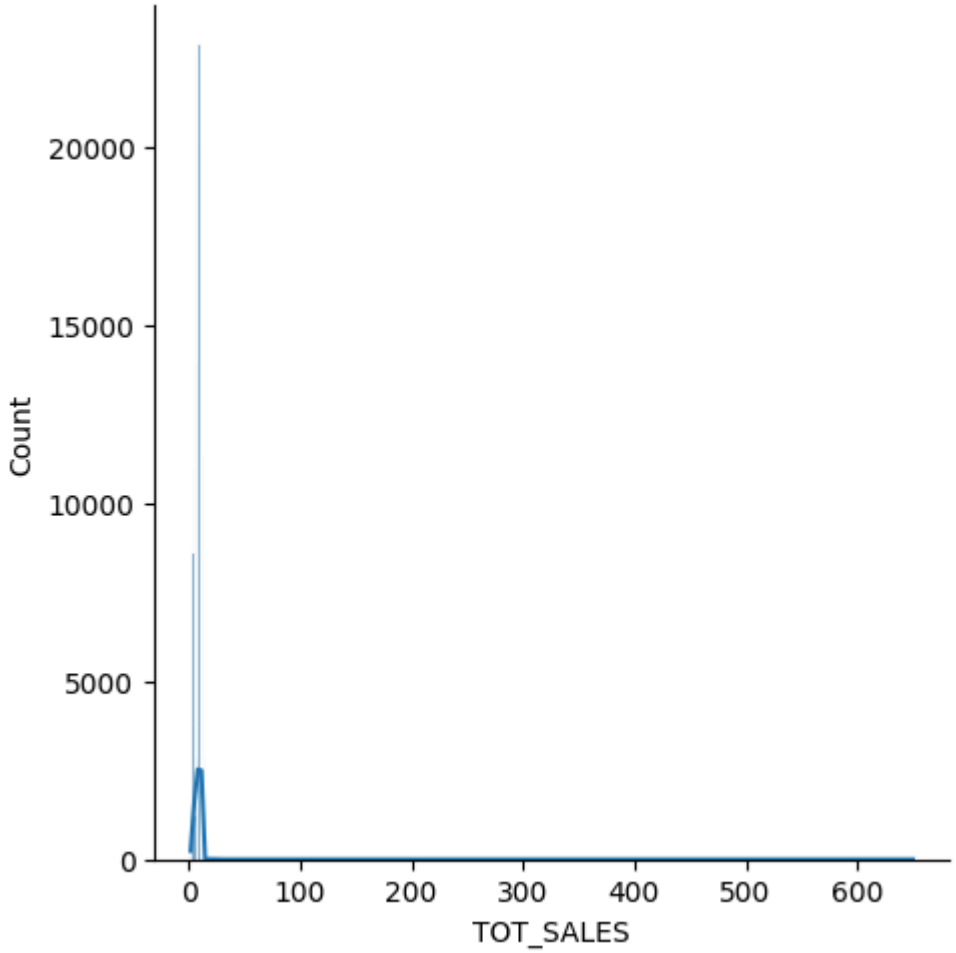
```
DATE              datetime64[ns]
STORE_NBR                  int64
LYLTY_CARD_NBR             int64
TXN_ID                     int64
PROD_NBR                   int64
PROD_NAME                 object
PROD_QTY                   int64
TOT_SALES                float64
dtype: object
```

Now we are going to examine the outliers

In [10]:
```
import matplotlib.pyplot as plt
import seaborn as sns
```

In [11]:
```
sns.displot(transaction_data.TOT_SALES, kde = True)
```

Out[11]:
```
<seaborn.axisgrid.FacetGrid at 0x1bfe71c70e0>
```



Now lets check the mean value of Total sales

In [12]:
```
numericdata = transaction_data.select_dtypes(['float','int'])
numericdata.head()
```

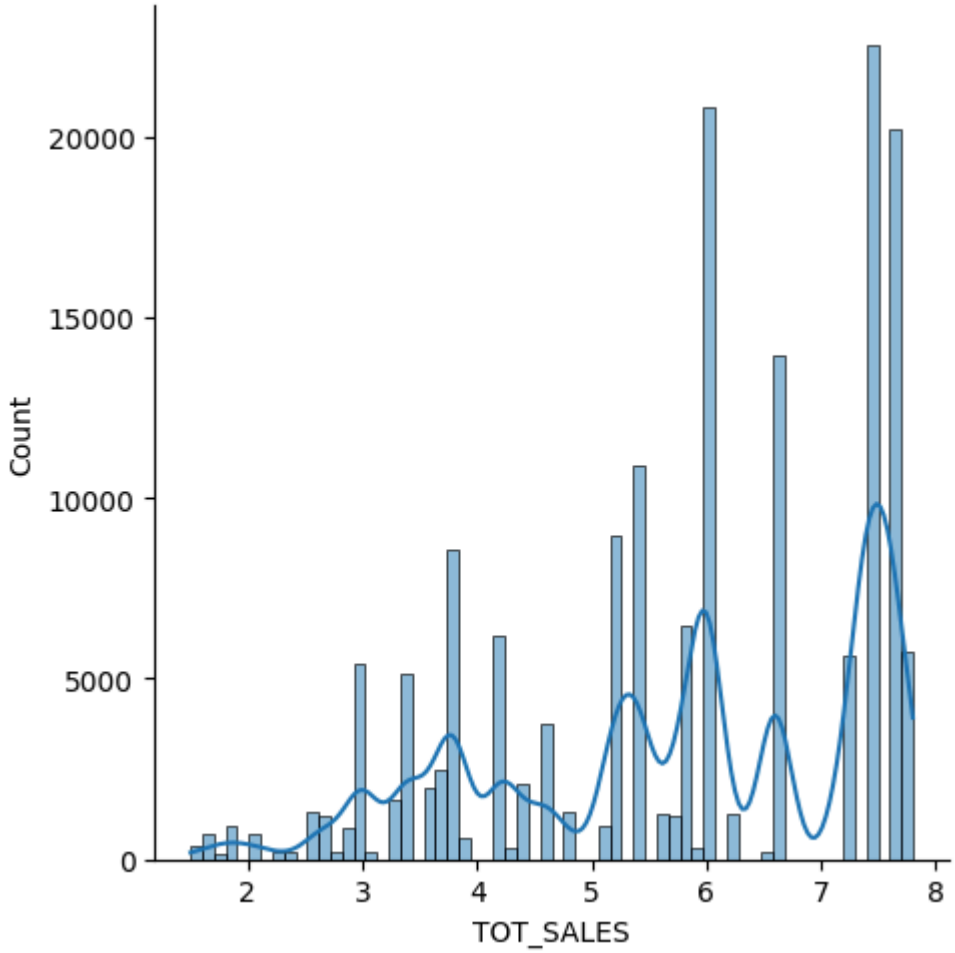
Out[12]:

	STORE_NBR	LYLTY_CARD_NBR	TXN_ID	PROD_NBR	PROD_QTY	TOT_SALES
0	1	1000	1	5	2	6.0
1	1	1307	348	66	3	6.3
2	1	1343	383	61	2	2.9
3	2	2373	974	69	5	15.0
4	2	2426	1038	108	3	13.8

In [13]:
```
x = numericdata[numericdata['TOT_SALES']< 8.000]
```

In [14]:
```
sns.displot(x.TOT_SALES, kde = True)
```

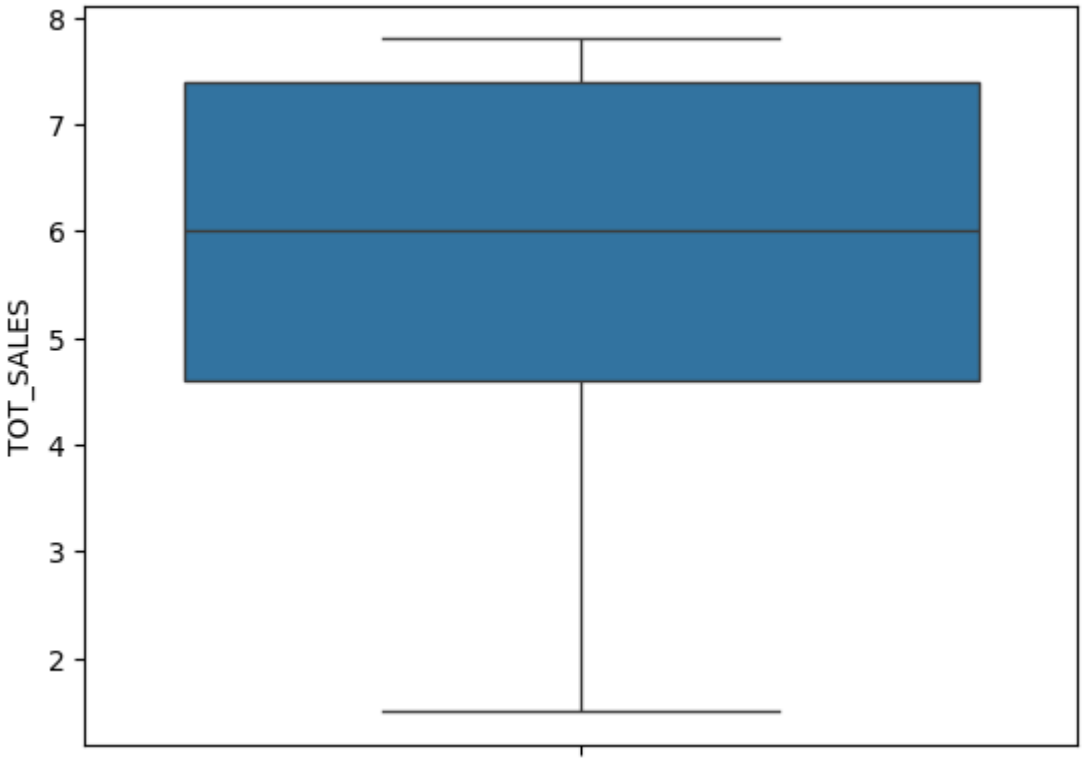
Out[14]:
```
<seaborn.axisgrid.FacetGrid at 0x1bfe7dac550>
```



now we can check in boxplot too

In [15]:
```
sns.boxplot(x.TOT_SALES)
```

Out[15]:
```
<Axes: ylabel='TOT_SALES'>
```



In [7]:
```
import pandas as pd

file_path = "C:/quantium/"
transaction_data = pd.read_excel(file_path + "QVI_transaction_data.xlsx")
```

In [8]:
```
# Filter out bulk purchases (possible promo or error)
transaction_data = transaction_data[transaction_data['PROD_QTY'] == 1]
```

In [9]:
```
transaction_data.head()
```

Out[9]:

	DATE	STORE_NBR	LYLTY_CARD_NBR	TXN_ID	PROD_NBR	PROD_NAME	PROD_QTY
5	2019-05-19	4	4074	2982	57	Old El Paso Salsa Dip Tomato Mild 300g	
6	2019-05-16	4	4149	3333	16	Smiths Crinkle Chips Salt & Vinegar 330g	
7	2019-05-16	4	4196	3539	24	Grain Waves Sweet Chilli 210g	
8	2018-08-20	5	5026	4525	42	Doritos Corn Chip Mexican Jalapeno 150g	
10	2019-05-17	7	7215	7176	16	Smiths Crinkle Chips Salt & Vinegar 330g	

Merge on LYLTY_CARD_NBR to add customer segments

In [11]:
```
customer_data = pd.read_csv(file_path + "QVI_purchase_behaviour.csv")
```

In [12]:
```
merged_data = pd.merge(transaction_data, customer_data, on='LYLTY_CARD_NBR', how='
```

In [14]:
```
merged_data.head()
```

Out[14]:

	DATE	STORE_NBR	LYLTY_CARD_NBR	TXN_ID	PROD_NBR	PROD_NAME	PROD_QTY
0	2019-05-19	4	4074	2982	57	Old El Paso Salsa Dip Tomato Mild 300g	1
1	2019-05-16	4	4149	3333	16	Smiths Crinkle Chips Salt & Vinegar 330g	1
2	2019-05-16	4	4196	3539	24	Grain Waves Sweet Chilli 210g	1
3	2018-08-20	5	5026	4525	42	Doritos Corn Chip Mexican Jalapeno 150g	1
4	2019-05-17	7	7215	7176	16	Smiths Crinkle Chips Salt & Vinegar 330g	1

Total, mean, and count of sales by customer segment

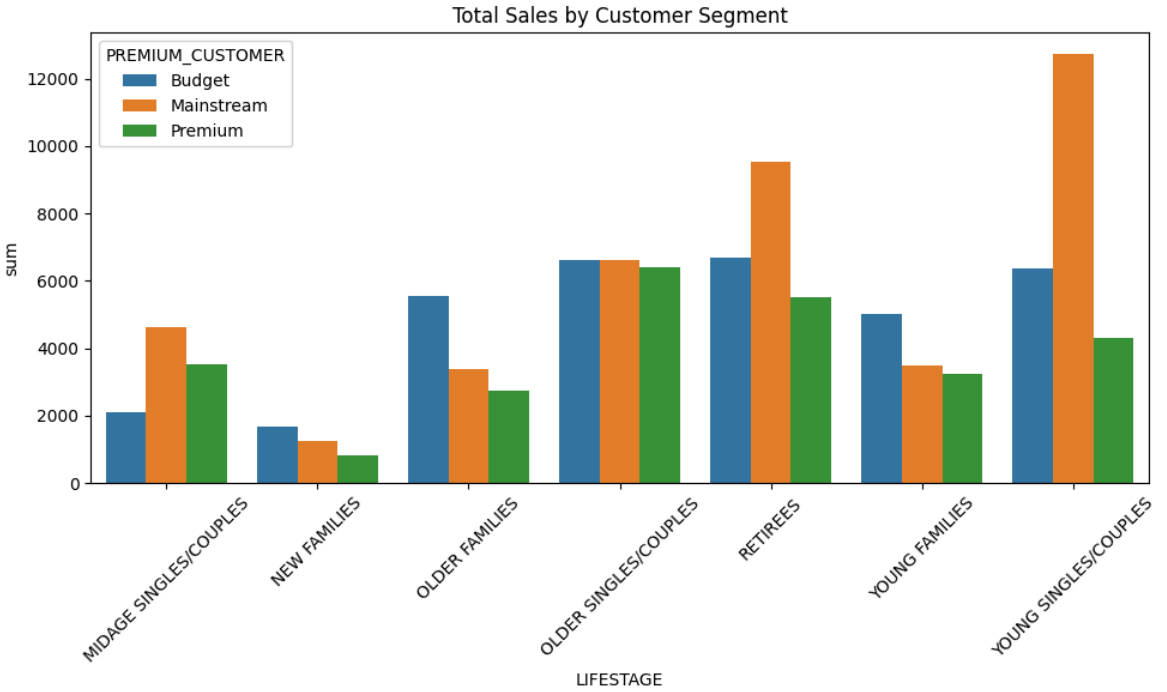
In [15]:
```
segment_analysis = merged_data.groupby(['LIFESTAGE', 'PREMIUM_CUSTOMER'])['TOT_SAL
print(segment_analysis)
```

```
                LIFESTAGE PREMIUM_CUSTOMER      sum      mean  count
0   MIDAGE SINGLES/COUPLES           Budget   2096.90  3.524202    595
1   MIDAGE SINGLES/COUPLES       Mainstream   4634.30  4.001986   1158
2   MIDAGE SINGLES/COUPLES          Premium   3518.65  3.575864    984
3            NEW FAMILIES           Budget   1687.15  3.708022    455
4            NEW FAMILIES       Mainstream   1248.60  3.629651    344
5            NEW FAMILIES          Premium    835.90  3.698673    226
6          OLDER FAMILIES           Budget   5552.45  3.655332   1519
7          OLDER FAMILIES       Mainstream   3376.70  3.727042    906
8          OLDER FAMILIES          Premium   2752.80  3.646093    755
9    OLDER SINGLES/COUPLES           Budget   6610.65  3.768900   1754
10   OLDER SINGLES/COUPLES       Mainstream   6608.95  3.667564   1802
11   OLDER SINGLES/COUPLES          Premium   6422.05  3.784355   1697
12                RETIREES           Budget   6701.00  3.790158   1768
13                RETIREES       Mainstream   9529.15  3.673535   2594
14                RETIREES          Premium   5528.35  3.865979   1430
15         YOUNG FAMILIES           Budget   5004.25  3.695901   1354
16         YOUNG FAMILIES       Mainstream   3489.45  3.704299    942
17         YOUNG FAMILIES          Premium   3223.50  3.779015    853
18   YOUNG SINGLES/COUPLES           Budget   6360.90  3.360222   1893
19   YOUNG SINGLES/COUPLES       Mainstream  12728.90  3.966625   3209
20   YOUNG SINGLES/COUPLES          Premium   4309.40  3.366719   1280
```

In [16]:
```
import seaborn as sns
import matplotlib.pyplot as plt
```

Barplot: Total sales by customer segment

In [18]:
```
plt.figure(figsize=(10,6))
sns.barplot(data=segment_analysis, x='LIFESTAGE', y='sum', hue='PREMIUM_CUSTOMER')
plt.title('Total Sales by Customer Segment')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



In [ ]: