exercise-9

November 4, 2024

1 1. Use the given xlsv file for this exercise

2 2. Read file using pandas

[1]:
```
import pandas as pd

from google.colab import files
uploaded = files.upload()
```

```
<IPython.core.display.HTML object>
```

```
Saving Technological-Products-Sample-Datas.csv to Technological-Products-Sample-
Datas.csv
```

[2]:
```
data = pd.read_csv("Technological-Products-Sample-Datas.csv")
dataFrame = pd.DataFrame(data)
dataFrame
```

[2]:

	Brand	Device	Model	Country of Origin	Date of Release	\
0	Apple	iPhone	13 Pro Max	United States	9/24/2021	
1	Samsung	Galaxy	S21 Ultra	South Korea	1/29/2021	
2	Google	Pixel	6 Pro	United States	10/19/2021	
3	Sony	PlayStation	5	Japan	11/12/2020	
4	Microsoft	Surface	Laptop 4	United States	4/15/2021	
5	Dell	XPS	13	United States	1/28/2021	
6	HP	Spectre	x360	United States	4/16/2021	
7	Lenovo	ThinkPad	X1 Carbon	China	2/15/2021	
8	Asus	ROG	Zephyrus G14	Taiwan	3/15/2021	
9	Acer	Predator	Helios 300	Taiwan	2/17/2021	
10	Apple	MacBook	Pro 14-inch	United States	10/26/2021	
11	Samsung	Odyssey	G9	South Korea	7/22/2020	
12	Google	Pixelbook	Go	United States	6/17/2021	
13	Sony	Xperia	1 III	Japan	8/19/2021	
14	Microsoft	Xbox	Series X	United States	11/10/2020	
15	Dell	Alienware	m15 R5	United States	4/20/2021	
16	HP	Pavilion	x360	United States	5/14/2021	
17	Lenovo	IdeaPad	5 Pro	China	3/10/2021	
18	Asus	ZenBook	14	Taiwan	1/15/2021	

19	Acer	Swift	3	Taiwan	2/10/2021
20	Apple	iPad	Pro 12.9-inch	United States	5/21/2021
21	Samsung	Galaxy	Tab S7+	South Korea	8/21/2020
22	Google	Nest	Hub Max	United States	3/30/2021
23	Sony	WH	1000XM4	Malaysia	8/6/2020
24	Microsoft	Surface	Pro 8	United States	10/5/2021
25	Dell	UltraSharp	U2720Q	China	12/25/2020
26	HP	Elite	Dragonfly	United States	6/28/2021
27	Lenovo	Legion	7i	China	4/5/2021
28	Asus	TUF	Gaming A15	Taiwan	3/20/2021
29	Acer	Aspire	5	Taiwan	1/30/2021

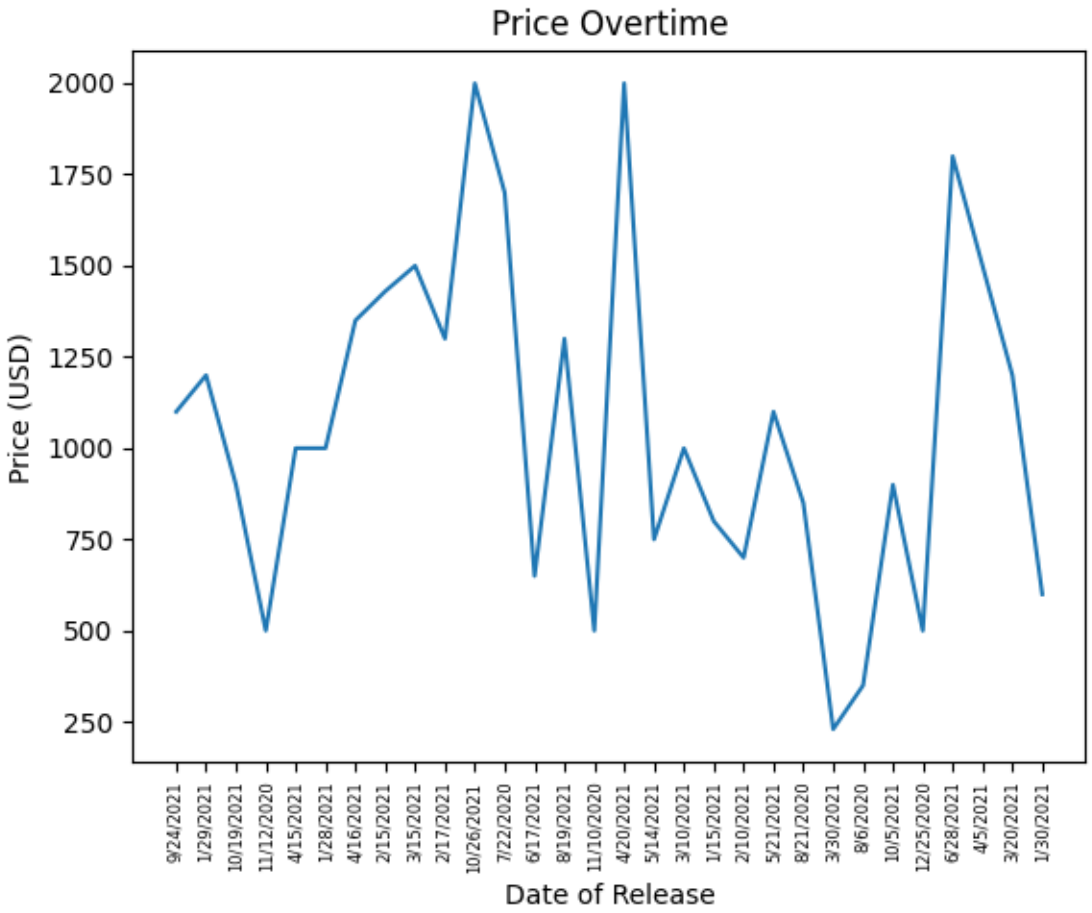
	Price (USD)
0	1099
1	1199
2	899
3	499
4	999
5	999
6	1349
7	1429
8	1499
9	1299
10	1999
11	1699
12	649
13	1299
14	499
15	1999
16	749
17	999
18	799
19	699
20	1099
21	849
22	229
23	349
24	899
25	499
26	1799
27	1499
28	1199
29	599

3 3. Analyze the given data and generate the appropriate plot including its properties

[3]:
```
import matplotlib.pyplot as plt
```

a. Read each device's date of release and sales. Show it using a line plot.

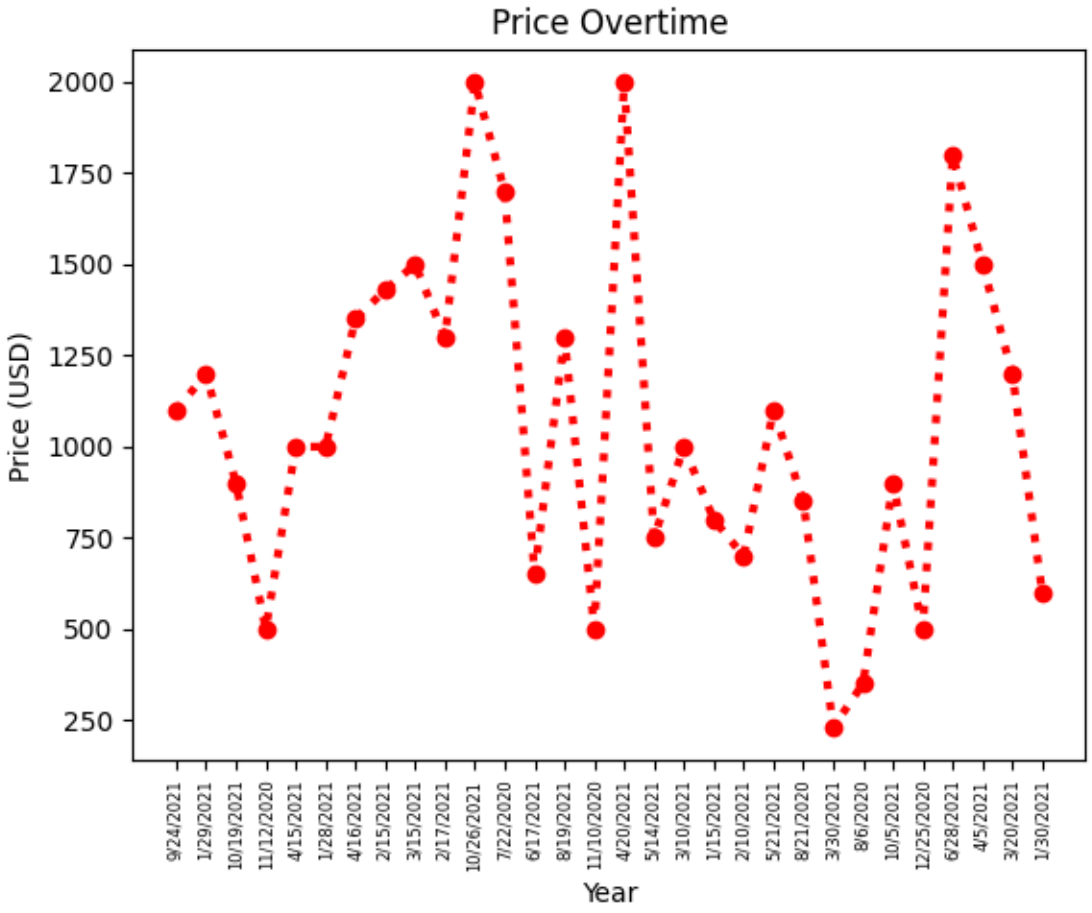
[4]:
```
x = dataFrame["Date of Release"]
y = dataFrame["Price (USD)"]

plt.plot(x, y)
plt.xticks(fontsize=6, rotation='vertical')
plt.title("Price Overtime")
plt.xlabel("Date of Release")
plt.ylabel("Price (USD)")
plt.show()
```



b. Get the total sales and show the plot with the following style properties.

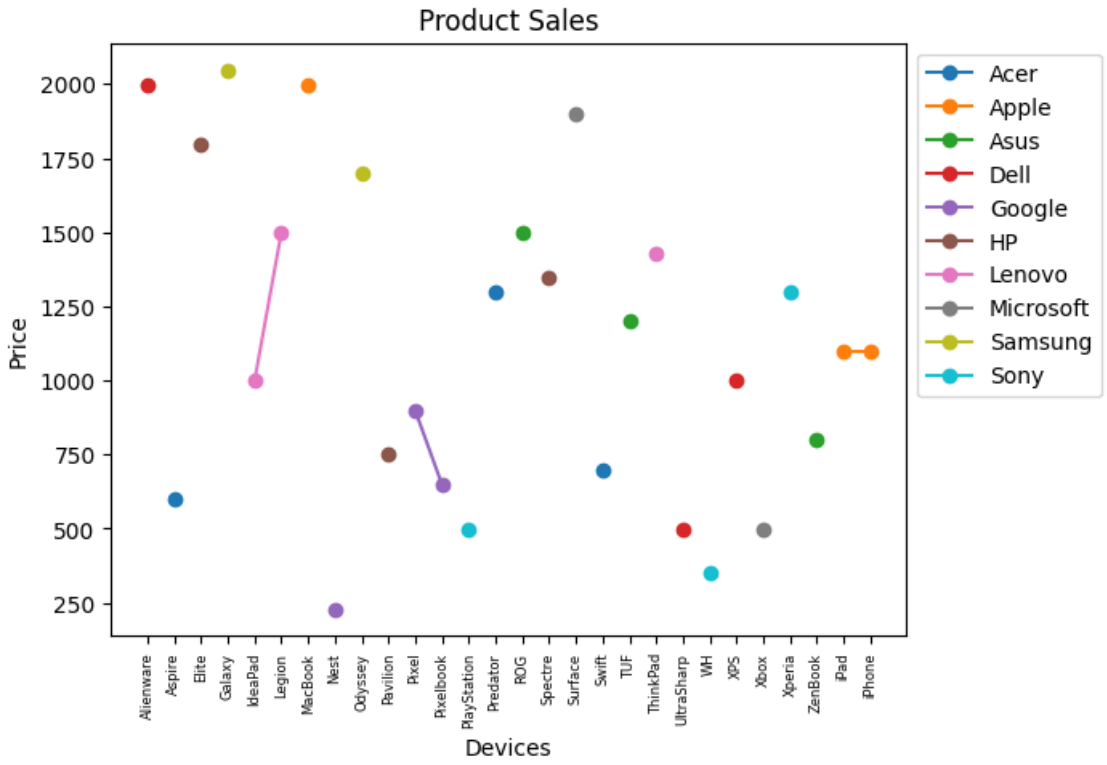
[8]:
```
x = dataFrame["Date of Release"]
y = dataFrame["Price (USD)"]

plt.plot(x, y, 'o:b', color ='red', marker='o', mfc='red',linewidth=3)
plt.xticks(fontsize=6, rotation='vertical')
plt.title("Price Overtime")
plt.xlabel("Year")
plt.ylabel("Price (USD)")
plt.show()
```

```
<ipython-input-8-ce95a204a41c>:4: UserWarning: marker is redundantly defined by
the 'marker' keyword argument and the fmt string "o:b" (-> marker='o'). The
keyword argument will take precedence.
  plt.plot(x, y, 'o:b', color ='red', marker='o', mfc='red',linewidth=3)
<ipython-input-8-ce95a204a41c>:4: UserWarning: color is redundantly defined by
the 'color' keyword argument and the fmt string "o:b" (-> color='b'). The
keyword argument will take precedence.
  plt.plot(x, y, 'o:b', color ='red', marker='o', mfc='red',linewidth=3)
```



c. Read all product sales data and show it using multiline plot.

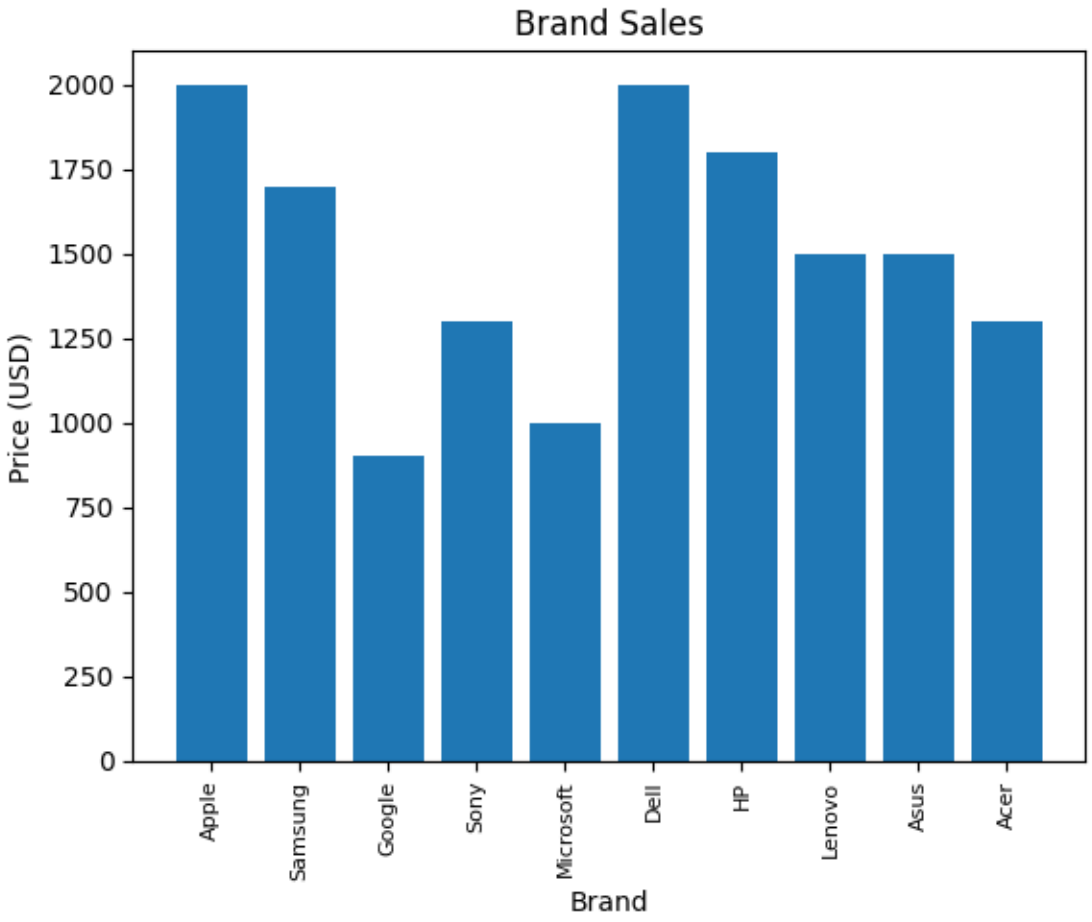
[28]:
```
sales = dataFrame.groupby(['Device', 'Brand'])['Price (USD)'].sum().unstack()

for brand in sales.columns:
    plt.plot(sales.index, sales[brand], marker='o', label=brand)

plt.legend(bbox_to_anchor=(1,1))
plt.title("Product Sales")
plt.xlabel("Devices")
plt.xticks(fontsize=6, rotation='vertical')
plt.ylabel("Price")
plt.show()
```



d. Read all brand sales data and show it using bar graph

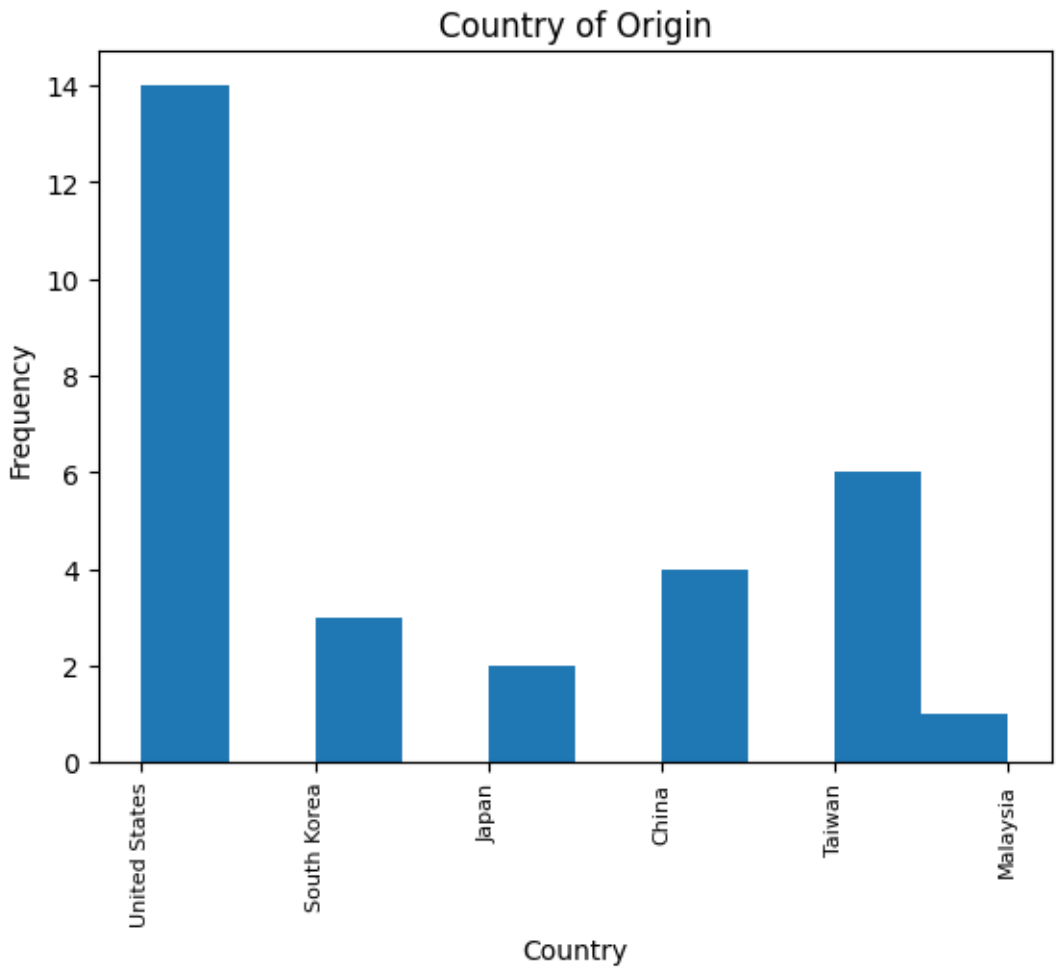
[29]:
```
xbar = dataFrame["Brand"]
ybar = dataFrame["Price (USD)"]

plt.bar(xbar, ybar)
plt.xticks(fontsize=8, rotation='vertical')
plt.title("Brand Sales")
plt.xlabel("Brand")
```

```
plt.ylabel("Price (USD)")
plt.show()
```



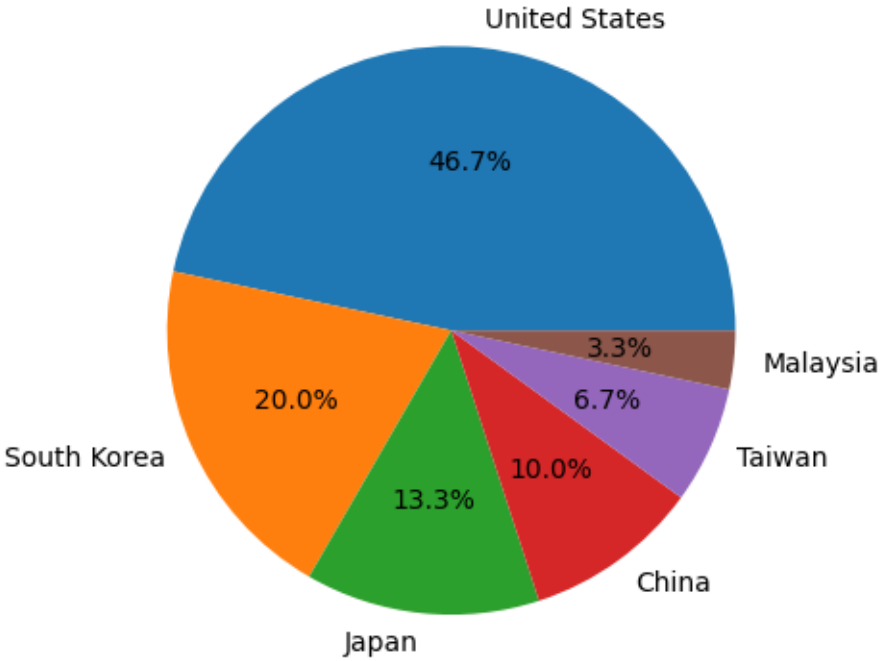
e. Read each device and show it using histogram to see the most common country of origin.

[30]:
```
xhist = dataFrame["Country of Origin"]

plt.hist(xhist)
plt.xticks(fontsize=8, rotation='vertical')
plt.title("Country of Origin")
plt.xlabel("Country")
plt.ylabel("Frequency")
plt.show()
```



f. Create a pie chart that shows each brand's Country of Origin.

[31]:
```
labelpie = dataFrame["Country of Origin"].unique()
xpie = dataFrame["Country of Origin"].value_counts()

plt.pie(xpie, labels=labelpie, autopct='%1.1f%%')
plt.show()
```



4 4. Save your file to matplotlib(surname).pdf