In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import os
```

In [2]:

```
path = r'D:\Career\Udemy\DA\Commerce Data Analysis\Sales_Data'
```

In [3]:

```
files = [file for file in os.listdir(path)]
for file in files:
    print(file)
```

```
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 [4]:

```
all_data = pd.DataFrame()
for file in files:
    current_df = pd.read_csv(path+'/'+file)
    all_data = pd.concat([all_data,current_df])

all_data.head()
```

Out[4]:

	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 [5]:

```
all_data.isnull().sum()
```

Out[5]:

```
Order ID            545
Product             545
Quantity Ordered    545
Price Each          545
Order Date          545
Purchase Address    545
dtype: int64
```

In [6]:

```
all_data.dropna(inplace = True)
```

Monthly sales

In [7]:

```
def month(x):
    return x.split('/')[0]
```

In [8]:

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

In [9]:

```
filter = all_data['Month'] == 'Order Date'
```

In [10]:

```
all_data = all_data[~filter]
all_data.head()
```

Out[10]:

	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 [11]:

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

In [12]:

```
all_data['Quantity Ordered'] = all_data['Quantity Ordered'].astype(int)
```

In [13]:

```
all_data['Price Each'] = all_data['Price Each'].astype(float)
```

In [14]:

```
all_data['Price'] = all_data['Quantity Ordered'] * all_data['Price Each']
```

In [15]:

```
all_data.head()
```

Out[15]:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Price
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 [16]:

```
all_data.groupby('Month')['Price'].sum()
```

Out[16]:

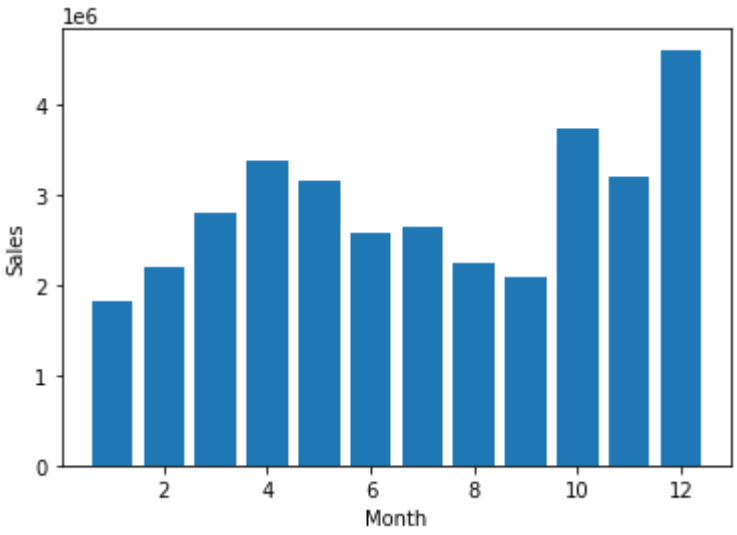
```
Month
1     1.822257e+06
2     2.202022e+06
3     2.807100e+06
4     3.390670e+06
5     3.152607e+06
6     2.577802e+06
7     2.647776e+06
8     2.244468e+06
9     2.097560e+06
10    3.736727e+06
11    3.199603e+06
12    4.613443e+06
Name: Price, dtype: float64
```

In [17]:

```
months = range(1,13)
plt.bar(months,all_data.groupby('Month')['Price'].sum())
plt.xlabel('Month')
plt.ylabel('Sales')
```

Out[17]:

```
Text(0, 0.5, 'Sales')
```



City has maximum order

In [18]:

```
def city(x):
    return x.split(',')[1]
```

In [19]:

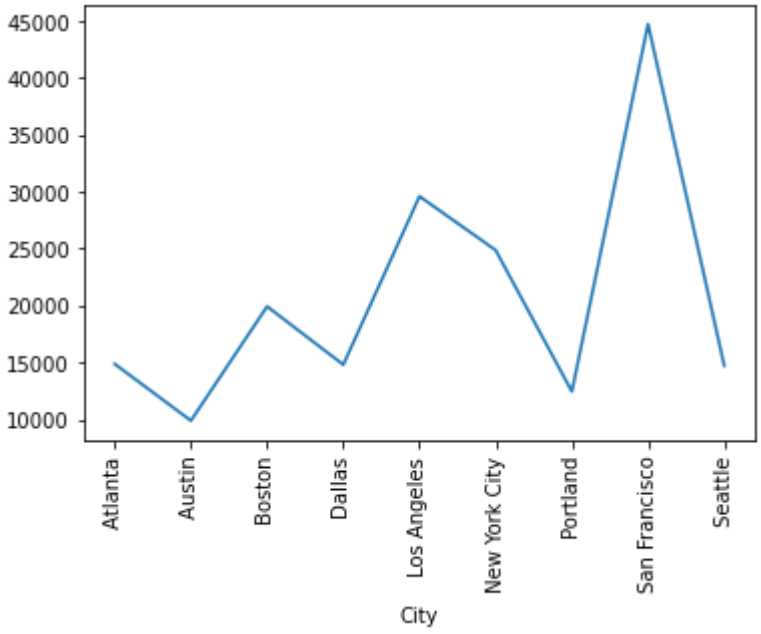
```
all_data['City'] = all_data['Purchase Address'].apply(city)
```

In [20]:

```
all_data.groupby('City')['City'].count().plot()
plt.xticks(rotation = 'vertical')
```

Out[20]:

```
(array([-1.,  0.,  1.,  2.,  3.,  4.,  5.,  6.,  7.,  8.,  9.]),
 [Text(-1.0, 0, ' Seattle'),
  Text(0.0, 0, ' Atlanta'),
  Text(1.0, 0, ' Austin'),
  Text(2.0, 0, ' Boston'),
  Text(3.0, 0, ' Dallas'),
  Text(4.0, 0, ' Los Angeles'),
  Text(5.0, 0, ' New York City'),
  Text(6.0, 0, ' Portland'),
  Text(7.0, 0, ' San Francisco'),
  Text(8.0, 0, ' Seattle'),
  Text(9.0, 0, '')])
```

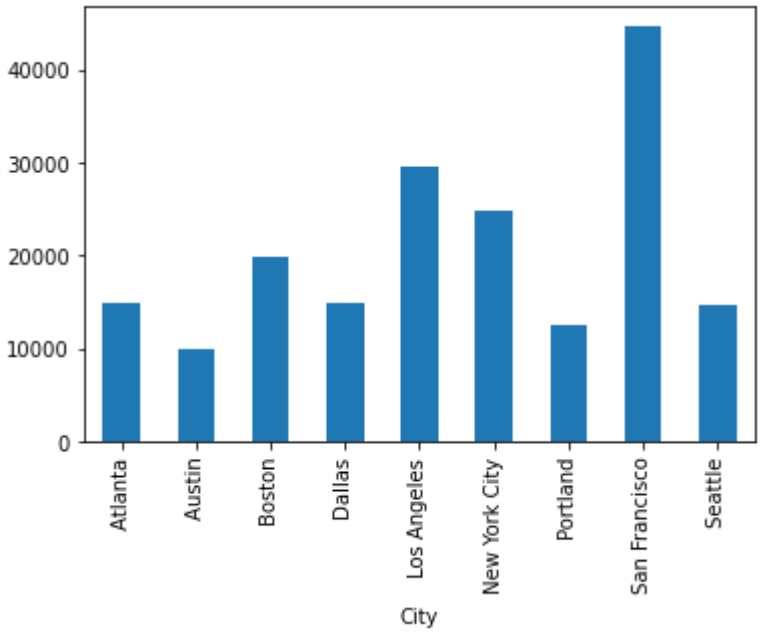


In [21]:

```
all_data.groupby('City')['City'].count().plot.bar()
```

Out[21]:

```
<AxesSubplot:xlabel='City'>
```



In [22]:

```
all_data['Hour'] = pd.to_datetime(all_data['Order Date']).dt.hour
```

In [23]:

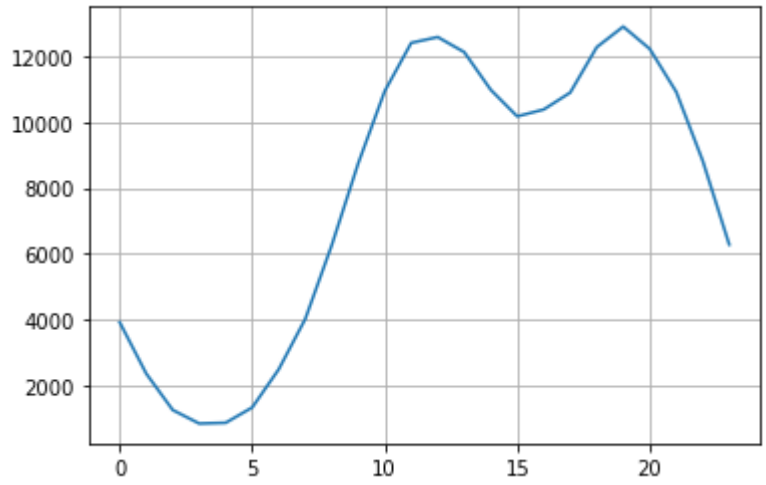
```
keys = []
hours = []
for key,hour_df in all_data.groupby('Hour'):
    keys.append(key)
    hours.append(len(hour_df))
```

In [24]:

```
plt.grid()
plt.plot(keys,hours)
```

Out[24]:

```
[<matplotlib.lines.Line2D at 0x1a1b2ac1b50>]
```



What product sold the most?

In [25]:

```
all_data.groupby('Product')['Quantity Ordered'].sum()
```

Out[25]:

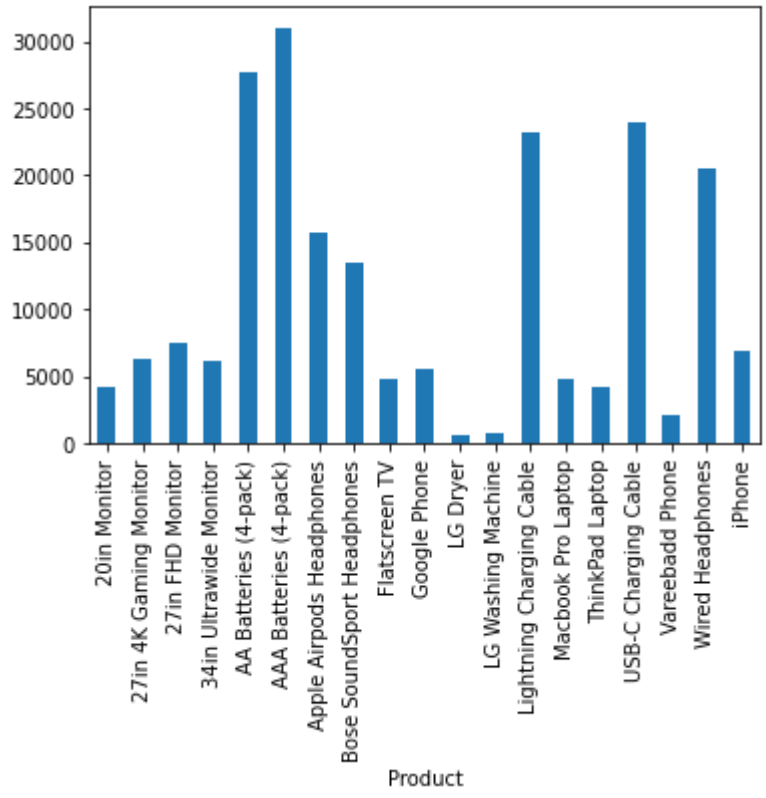
```
Product
20in Monitor                   4129
27in 4K Gaming Monitor         6244
27in FHD Monitor               7550
34in Ultrawide Monitor         6199
AA Batteries (4-pack)         27635
AAA Batteries (4-pack)        31017
Apple Airpods Headphones      15661
Bose SoundSport Headphones    13457
Flatscreen TV                  4819
Google Phone                   5532
LG Dryer                        646
LG Washing Machine              666
Lightning Charging Cable      23217
Macbook Pro Laptop             4728
ThinkPad Laptop                4130
USB-C Charging Cable          23975
Vareebadd Phone                2068
Wired Headphones              20557
iPhone                         6849
Name: Quantity Ordered, dtype: int32
```

In [26]:

```
all_data.groupby('Product')['Quantity Ordered'].sum().plot(kind = 'bar')
```

Out[26]:

```
<AxesSubplot:xlabel='Product'>
```

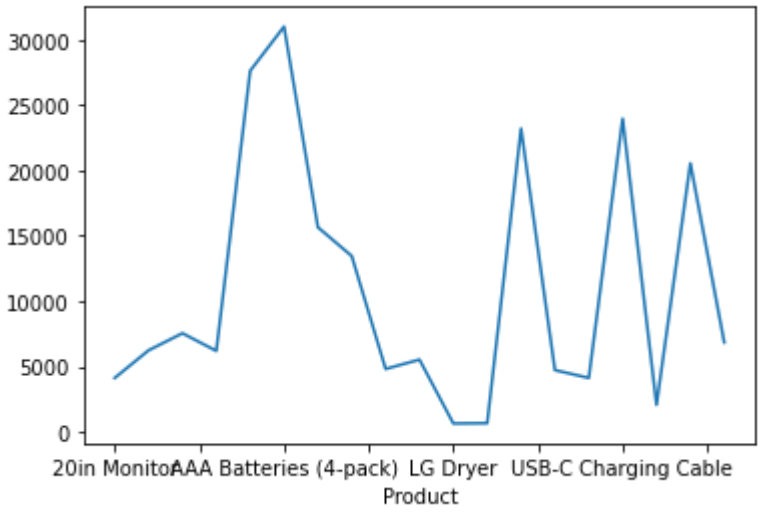


In [27]:

```
all_data.groupby('Product')['Quantity Ordered'].sum().plot(kind = 'line')
```

Out[27]:

```
<AxesSubplot:xlabel='Product'>
```



In [28]:

```
all_data.head()
```

Out[28]:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Price	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 [29]:

```
all_data.groupby('Product')['Price Each'].mean()
```

Out[29]:

```
Product
20in Monitor                  109.99
27in 4K Gaming Monitor        389.99
27in FHD Monitor              149.99
34in Ultrawide Monitor        379.99
AA Batteries (4-pack)           3.84
AAA Batteries (4-pack)          2.99
Apple Airpods Headphones      150.00
Bose SoundSport Headphones     99.99
Flatscreen TV                 300.00
Google Phone                  600.00
LG Dryer                      600.00
LG Washing Machine            600.00
Lightning Charging Cable       14.95
Macbook Pro Laptop           1700.00
ThinkPad Laptop               999.99
USB-C Charging Cable           11.95
Vareebadd Phone               400.00
Wired Headphones               11.99
iPhone                        700.00
Name: Price Each, dtype: float64
```

In [30]:

```
Product = all_data.groupby('Product')['Quantity Ordered'].sum().index
Quantity = all_data.groupby('Product')['Quantity Ordered'].sum()
Prices = all_data.groupby('Product')['Price Each'].mean()
```

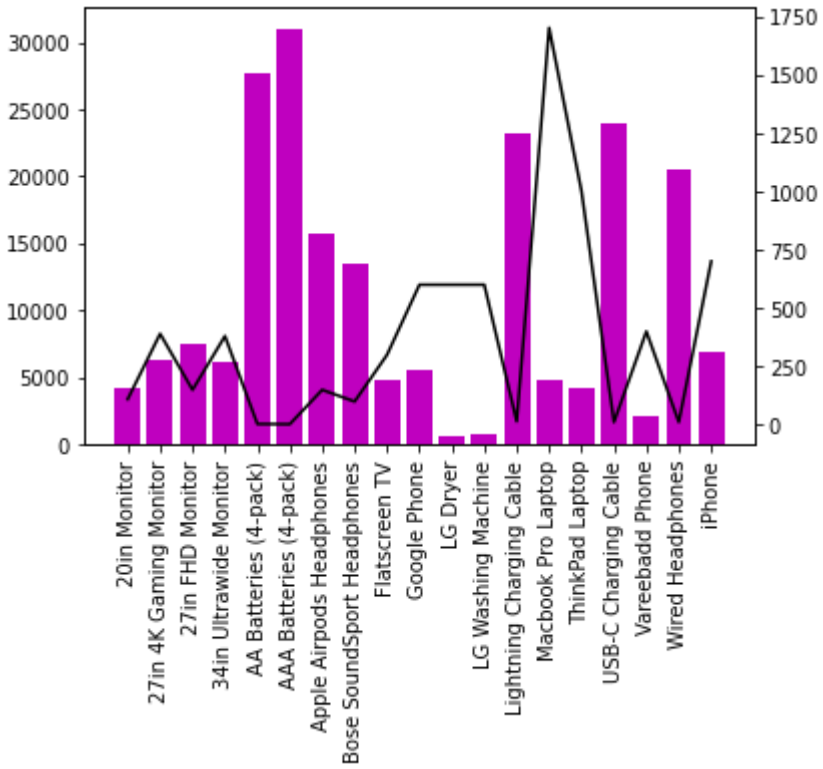
In [31]:

```
fig,ax1 = plt.subplots()
ax2 = ax1.twinx()
ax1.bar(Product,Quantity,color = 'm')
ax2.plot(Product,Prices,color = 'k')
ax1.set_xticklabels(Product, rotation = 'vertical',size = 10)
```

```
<ipython-input-31-4083e7a9a470>:5: UserWarning: FixedFormatter should only b
e used together with FixedLocator
  ax1.set_xticklabels(Product, rotation = 'vertical',size = 10)
```

Out[31]:

```
[Text(0, 0, '20in Monitor'),
 Text(1, 0, '27in 4K Gaming Monitor'),
 Text(2, 0, '27in FHD Monitor'),
 Text(3, 0, '34in Ultrawide Monitor'),
 Text(4, 0, 'AA Batteries (4-pack)'),
 Text(5, 0, 'AAA Batteries (4-pack)'),
 Text(6, 0, 'Apple Airpods Headphones'),
 Text(7, 0, 'Bose SoundSport Headphones'),
 Text(8, 0, 'Flatscreen TV'),
 Text(9, 0, 'Google Phone'),
 Text(10, 0, 'LG Dryer'),
 Text(11, 0, 'LG Washing Machine'),
 Text(12, 0, 'Lightning Charging Cable'),
 Text(13, 0, 'Macbook Pro Laptop'),
 Text(14, 0, 'ThinkPad Laptop'),
 Text(15, 0, 'USB-C Charging Cable'),
 Text(16, 0, 'Vareebadd Phone'),
 Text(17, 0, 'Wired Headphones'),
 Text(18, 0, 'iPhone')]
```



what products are more often sold together?

In [32]:

```
df = all_data['Order ID'].duplicated(keep = False)
df2 = all_data[df]
df2.head()
```

Out[32]:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Price	City	Hour
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
18	176574	Google Phone	1	600.00	04/03/19 19:42	20 Hill St, Los Angeles, CA 90001	4	600.00	Los Angeles	19
19	176574	USB-C Charging Cable	1	11.95	04/03/19 19:42	20 Hill St, Los Angeles, CA 90001	4	11.95	Los Angeles	19
30	176585	Bose SoundSport Headphones	1	99.99	04/07/19 11:31	823 Highland St, Boston, MA 02215	4	99.99	Boston	11

In [33]:

```
df2['Grouped'] = df2.groupby('Order ID')['Product'].transform(lambda x : ','.join(x))
```

```
<ipython-input-33-79186fd4113c>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/
stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pand
as.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-v
ersus-a-copy)
  df2['Grouped'] = df2.groupby('Order ID')['Product'].transform(lambda x :
 ','.join(x))
```

In [34]:

```
df2.head()
```

Out[34]:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Price	City	Hour
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
18	176574	Google Phone	1	600.00	04/03/19 19:42	20 Hill St, Los Angeles, CA 90001	4	600.00	Los Angeles	19
19	176574	USB-C Charging Cable	1	11.95	04/03/19 19:42	20 Hill St, Los Angeles, CA 90001	4	11.95	Los Angeles	19
30	176585	Bose SoundSport Headphones	1	99.99	04/07/19 11:31	823 Highland St, Boston, MA 02215	4	99.99	Boston	11

In [35]:

```
df2 = df2.drop_duplicates(subset = 'Order ID')
```

In [36]:

```
df2.head()
```

Out[36]:

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Price	City	Hour
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
18	176574	Google Phone	1	600.00	04/03/19 19:42	20 Hill St, Los Angeles, CA 90001	4	600.00	Los Angeles	19
30	176585	Bose SoundSport Headphones	1	99.99	04/07/19 11:31	823 Highland St, Boston, MA 02215	4	99.99	Boston	11
32	176586	AAA Batteries (4-pack)	2	2.99	04/10/19 17:00	365 Center St, San Francisco, CA 94016	4	5.98	San Francisco	17
119	176672	Lightning Charging Cable	1	14.95	04/12/19 11:07	778 Maple St, New York City, NY 10001	4	14.95	New York City	11

In [37]:

```
df2['Grouped'].value_counts()[0:5].plot(kind = 'pie')
```

Out[37]:

```
<AxesSubplot:ylabel='Grouped'>
```