In [1]:
```
# Importing libraries

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [3]:
```
# Importing Dataset

df = pd.read_csv("Sales Data.csv")
df.head()
```

Out[3]:

	Unnamed: 0	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Sa
0	0	295665	Macbook Pro Laptop	1	1700.00	2019-12-30 00:01:00	136 Church St, New York City, NY 10001	12	1700
1	1	295666	LG Washing Machine	1	600.00	2019-12-29 07:03:00	562 2nd St, New York City, NY 10001	12	600
2	2	295667	USB-C Charging Cable	1	11.95	2019-12-12 18:21:00	277 Main St, New York City, NY 10001	12	11
3	3	295668	27in FHD Monitor	1	149.99	2019-12-22 15:13:00	410 6th St, San Francisco, CA 94016	12	149
4	4	295669	USB-C Charging Cable	1	11.95	2019-12-18 12:38:00	43 Hill St, Atlanta, GA 30301	12	11

In [6]:
```
df.shape
```

Out[6]:
```
(185950, 11)
```

In [7]:
```
# Droping unnessasary columns

df.drop(['Order ID', 'Unnamed: 0', 'Order Date', 'Purchase Address'], axis = 1,
```

In [8]:
```
df.head(2)
```

Out[8]:

	Product	Quantity Ordered	Price Each	Month	Sales	City	Hour
0	Macbook Pro Laptop	1	1700.0	12	1700.0	New York City	0
1	LG Washing Machine	1	600.0	12	600.0	New York City	7

In [9]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 185950 entries, 0 to 185949
Data columns (total 7 columns):
 #   Column            Non-Null Count   Dtype  
---  ------            --------------   -----  
 0   Product           185950 non-null  object 
 1   Quantity Ordered  185950 non-null  int64  
 2   Price Each        185950 non-null  float64
 3   Month             185950 non-null  int64  
 4   Sales             185950 non-null  float64
 5   City              185950 non-null  object 
 6   Hour              185950 non-null  int64  
dtypes: float64(2), int64(3), object(2)
memory usage: 9.9+ MB
```

Observations

1. Categorical Columns:

- Product
- City

2. Numerical Columns:

- Quantity Ordered
- Price Each
- Month
- Hour

In [11]:
```
# Checking for missing values

df.isnull().sum()
```

Out[11]:
```
Product             0
Quantity Ordered    0
Price Each          0
Month               0
Sales               0
City                0
Hour                0
dtype: int64
```

There are no missing values.

In [12]:
```
# Checking for duplicates

df.duplicated().sum()
```

Out[12]: 141128

There are repeated values, we'll need to remove them

In [13]:
```
df = df.drop_duplicates()
df.shape
```

Out[13]: (44822, 7)

In [14]:
```
df.head(2)
```

Out[14]:

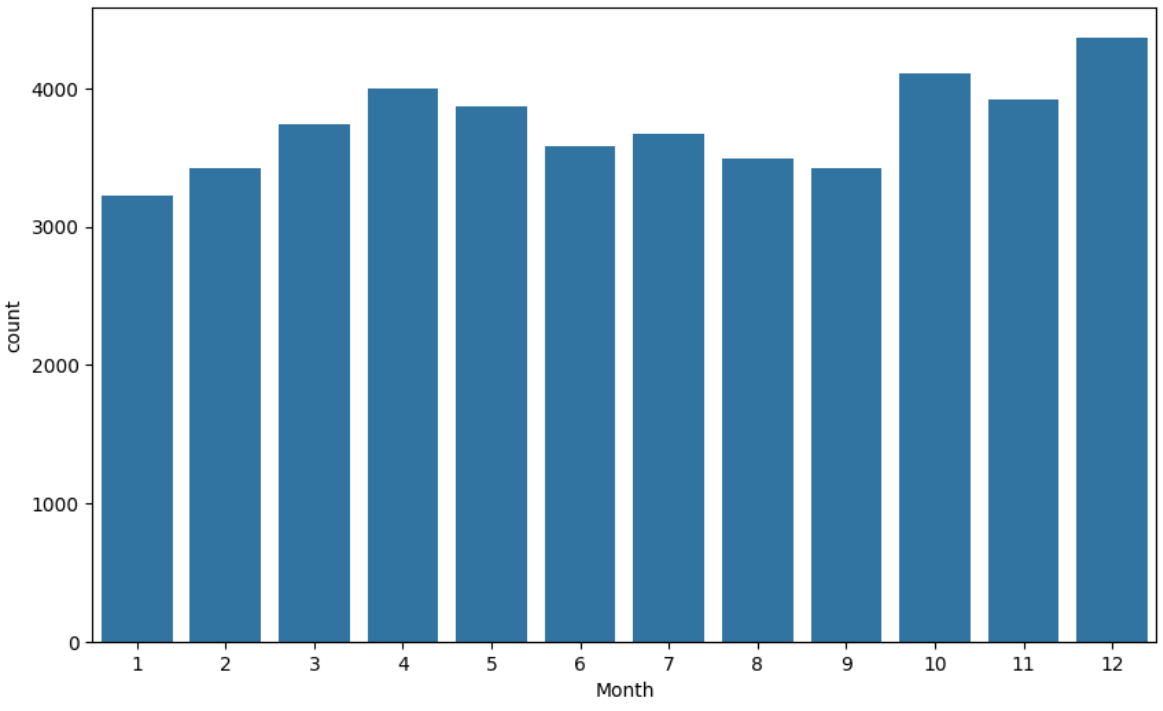
	Product	Quantity Ordered	Price Each	Month	Sales	City	Hour
0	Macbook Pro Laptop	1	1700.0	12	1700.0	New York City	0
1	LG Washing Machine	1	600.0	12	600.0	New York City	7

Visualizing the Data

In [17]:
```
# Which month has the highest sales?

plt.figure(figsize = (10,6))
sns.countplot(data = df, x = 'Month')
```

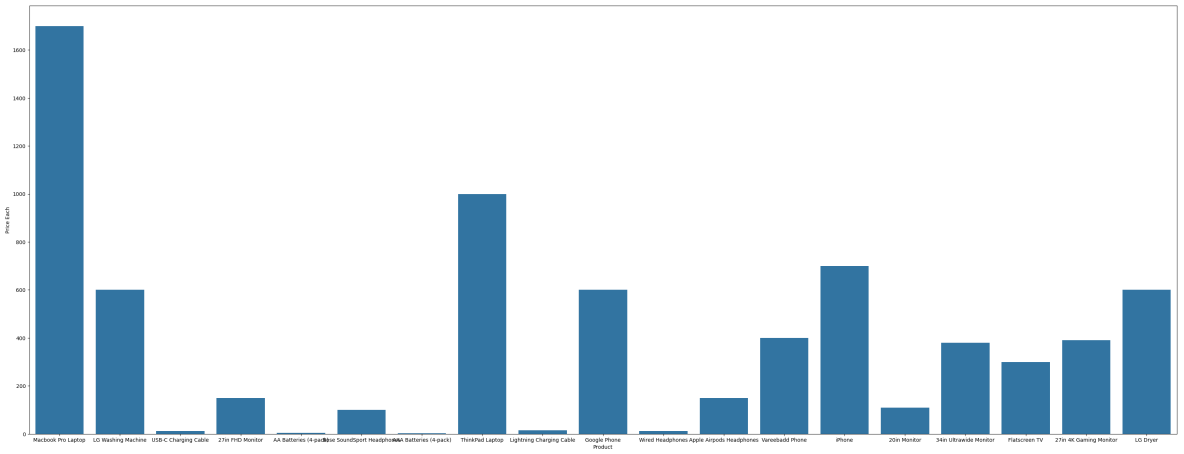
Out[17]: <Axes: xlabel='Month', ylabel='count'>



As we can see, December has the highest sales

In [19]:
```
plt.figure(figsize = (40,15))
sns.barplot(data = df, x = 'Product', y = 'Price Each')
```

Out[19]: <Axes: xlabel='Product', ylabel='Price Each'>



In [22]:
```
df[['Product', 'Price Each']].drop_duplicates()
```

Out[22]:

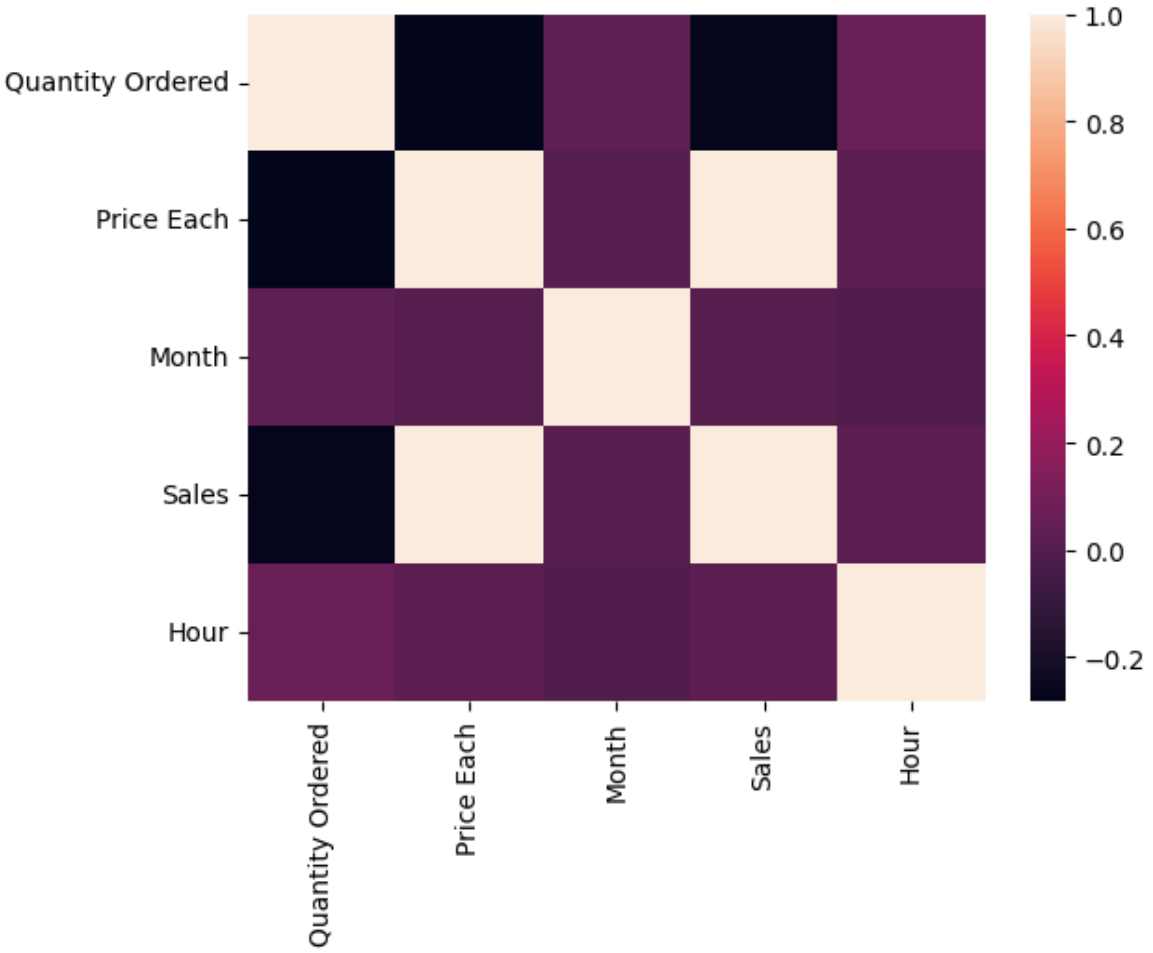
	Product	Price Each
0	Macbook Pro Laptop	1700.00
1	LG Washing Machine	600.00
2	USB-C Charging Cable	11.95
3	27in FHD Monitor	149.99
5	AA Batteries (4-pack)	3.84
8	Bose SoundSport Headphones	99.99
9	AAA Batteries (4-pack)	2.99
11	ThinkPad Laptop	999.99
15	Lightning Charging Cable	14.95
16	Google Phone	600.00
19	Wired Headphones	11.99
25	Apple Airpods Headphones	150.00
36	Vareebadd Phone	400.00
54	iPhone	700.00
56	20in Monitor	109.99
61	34in Ultrawide Monitor	379.99
89	Flatscreen TV	300.00
143	27in 4K Gaming Monitor	389.99
745	LG Dryer	600.00

MacBook has the highest Price

In [24]:
```
numeric_df = df.select_dtypes(include=['number'])
correlation_matrix = numeric_df.corr()

sns.heatmap(correlation_matrix)
```

Out[24]: <Axes: >



In [27]:
```
products = df["Product"].unique()
```

In [36]:
```
product_quantities = df[["Product", "Quantity Ordered"]].groupby("Product").sum(
product_quantities = product_quantities.sort_values("Quantity Ordered", ascendin

product_quantities.head()
```

Out[36]:

Product	Quantity Ordered
AAA Batteries (4-pack)	11926
AA Batteries (4-pack)	9138
USB-C Charging Cable	5014
Wired Headphones	4672
Lightning Charging Cable	4533

We can see the 5 most sold products in sense of number of quantity orders.