In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
from IPython import get_ipython
import warnings
warnings.filterwarnings("ignore")
```

In [2]:

```
data = pd.read_csv('sales.csv')
```

In [3]:

```
data.head()
```

Out[3]:

	Store ID	Store_Area	Items_Available	Daily_Customer_Count	Store_Sales
0	1	1659	1961	530	66490
1	2	1461	1752	210	39820
2	3	1340	1609	720	54010
3	4	1451	1748	620	53730
4	5	1770	2111	450	46620

In [4]:

```
data.tail()
```

Out[4]:

	Store ID	Store_Area	Items_Available	Daily_Customer_Count	Store_Sales
891	892	1582	1910	1080	66390
892	893	1387	1663	850	82080
893	894	1200	1436	1060	76440
894	895	1299	1560	770	96610
895	896	1174	1429	1110	54340

In [5]:

```
1 data.shape
```

Out[5]:

```
(896, 5)
```

In [6]:

```
1 data.columns
```

Out[6]:

```
Index(['Store ID ', 'Store_Area', 'Items_Available', 'Daily_Customer_Coun
t',
       'Store_Sales'],
      dtype='object')
```

In [7]:

```
1 data.duplicated().sum()
```

Out[7]:

```
0
```

In [8]:

```
1 data.isnull().sum()
```

Out[8]:

```
Store ID               0
Store_Area             0
Items_Available        0
Daily_Customer_Count   0
Store_Sales            0
dtype: int64
```

In [9]:

```
1 data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 896 entries, 0 to 895
Data columns (total 5 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Store ID             896 non-null    int64
 1   Store_Area           896 non-null    int64
 2   Items_Available      896 non-null    int64
 3   Daily_Customer_Count 896 non-null    int64
 4   Store_Sales          896 non-null    int64
dtypes: int64(5)
memory usage: 35.1 KB
```

In [10]:

```
1 data.describe()
```

Out[10]:

	Store ID	Store_Area	Items_Available	Daily_Customer_Count	Store_Sales
count	896.000000	896.000000	896.000000	896.000000	896.000000
mean	448.500000	1485.409598	1782.035714	786.350446	59351.305804
std	258.797218	250.237011	299.872053	265.389281	17190.741895
min	1.000000	775.000000	932.000000	10.000000	14920.000000
25%	224.750000	1316.750000	1575.500000	600.000000	46530.000000
50%	448.500000	1477.000000	1773.500000	780.000000	58605.000000
75%	672.250000	1653.500000	1982.750000	970.000000	71872.500000
max	896.000000	2229.000000	2667.000000	1560.000000	116320.000000

In [11]:

```
1 data.nunique()
```

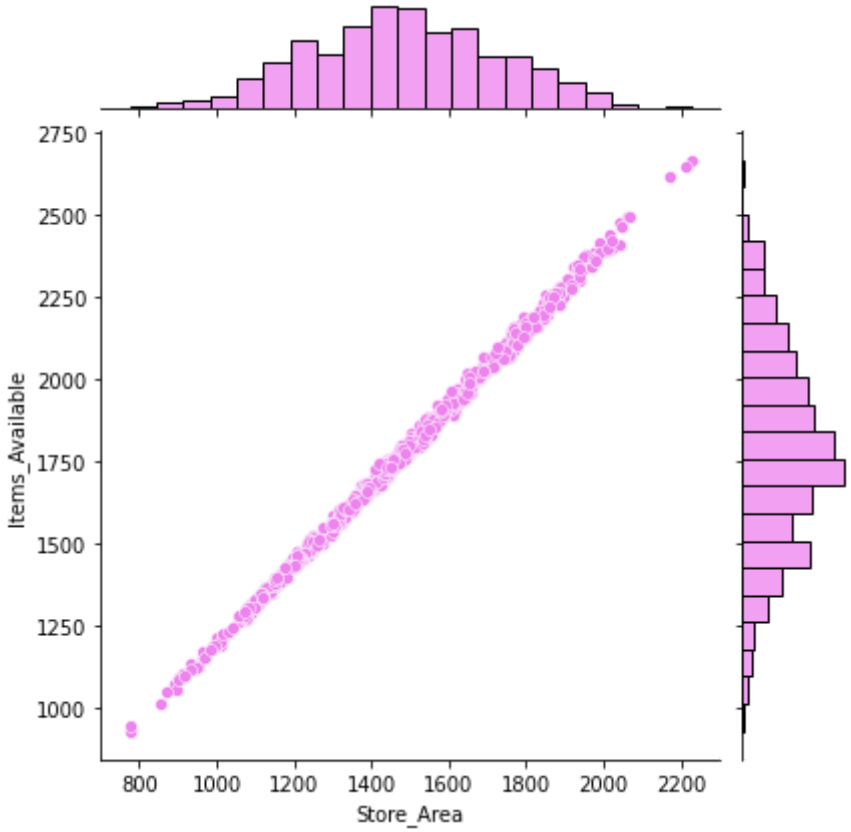
Out[11]:

```
Store ID                896
Store_Area              583
Items_Available         616
Daily_Customer_Count    130
Store_Sales             816
dtype: int64
```

In [17]:

```
plt.figure(figsize=(15,6))
sns.jointplot(x='Store_Area', y='Items_Available',
              data=data, color="violet")
plt.show()
```

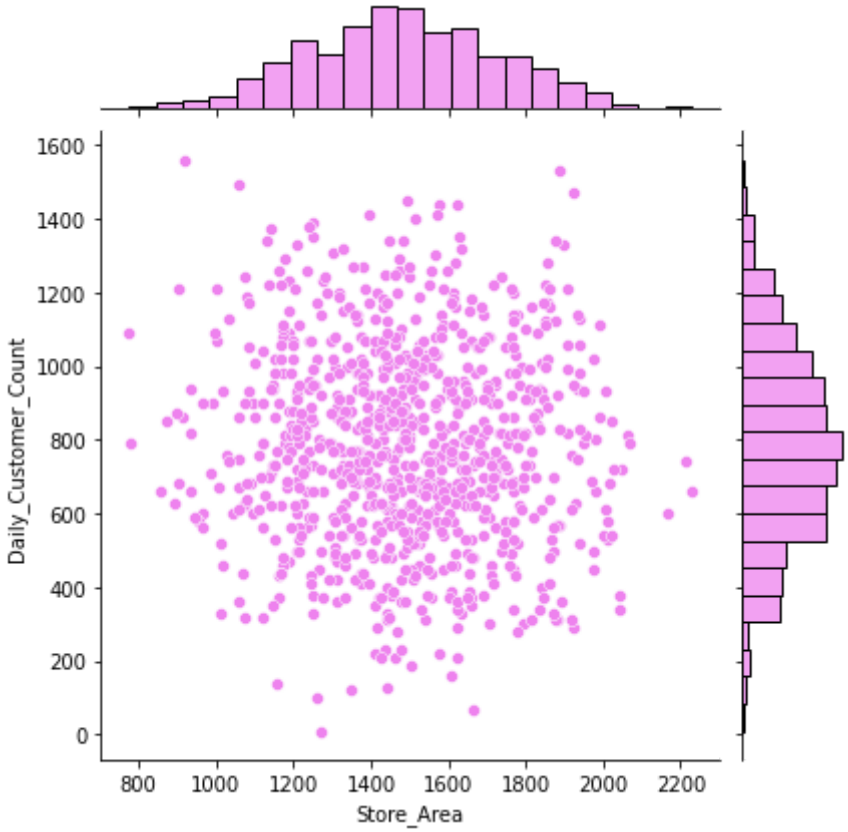
<Figure size 1080x432 with 0 Axes>



In [18]:

```
plt.figure(figsize=(15,6))
sns.jointplot(x='Store_Area', y='Daily_Customer_Count',
              data=data, color="violet")
plt.show()
```

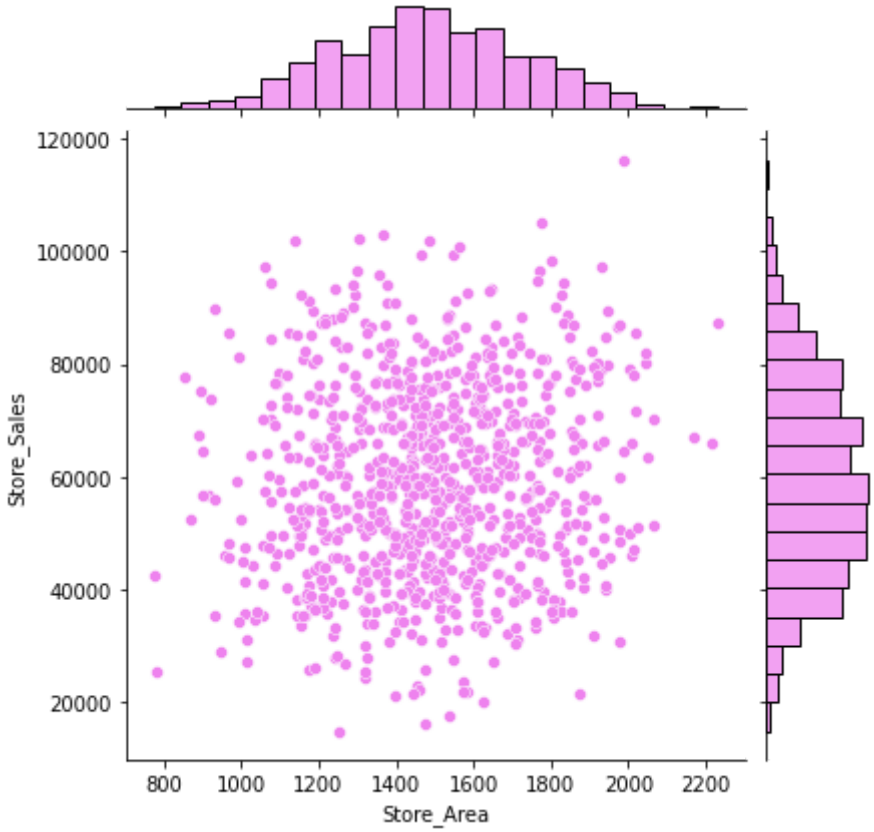
```
<Figure size 1080x432 with 0 Axes>
```



In [19]:

```
plt.figure(figsize=(15,6))
sns.jointplot(x='Store_Area', y='Store_Sales',
              data=data, color="violet")
plt.show()
```

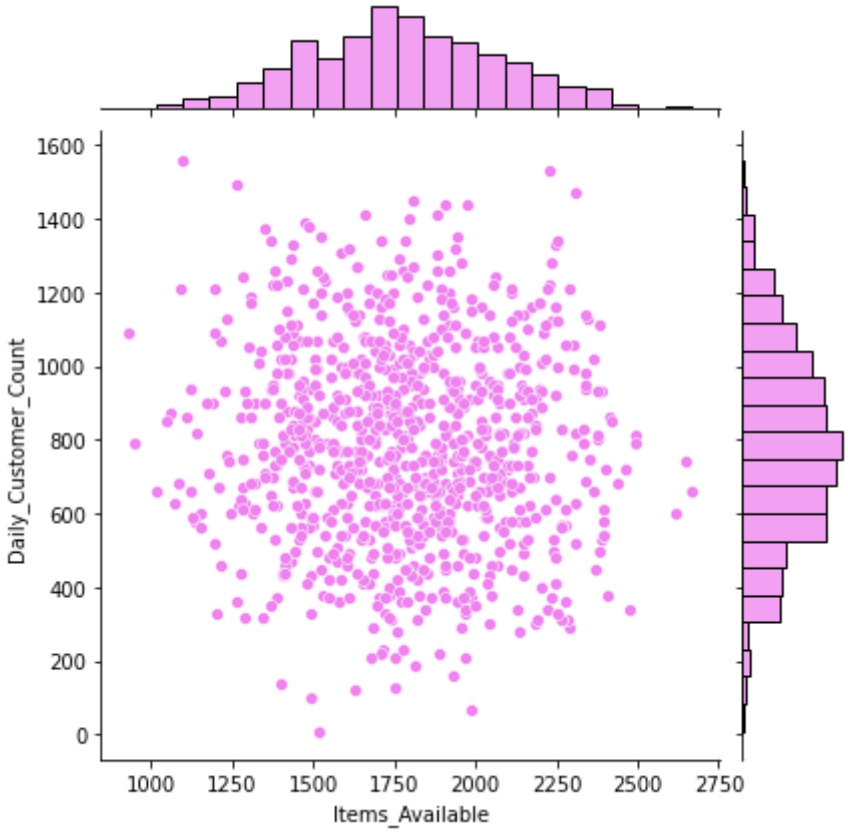
```
<Figure size 1080x432 with 0 Axes>
```



In [20]:

```
plt.figure(figsize=(15,6))
sns.jointplot(x='Items_Available', y='Daily_Customer_Count',
              data=data, color="violet")
plt.show()
```

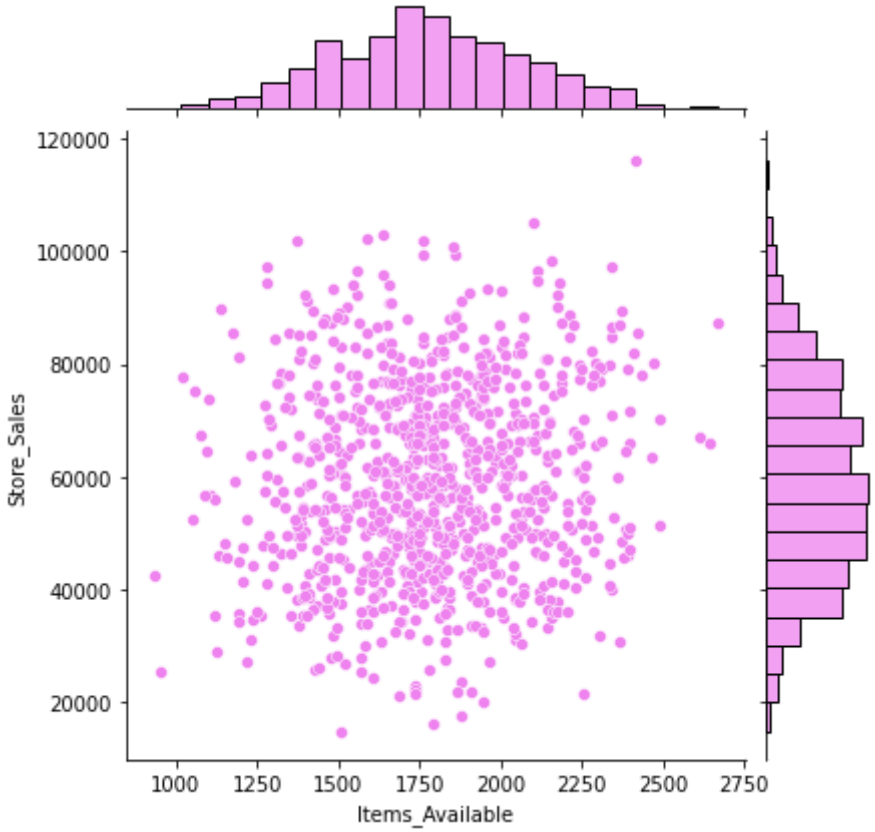
<Figure size 1080x432 with 0 Axes>



In [21]:

```
plt.figure(figsize=(15,6))
sns.jointplot(x='Items_Available', y='Store_Sales',
              data=data, color="violet")
plt.show()
```

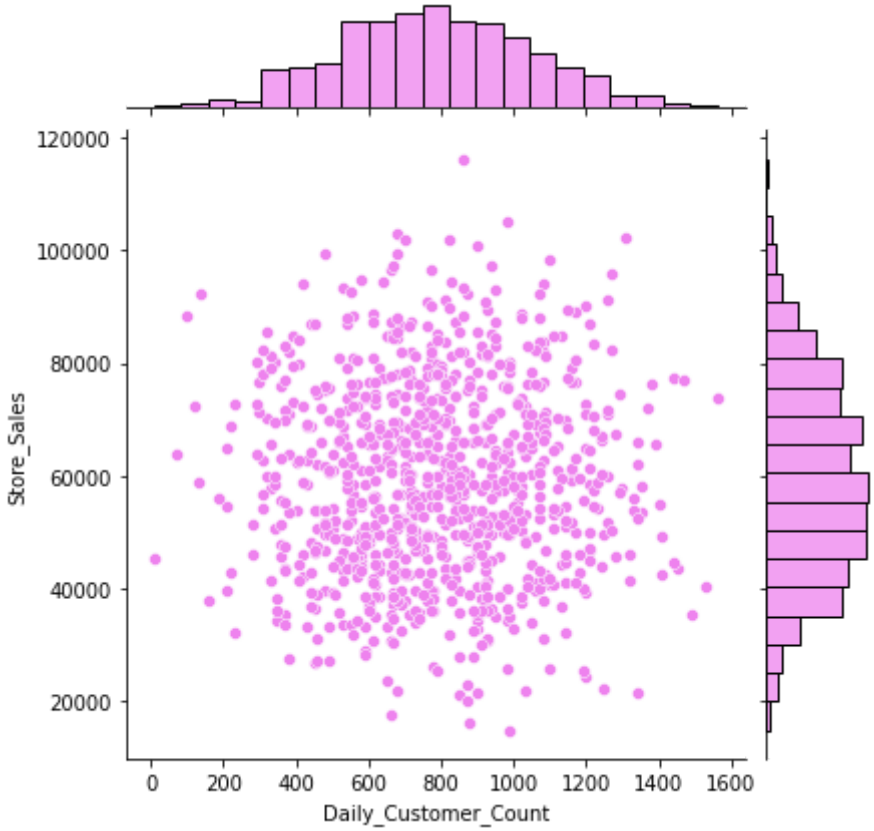
```
<Figure size 1080x432 with 0 Axes>
```



In [22]:

```
plt.figure(figsize=(15,6))
sns.jointplot(x='Daily_Customer_Count', y='Store_Sales',
              data=data, color="violet")
plt.show()
```

```
<Figure size 1080x432 with 0 Axes>
```



In [25]:

```
data_items_available = data.sort_values(by=['Items_Available'], ascending = False)
```

In [26]:

```
1 data_items_available
```

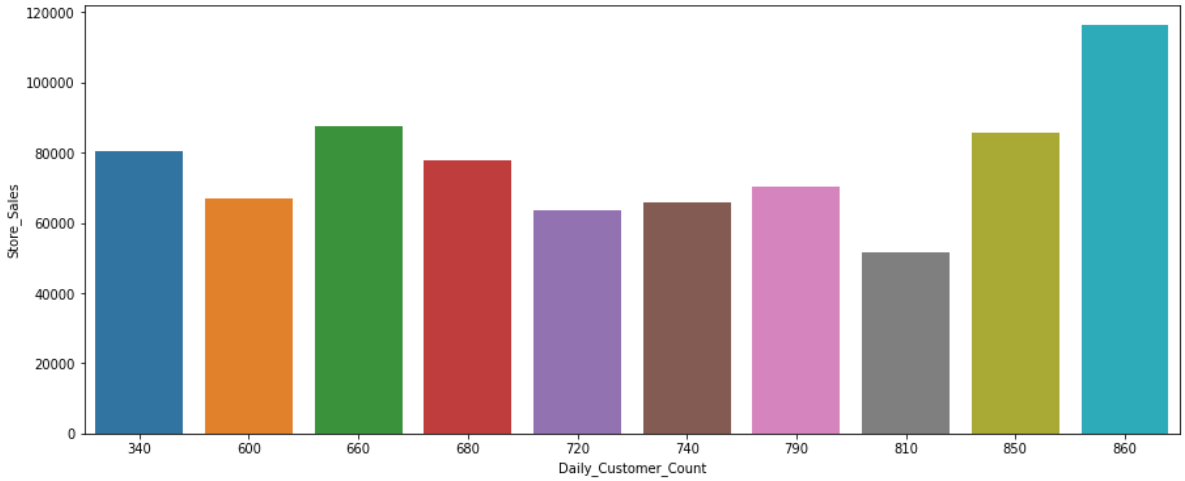
Out[26]:

	Store ID	Store_Area	Items_Available	Daily_Customer_Count	Store_Sales
466	467	2229	2667	660	87410
540	541	2214	2647	740	65900
91	92	2169	2617	600	67080
398	399	2063	2493	810	51480
849	850	2067	2492	790	70230
...	...	...	...	...	...
406	407	896	1059	870	75110
549	550	869	1050	850	52540
744	745	854	1018	660	77740
865	866	780	951	790	25600
158	159	775	932	1090	42530

896 rows × 5 columns

In [30]:

```
1 plt.figure(figsize=(15,6))
2 sns.barplot(x='Daily_Customer_Count', y='Store_Sales',
3             data=data_items_available.head(10))
4 plt.show()
```



In [31]:

```
1 data_daily_customer_count = data.sort_values(by=['Daily_Customer_Count'], ascending
```

In [32]:

```
1 data_daily_customer_count
```

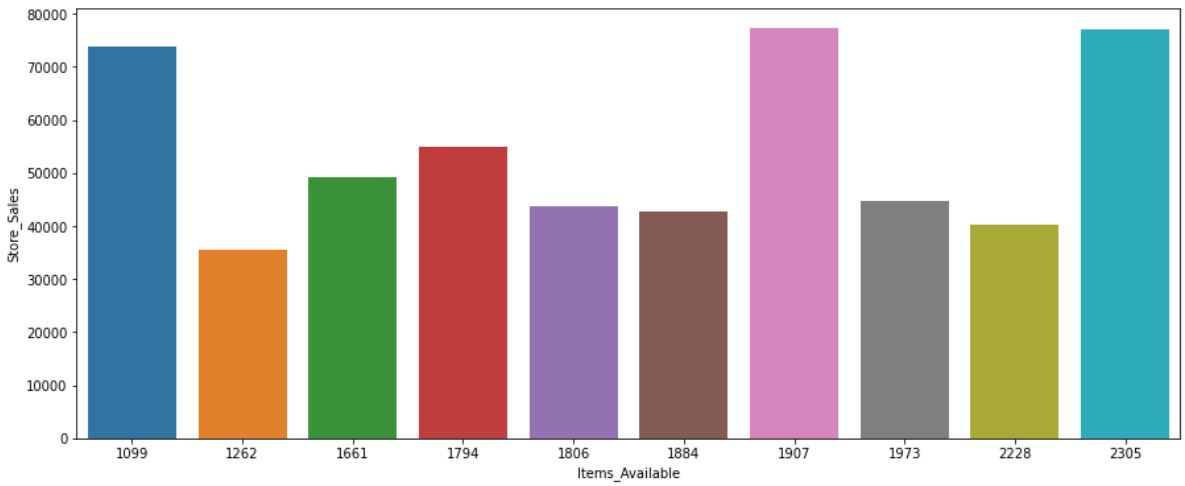
Out[32]:

	Store ID	Store_Area	Items_Available	Daily_Customer_Count	Store_Sales
848	849	919	1099	1560	73810
349	350	1886	2228	1530	40350
535	536	1057	1262	1490	35420
94	95	1921	2305	1470	77120
312	313	1494	1806	1450	43640
...	...	...	...	...	...
146	147	1442	1750	130	58920
230	231	1347	1628	120	72350
876	877	1259	1493	100	88270
151	152	1662	1986	70	63730
39	40	1270	1516	10	45480

896 rows × 5 columns

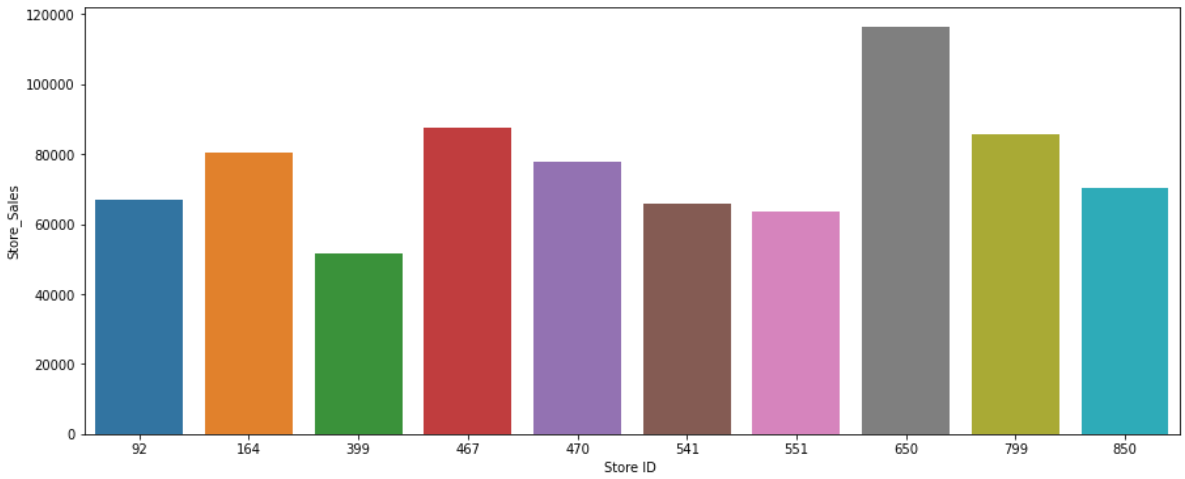
In [33]:

```
1 plt.figure(figsize=(15,6))
2 sns.barplot(x='Items_Available', y='Store_Sales',
3             data=data_daily_customer_count.head(10))
4 plt.show()
```



In [37]:

```
plt.figure(figsize=(15,6))
sns.barplot(x='Store ID ', y='Store_Sales',
            data=data_items_available.head(10))
plt.show()
```



In [38]:

```
plt.figure(figsize=(15,6))
sns.barplot(x='Store ID ', y='Store_Sales',
            data=data_daily_customer_count.head(10))
plt.show()
```