

# Sales Analysis

## Import Libraries

```
In [1]: import pandas as pd
import matplotlib.pyplot as plt
```

```
In [2]: df= pd.read_csv(r"D:\Console Flare\Pandas-Data-Science-Tasks-master\SalesAnalysis\Output\all_data.csv")
```

```
In [3]: df.head(5)
```

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address
0	176558	USB-C Charging Cable	2	11.95	04/19/19 08:46	917 1st St, Dallas, TX 75001
1	NaN	NaN	NaN	NaN	NaN	NaN
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215
3	176560	Google Phone	1	600	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001

## Clean up the data!

### Drop rows of NaN values

```
In [4]: nan_df= df[df.isna().any(axis=1)]
nan_df.head()

df=df.dropna(how='all')
df.head()
```

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address
0	176558	USB-C Charging Cable	2	11.95	04/19/19 08:46	917 1st St, Dallas, TX 75001
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215
3	176560	Google Phone	1	600	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001
5	176561	Wired Headphones	1	11.99	04/30/19 09:27	333 8th St, Los Angeles, CA 90001

### Find 'Or' and delete it

```
In [5]: df= df[df['Order Date'].str[0:2]!='Or']
```

### Convert columns to the correct type

```
In [6]: df['Quantity Ordered']= pd.to_numeric(df['Quantity Ordered'])# Make int
df['Price Each']=pd.to_numeric(df['Price Each'])# Make float
```

## Augment data with additional columns

### Add Month column

```
In [7]: df['Month']= df['Order Date'].str[0:2]
df['Month']= df['Month'].astype('int32')
df.head()
```

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month
0	176558	USB-C Charging Cable	2	11.95	04/19/19 08:46	917 1st St, Dallas, TX 75001	4
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215	4
3	176560	Google Phone	1	600.00	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4
5	176561	Wired Headphones	1	11.99	04/30/19 09:27	333 8th St, Los Angeles, CA 90001	4

### Add a city column

```
In [21]: # use a .apply()
df['City']=df['Purchase Address'].apply(lambda x: x.split(',')[1])
df.head()
```

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Sales	City
0	176558	USB-C Charging Cable	2	11.95	04/19/19 08:46	917 1st St, Dallas, TX 75001	4	23.90	Dallas
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215	4	99.99	Boston
3	176560	Google Phone	1	600.00	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4	600.00	Los Angeles
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4	11.99	Los Angeles
5	176561	Wired Headphones	1	11.99	04/30/19 09:27	333 8th St, Los Angeles, CA 90001	4	11.99	Los Angeles

### Add a Sales column

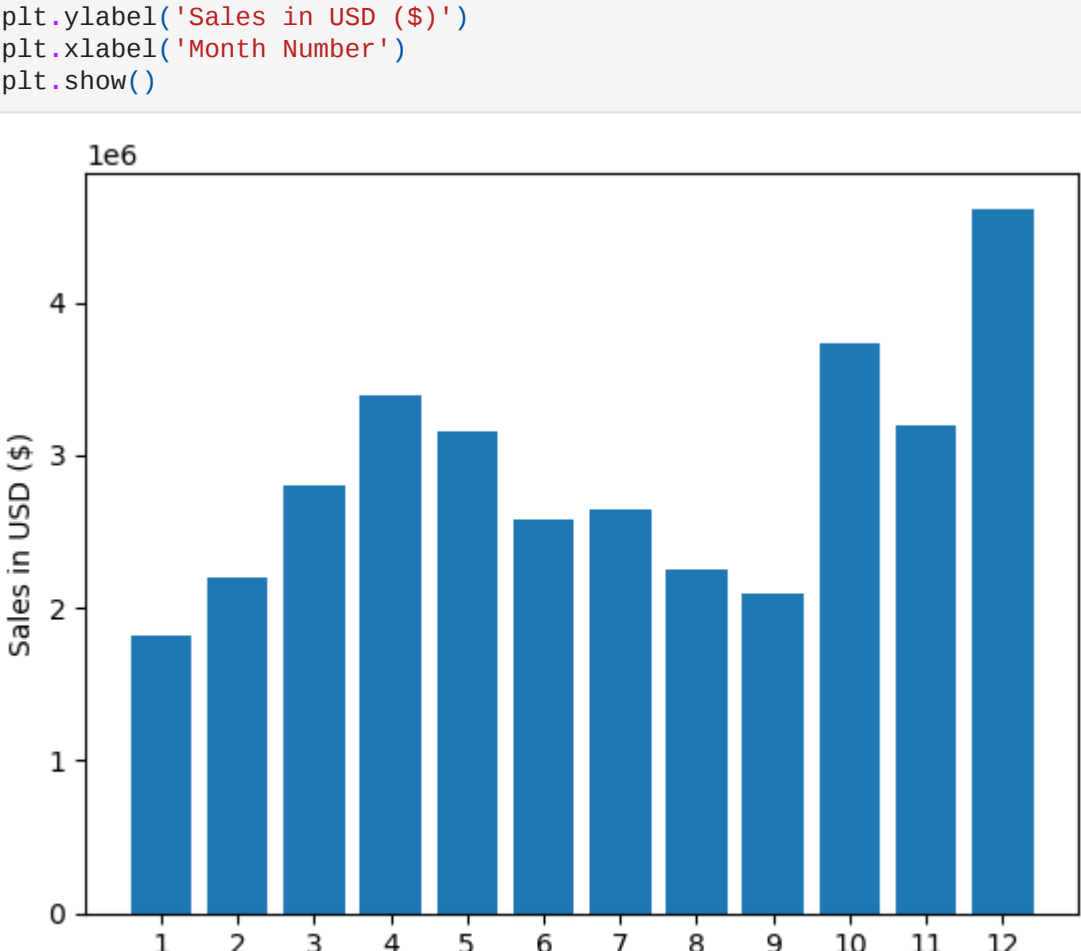
```
In [8]: df['Sales']=df['Quantity Ordered'] * df['Price Each']
df.head()
```

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Sales
0	176558	USB-C Charging Cable	2	11.95	04/19/19 08:46	917 1st St, Dallas, TX 75001	4	23.90
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215	4	99.99
3	176560	Google Phone	1	600.00	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4	600.00
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	4	11.99
5	176561	Wired Headphones	1	11.99	04/30/19 09:27	333 8th St, Los Angeles, CA 90001	4	11.99

## Best month for Sales

```
In [9]: results =df.groupby('Month').sum()
```

```
In [14]: months=range(1,13)
plt.bar(months,results['Sales'])
plt.xticks(months)
plt.ylabel('Sales in USD ($)')
plt.xlabel('Month Number')
plt.show()
```



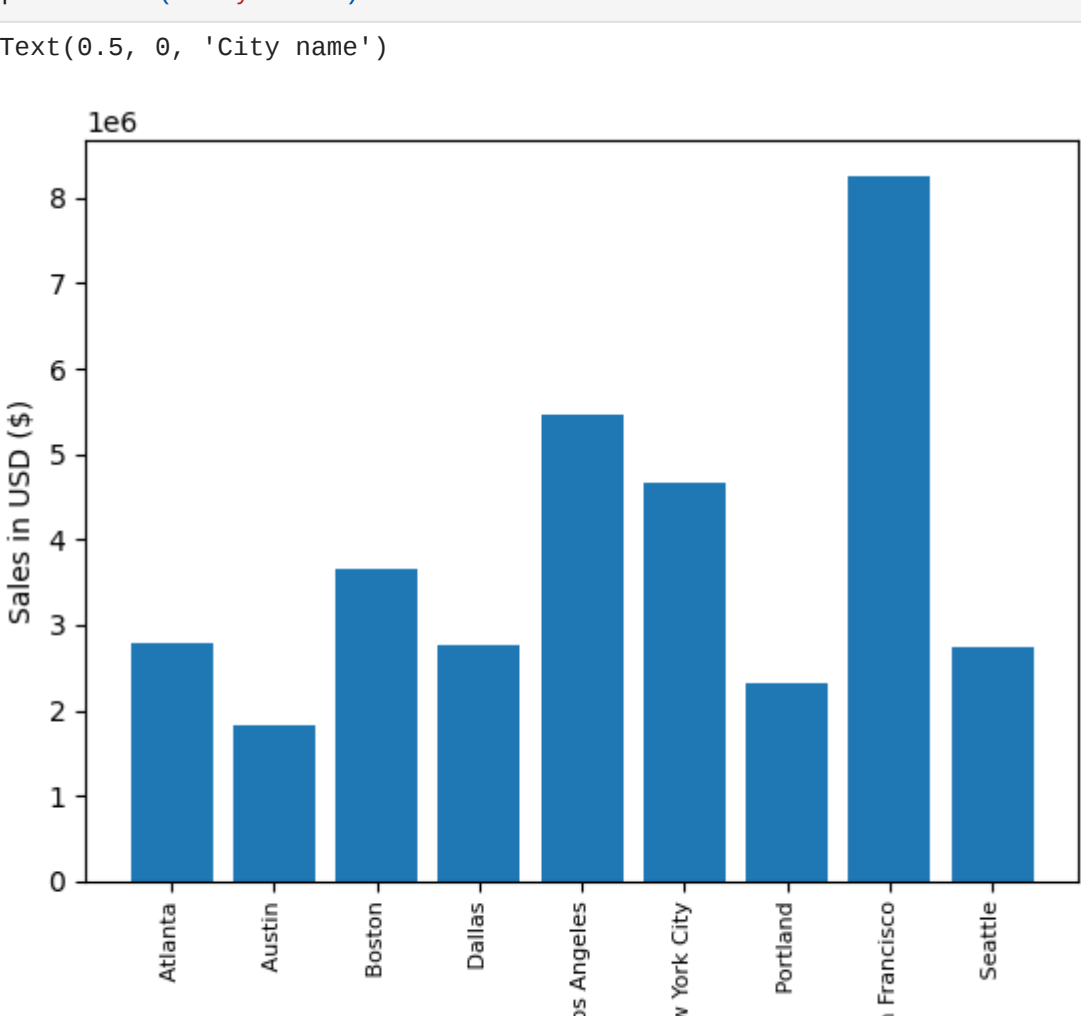
## City had the highest number of sales?

```
In [31]: result=df.groupby('City').sum()
result
```

	Quantity Ordered	Price Each	Month	Sales
City				
Atlanta	16602	2779908.20	104794	2795498.58
Austin	11153	1809873.61	69829	1819581.75
Boston	22528	3637409.77	141112	3661642.01
Dallas	16730	2752627.82	104620	2767975.40
Los Angeles	33289	5421435.23	208325	5452570.80
New York City	27932	4635370.83	175741	4664317.43
Portland	14053	2307747.47	87765	2320490.61
San Francisco	50239	8211461.74	315520	8262203.91
Seattle	16553	2733296.01	104941	2747755.48

```
In [37]: cities=[city for city,df in df.groupby('City')]
plt.bar(cities,result['Sales'])
plt.xticks(cities,rotation='vertical',size=8)
plt.ylabel('Sales in USD ($)')
plt.xlabel('City name')
plt.show()
```

```
Out[37]: Text(0.5, 0, 'City name')
```



## what time should we display advertisements to maximize likelihood of customer's buying product?

```
In [40]: df['Order Date']=pd.to_datetime(df['Order Date'])
```

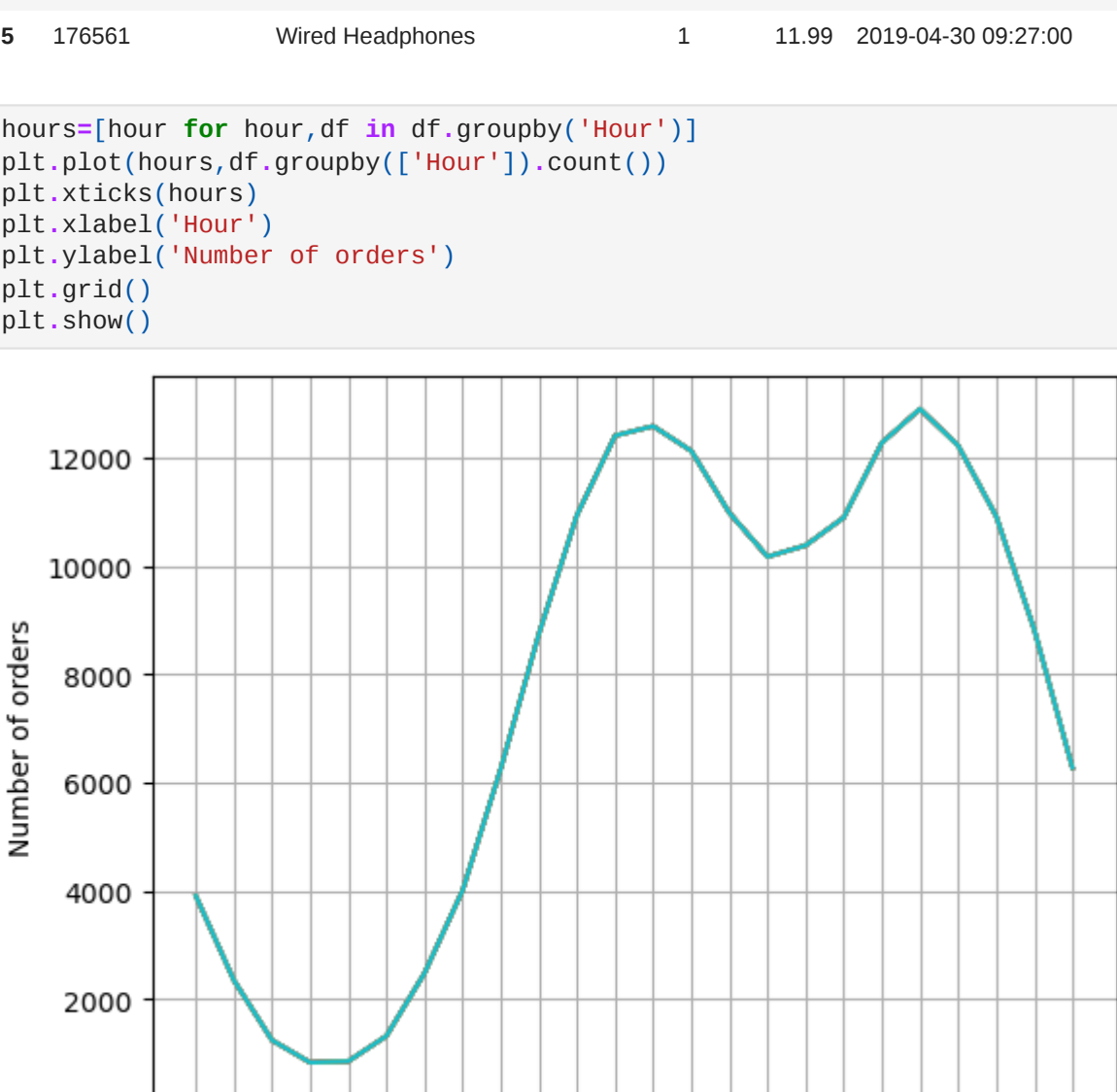
```
In [39]: df.head()
```

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Sales	City
0	176558	USB-C Charging Cable	2	11.95	2019-04-19 08:46:00	917 1st St, Dallas, TX 75001	4	23.90	Dallas
2	176559	Bose SoundSport Headphones	1	99.99	2019-04-07 22:30:00	682 Chestnut St, Boston, MA 02215	4	99.99	Boston
3	176560	Google Phone	1	600.00	2019-04-12 14:38:00	669 Spruce St, Los Angeles, CA 90001	4	600.00	Los Angeles
4	176560	Wired Headphones	1	11.99	2019-04-12 14:38:00	669 Spruce St, Los Angeles, CA 90001	4	11.99	Los Angeles
5	176561	Wired Headphones	1	11.99	2019-04-30 09:27:00	333 8th St, Los Angeles, CA 90001	4	11.99	Los Angeles

```
In [43]: df['Hour']=df['Order Date'].dt.hour
df['Minute']=df['Order Date'].dt.minute
df.head()
```

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Sales	City	Hour	Minute
0	176558	USB-C Charging Cable	2	11.95	2019-04-19 08:46:00	917 1st St, Dallas, TX 75001	4	23.90	Dallas	8	46
2	176559	Bose SoundSport Headphones	1	99.99	2019-04-07 22:30:00	682 Chestnut St, Boston, MA 02215	4	99.99	Boston	22	30
3	176560	Google Phone	1	600.00	2019-04-12 14:38:00	669 Spruce St, Los Angeles, CA 90001	4	600.00	Los Angeles	14	38
4	176560	Wired Headphones	1	11.99	2019-04-12 14:38:00	669 Spruce St, Los Angeles, CA 90001	4	11.99	Los Angeles	14	38
5	176561	Wired Headphones	1	11.99	2019-04-30 09:27:00	333 8th St, Los Angeles, CA 90001	4	11.99	Los Angeles	9	27

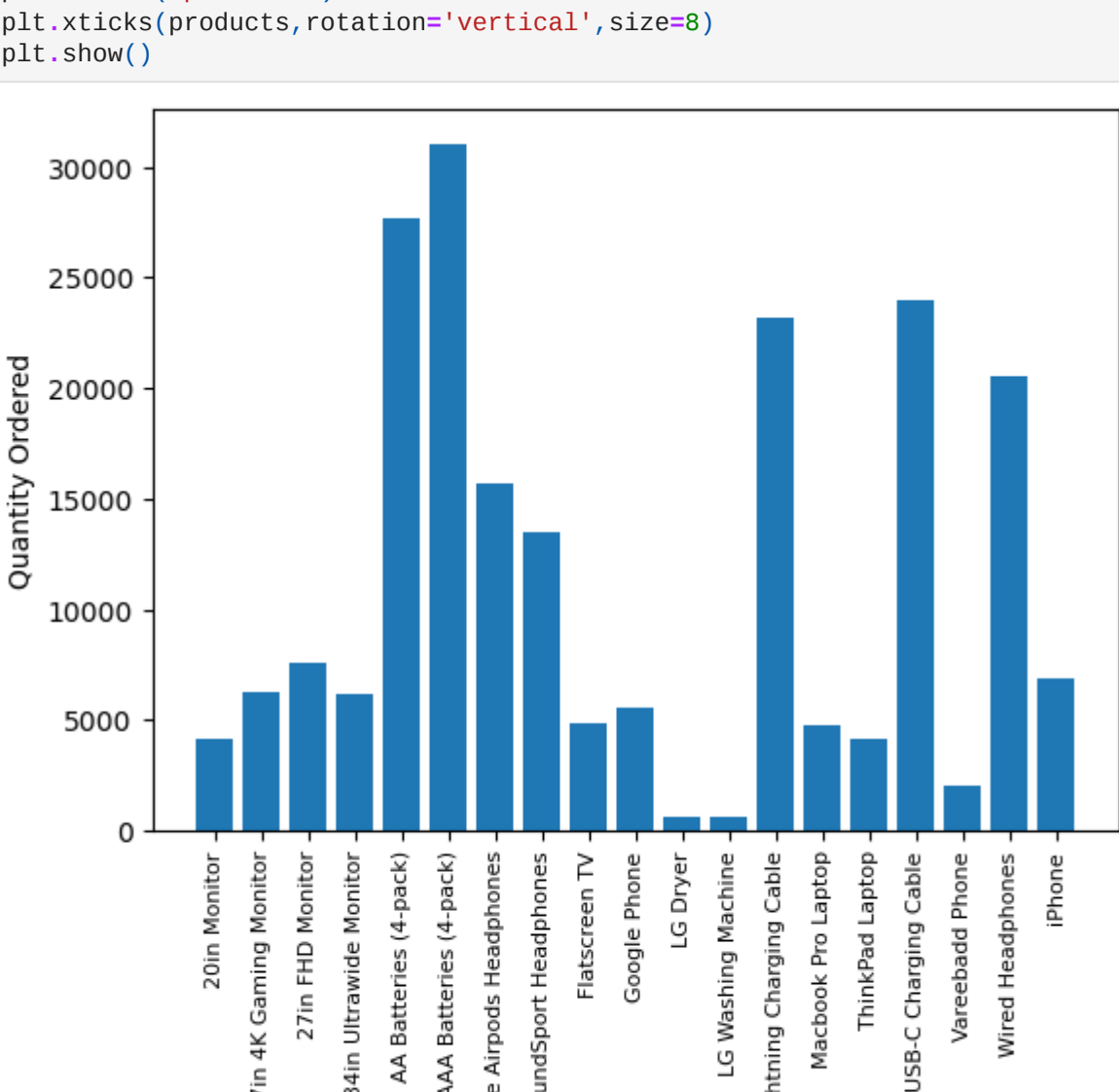
```
In [47]: hours=[hour for hour,df in df.groupby('Hour')]
plt.pl(hours,df.groupby(['Hour']).count())
plt.xticks(hours)
plt.xlabel('Hour')
plt.ylabel('Number of orders')
plt.grid()
plt.show()
```



## Most sold product

```
In [71]: product_group=df.groupby('Product')
quantity_order =product_group.sum()['Quantity Ordered']
```

```
products =[product for product, df in product_group]
plt.bar(products,quantity_order)
plt.ylabel('Quantity Ordered')
plt.xlabel('products')
plt.xticks(products,rotation='vertical',size=8)
plt.show()
```

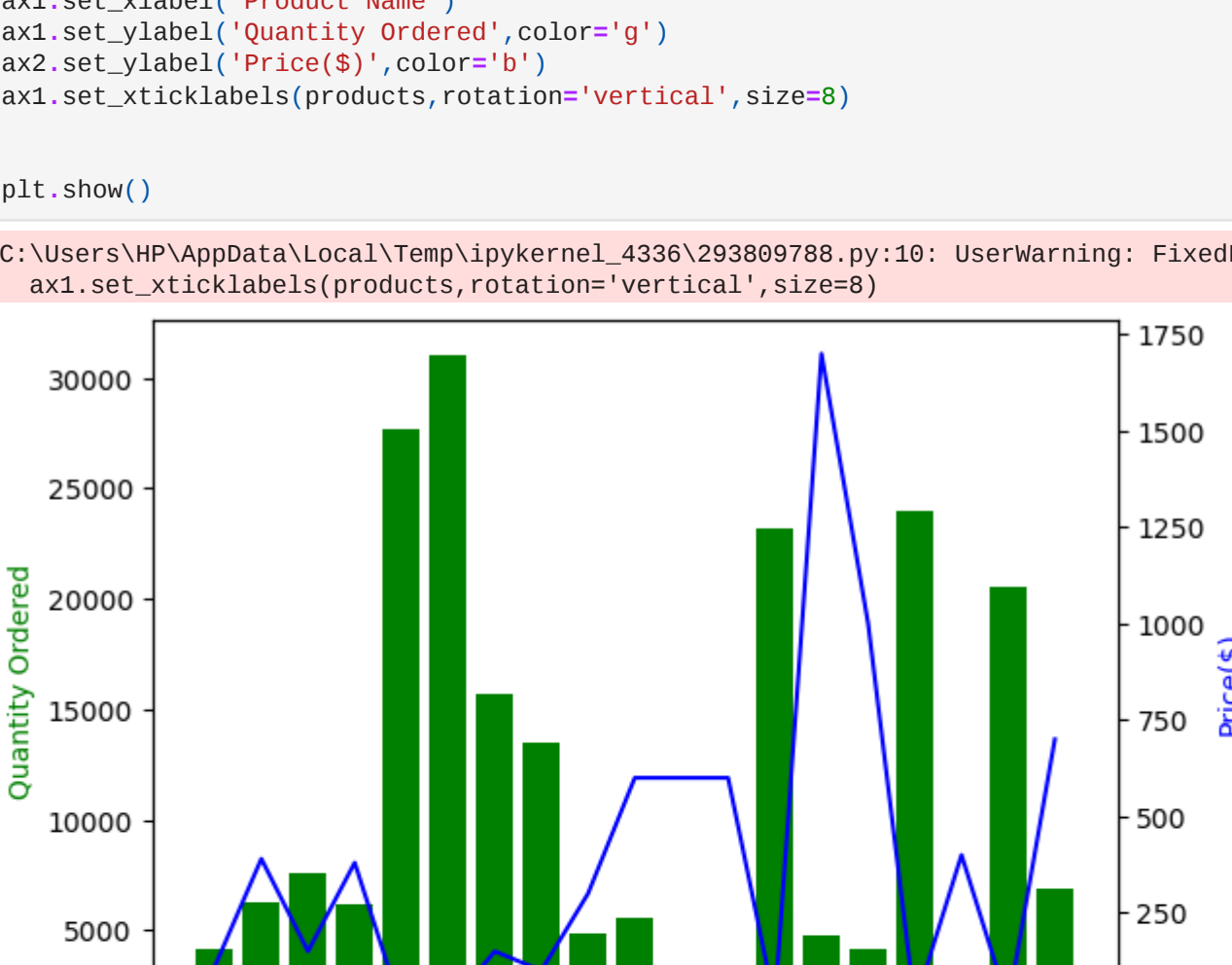


```
In [83]: prices = df.groupby('Product').mean()['Price Each']
fig, ax1 =plt.subplots()
ax2 = ax1.twinx()
ax1.bar(products,quantity_order,color='g')
ax2.plot(products,prices,color='b')
```

```
ax1.set_xlabel('Product Name')
ax1.set_ylabel('Quantity Ordered',color='g')
ax2.set_ylabel('Price($)',color='b')
ax1.set_xticklabels(products,rotation='vertical',size=8)
```

```
plt.show()

C:\Users\NP\AppData\Local\Temp\ipykernel_4336\292889788.py:10: UserWarning: FixedFormatter should only be used together with FixedLocator
ax1.set_xticklabels(products,rotation='vertical',size=8)
```



```
In [ ]:
```