Sales Data Analysis using Python Project

In [2]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import os
```

In [3]:
```
os.listdir("F:\Software\Data Science\Python Data Science Student Exercise\P
```

Out[3]:
```
['allData.csv',
 'Sales_April_2019.csv',
 'Sales_August_2019.csv',
 'Sales_December_2019.csv',
 'Sales_February_2019.csv',
 'Sales_January_2019.csv',
 'Sales_July_2019.csv',
 'Sales_June_2019.csv',
 'Sales_March_2019.csv',
 'Sales_May_2019.csv',
 'Sales_November_2019.csv',
 'Sales_October_2019.csv',
 'Sales_September_2019.csv']
```

In [18]:
```
Files=[]
for a in os.listdir('F:\Software\Data Science\Python Data Science Student E
    Files.append(a)
for a in Files:
    print(a)
```

```
allData.csv
Sales_April_2019.csv
Sales_August_2019.csv
Sales_December_2019.csv
Sales_February_2019.csv
Sales_January_2019.csv
Sales_July_2019.csv
Sales_June_2019.csv
Sales_March_2019.csv
Sales_May_2019.csv
Sales_November_2019.csv
Sales_October_2019.csv
Sales_September_2019.csv
```

In [19]:
```
#I want to load all the file one by one
allData=pd.DataFrame()
for a in Files:
    cr_df=pd.read_csv('F:/Software/Data Science/Python Data Science Student
    allData=pd.concat([allData,cr_df])
print(allData.shape)
```

```
(373700, 6)
```

In [21]:
```python
allData.to_csv('F:/Software/Data Science/Python Data Science Student Exerci
allData.head()
```

Out[21]:

	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

In [22]:
```python
allData.isnull().sum()
```

Out[22]:
```
Order ID            1090
Product             1090
Quantity Ordered    1090
Price Each          1090
Order Date          1090
Purchase Address    1090
dtype: int64
```

In [25]:
```python
allData=allData.dropna()
```

In [27]:
```python
allData.shape
```

Out[27]:
```
(372610, 6)
```

In [28]:
```python
allData.head()
```

Out[28]:

	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

Which Month is best for selling products ?

In [30]:

```
def month(m):
    m=m.split('/')
    return m[0]
m='04/19/19 08:46'
month(m)
```

Out[30]:

```
'04'
```

In [33]:

```
allData['Month']=allData['Order Date'].apply(month)
allData.head()
```

Out[33]:

	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	04
2	176559	Bose SoundSport Headphones	1	99.99	04/07/19 22:30	682 Chestnut St, Boston, MA 02215	04
3	176560	Google Phone	1	600	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	04
4	176560	Wired Headphones	1	11.99	04/12/19 14:38	669 Spruce St, Los Angeles, CA 90001	04
5	176561	Wired Headphones	1	11.99	04/30/19 09:27	333 8th St, Los Angeles, CA 90001	04

In [34]:

```
allData.dtypes
```

Out[34]:

```
Order ID            object
Product             object
Quantity Ordered    object
Price Each          object
Order Date          object
Purchase Address    object
Month               object
dtype: object
```

In [35]:

```
allData['Month'].unique()
```

Out[35]:

```
array(['04', '05', 'Order Date', '08', '09', '12', '01', '02', '03', '07',
       '06', '11', '10'], dtype=object)
```

In [36]:

```
allData=allData[allData.Month!='Order Date']
allData['Month'].unique()
```

Out[36]:

```
array(['04', '05', '08', '09', '12', '01', '02', '03', '07', '06', '11',
       '10'], dtype=object)
```

In [38]:

```
allData['Month']=allData['Month'].astype(int)
```

```
In [39]: allData.dtypes
```

```
Out[39]: Order ID            object
         Product             object
         Quantity Ordered    object
         Price Each          object
         Order Date          object
         Purchase Address    object
         Month                int32
         dtype: object
```

```
In [41]: allData['Quantity Ordered']=allData['Quantity Ordered'].astype(int)
         allData['Price Each']=allData['Price Each'].astype(float)
```

```
In [42]: allData.dtypes
```

```
Out[42]: Order ID              object
         Product               object
         Quantity Ordered       int32
         Price Each           float64
         Order Date            object
         Purchase Address      object
         Month                  int32
         dtype: object
```

```
In [44]: allData['Sales']=allData['Quantity Ordered']*allData['Price Each']
         allData.head()
```

Out[44]:

	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

In [45]:

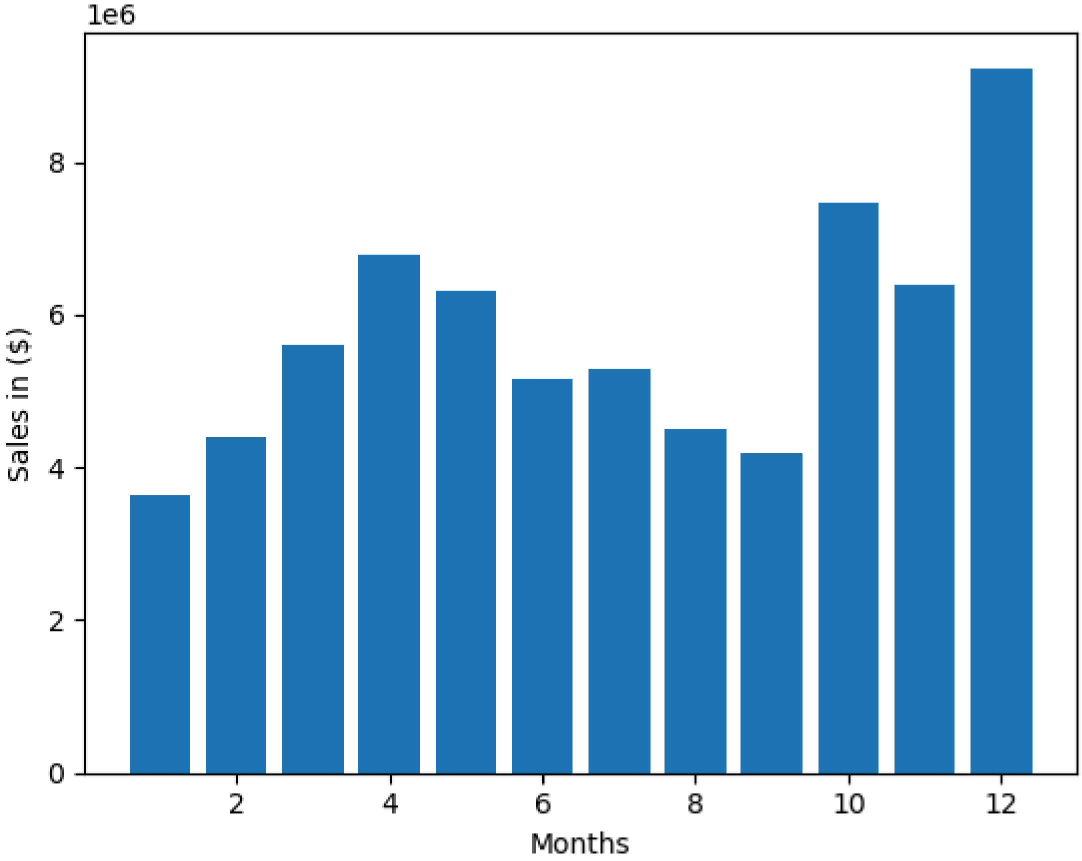
```
allData.groupby('Month')['Sales'].sum()
```

Out[45]:

```
Month
1     3644513.46
2     4404044.84
3     5614200.76
4     6781340.48
5     6305213.50
6     5155604.52
7     5295551.52
8     4488935.76
9     4195120.26
10    7473453.76
11    6399206.40
12    9226886.68
Name: Sales, dtype: float64
```

In [48]:

```
months=range(1,13)
plt.bar(months,allData.groupby('Month')['Sales'].sum())
plt.xlabel("Months")
plt.ylabel("Sales in ($)")
plt.show()
```



Which city orders the highest number of products?

In [49]:

```
def city(ct):
    ct=ct.split(',')[1]
    return ct
ct='917 1st St, Dallas, TX 75001'
city(ct)
```

Out[49]:

```
' Dallas'
```

In [50]:

```
allData['City']=allData['Purchase Address'].apply(city)
allData.head()
```

Out[50]:

	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

In [51]:

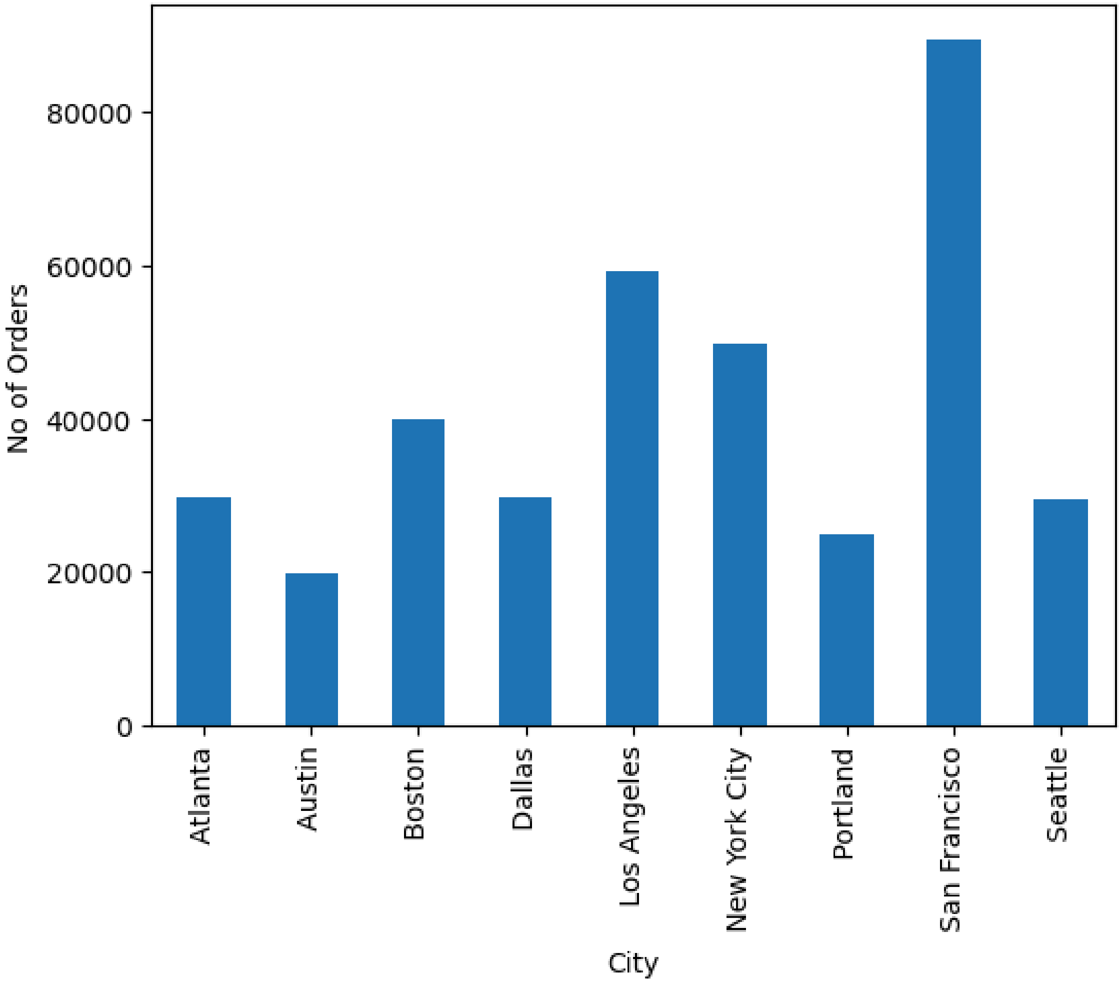
```
allData.groupby('City')['City'].count()
```

Out[51]:

```
City
 Atlanta          29762
 Austin           19810
 Boston           39868
 Dallas           29640
 Los Angeles      59210
 New York City    49752
 Portland         24930
 San Francisco    89464
 Seattle          29464
Name: City, dtype: int64
```

```
In [53]: allData.groupby('City')['City'].count().plot.bar()
         plt.ylabel("No of Orders")
```

```
Out[53]: Text(0, 0.5, 'No of Orders')
```



What time of the day do people order the most goods online ?

```
In [55]: allData['Order Date'].dtype
```

```
Out[55]: dtype('O')
```

```
In [56]: allData['Hour']=pd.to_datetime(allData['Order Date']).dt.hour
```

In [57]:

```
allData.head()
```

Out[57]:

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

In [67]:

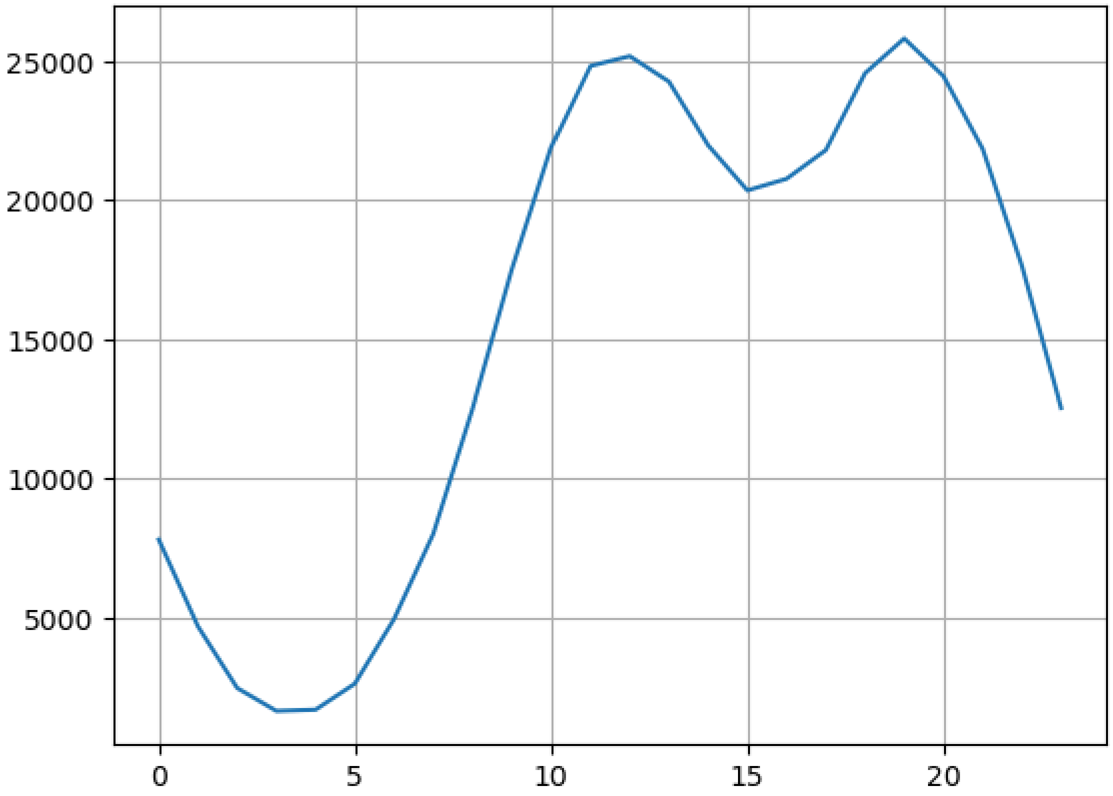
```
hour=[]
hour.sort()
for a in allData['Hour']:
    hour.append(a)
hour2=list(set(hour))
print(hour2)

cont=[]
for a in range(0,24):
    p=hour.count(hour2[a])
    cont.append(p)
print(cont)
```

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20,
21, 22, 23]
[7820, 4700, 2486, 1662, 1708, 2642, 4964, 8022, 12512, 17496, 21888, 2482
2, 25174, 24258, 21968, 20350, 20768, 21798, 24560, 25810, 24456, 21842, 1
7644, 12550]
```

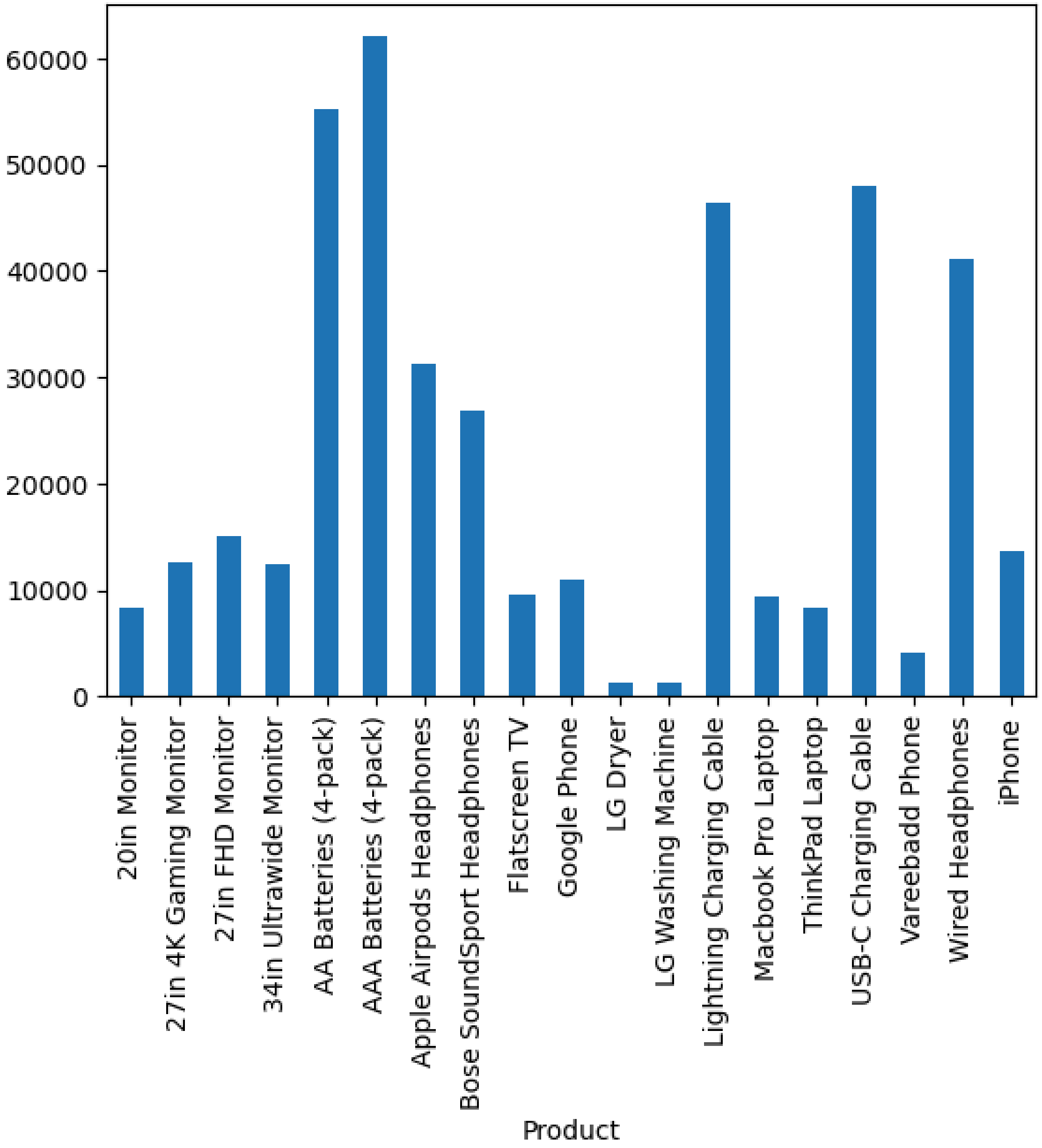
In [69]:

```
plt.grid()
plt.plot(hour2,cont)
plt.show()
```



Which product has the hightest demand and why?

```
In [70]: allData.groupby('Product')['Quantity Ordered'].sum().plot.bar()
         plt.show()
```



In [71]:

```
allData.groupby('Product')['Quantity Ordered'].mean()
```

Out[71]:

```
Product
20in Monitor                  1.006828
27in 4K Gaming Monitor        1.002247
27in FHD Monitor              1.005728
34in Ultrawide Monitor        1.002912
AA Batteries (4-pack)         1.343004
AAA Batteries (4-pack)        1.502689
Apple Airpods Headphones      1.007203
Bose SoundSport Headphones    1.009906
Flatscreen TV                 1.003958
Google Phone                  1.001267
LG Dryer                      1.000000
LG Washing Machine            1.000000
Lightning Charging Cable      1.071983
Macbook Pro Laptop            1.000847
ThinkPad Laptop               1.000484
USB-C Charging Cable          1.094599
Vareebadd Phone               1.001453
Wired Headphones              1.088709
iPhone                        1.001023
Name: Quantity Ordered, dtype: float64
```

In [72]:

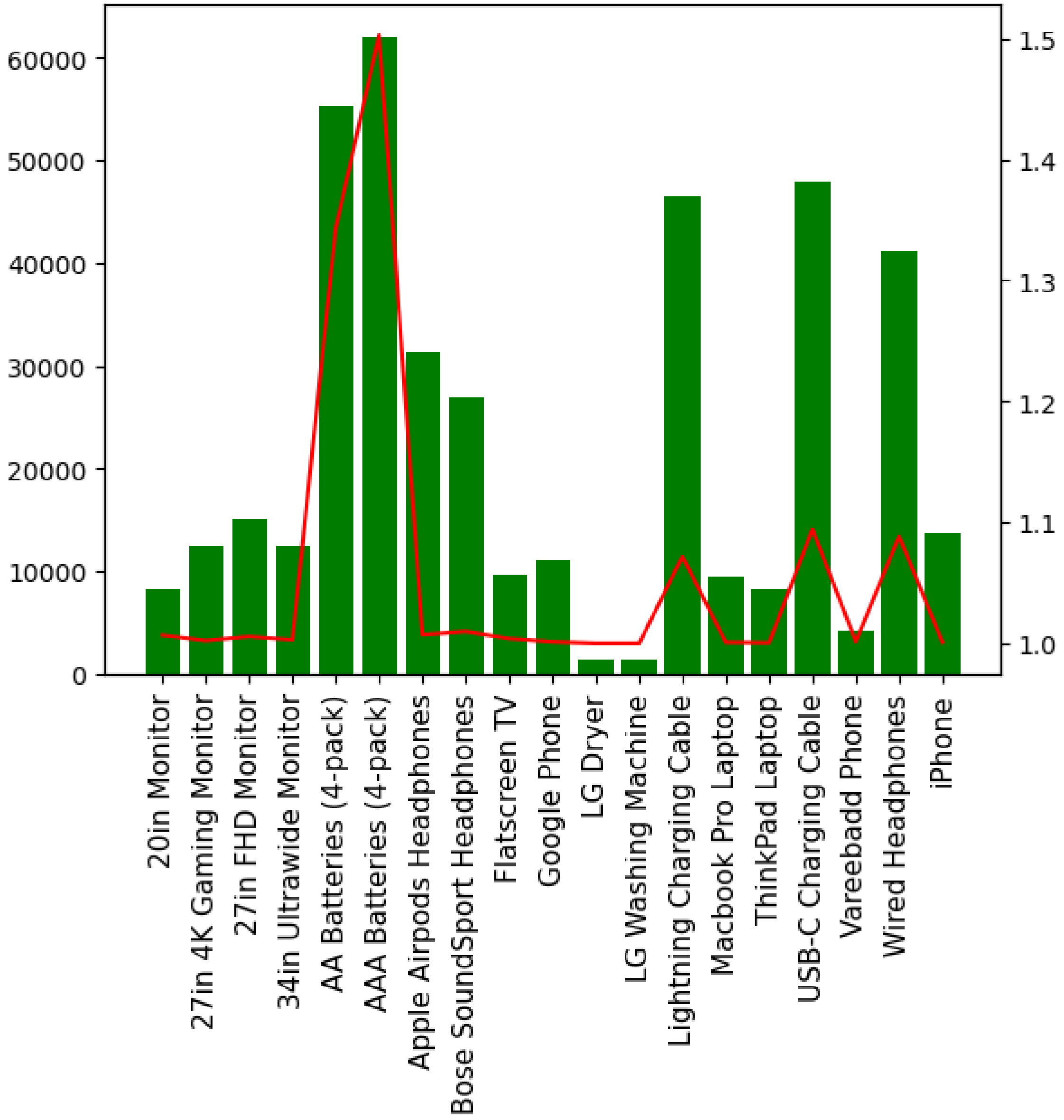
```
product=allData.groupby('Product')['Quantity Ordered'].sum().index
quantity=allData.groupby('Product')['Quantity Ordered'].sum()
prices=allData.groupby('Product')['Quantity Ordered'].mean()
```

In [78]:

```
plt.figure(figsize=(55,30))
fig,ax1=plt.subplots()
ax2=ax1.twinx()
ax1.bar(product,quantity,color='g')
ax2.plot(product,prices,'r')
ax1.set_xticklabels(product,rotation="vertical",size=11)
plt.show()
```

```
C:\Users\Sachin sirohi\AppData\Local\Temp\ipykernel_3424\3917295062.py:6:
UserWarning: FixedFormatter should only be used together with FixedLocator
  ax1.set_xticklabels(product,rotation="vertical",size=11)
```

```
<Figure size 5500x3000 with 0 Axes>
```



In [ ]: