

Name: James Russel E. Adia
Section: CPE21S4

Course: CPE 031

EXERCISE NO. 9 - MATPLOTLIB

Instructions:

✓ Python Matplotlib Exercise (100 points)

This Matplotlib exercise project helps Python developers learn and practice data visualization using Matplotlib by solving multiple questions and problems.

In this exercise, we are using Pandas and Matplotlib to visualize Technological Products Data.

1. Use the following csv file for this exercise.
2. Read this file using Pandas or NumPy or using in-built matplotlib function.
3. Analyze the data and generate the appropriate plot including its properties:
 - a. Read each Product's date of release and sales. Show it using a line plot.
 - b. Get the total sales and show line plot with the following style properties.
 - Line Style dotted and Line-color should be red
 - Show legend at the lower right location.
 - X label name = Year
 - Y label name = Sales
 - Add a circle marker.
 - Line marker color as read
 - Line width should be 3
 - c. Read all Product sales data and show it using a multiline plot. Label them.
 - d. Read each brand sales data and show it using the bar chart. Label them.
 - e. Read each Product and show it using the histogram to see the most common Country of Origin. Label them.
 - f. Create a pie chart that shows each brand Country of Origin. Label them.
4. Save your file into Matplotlib-«StudentName>.pdf

- Installing the libraries

```
!pip install matplotlib
!pip install pandas
```

```
Requirement already satisfied: matplotlib in /usr/local/lib/python3.10/dist-packages (3.8.0)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.3.0)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (4.54.1)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.4.7)
Requirement already satisfied: numpy<2,>=1.21 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (1.26.4)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (24.1)
Requirement already satisfied: pillow>=6.2.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (10.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (3.2.0)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib) (2.8.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (2.2.2)
Requirement already satisfied: numpy>=1.22.4 in /usr/local/lib/python3.10/dist-packages (from pandas) (1.26.4)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.10/dist-packages (from pandas) (2024.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
```

- Mounting the Google Drive to Import .csv file

```
# Mounting google drive and read .csv file
# importing pandas and matplotlib library
import matplotlib.pyplot as plt
import pandas as pd
from google.colab import drive
df = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/CPE 031 -
Visualization and Data Analysis/Matplotlib -
Adia/Technological-Products-Sample-Data - Tech Products.csv')
df
```

| | Brand | Device | Model | Country of Origin | Date of Release | Sales (USD) | |
|----|-----------|-------------|---------------|-------------------|-----------------|-------------|--|
| 0 | Apple | iPhone | 13 Pro Max | United States | 9/24/2021 | 1099 | |
| 1 | Samsung | Galaxy | S21 Ultra | South Korea | 1/29/2021 | 1199 | |
| 2 | Google | Pixel | 6 Pro | United States | 10/19/2021 | 899 | |
| 3 | Sony | PlayStation | 5 | Japan | 11/12/2020 | 499 | |
| 4 | Microsoft | Surface | Laptop 4 | United States | 4/15/2021 | 999 | |
| 5 | Dell | XPS | 13 | United States | 1/28/2021 | 999 | |
| 6 | HP | Spectre | x360 | United States | 4/16/2021 | 1349 | |
| 7 | Lenovo | ThinkPad | X1 Carbon | China | 2/15/2021 | 1429 | |
| 8 | Asus | ROG | Zephyrus G14 | Taiwan | 3/15/2021 | 1499 | |
| 9 | Acer | Predator | Helios 300 | Taiwan | 2/17/2021 | 1299 | |
| 10 | Apple | MacBook | Pro 14-inch | United States | 10/26/2021 | 1999 | |
| 11 | Samsung | Odyssey | G9 | South Korea | 7/22/2020 | 1699 | |
| 12 | Google | Pixelbook | Go | United States | 6/17/2021 | 649 | |
| 13 | Sony | Xperia | 1 III | Japan | 8/19/2021 | 1299 | |
| 14 | Microsoft | Xbox | Series X | United States | 11/10/2020 | 499 | |
| 15 | Dell | Alienware | m15 R5 | United States | 4/20/2021 | 1999 | |
| 16 | HP | Pavilion | x360 | United States | 5/14/2021 | 749 | |
| 17 | Lenovo | IdeaPad | 5 Pro | China | 3/10/2021 | 999 | |
| 18 | Asus | ZenBook | 14 | Taiwan | 1/15/2021 | 799 | |
| 19 | Acer | Swift | 3 | Taiwan | 2/10/2021 | 699 | |
| 20 | Apple | iPad | Pro 12.9-inch | United States | 5/21/2021 | 1099 | |
| 21 | Samsung | Galaxy | Tab S7+ | South Korea | 8/21/2020 | 849 | |
| 22 | Google | Nest | Hub Max | United States | 3/30/2021 | 229 | |
| 23 | Sony | WH | 1000XM4 | Malaysia | 8/6/2020 | 349 | |
| 24 | Microsoft | Surface | Pro 8 | United States | 10/5/2021 | 899 | |
| 25 | Dell | UltraSharp | U2720Q | China | 12/25/2020 | 499 | |
| 26 | HP | Elite | Dragonfly | United States | 6/28/2021 | 1799 | |
| 27 | Lenovo | Legion | 7i | China | 4/5/2021 | 1499 | |
| 28 | Asus | TUF | Gaming A15 | Taiwan | 3/20/2021 | 1199 | |
| 29 | Acer | Aspire | 5 | Taiwan | 1/30/2021 | 599 | |

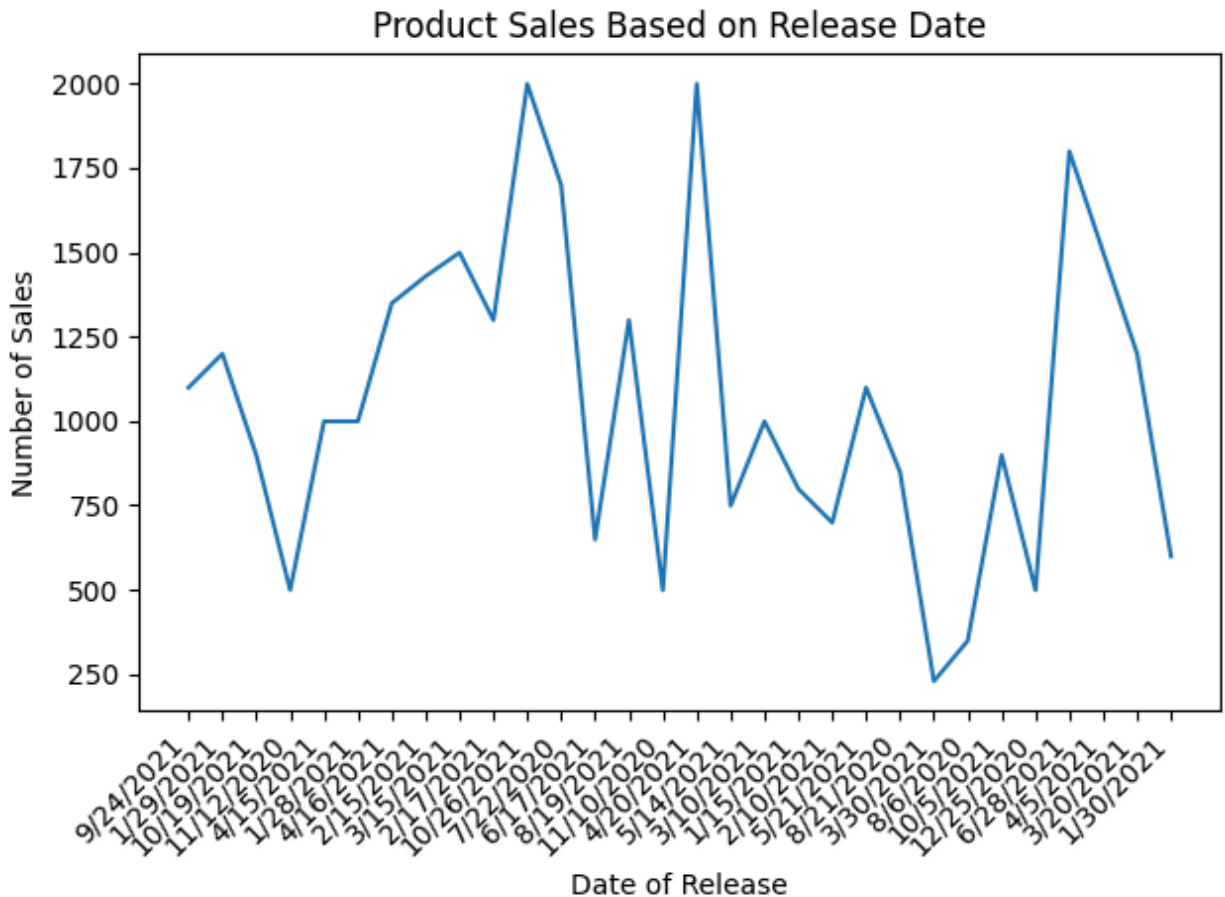
✓ a. Read each Product's date of release and sales. Show it using a line plot.

```
# Create line plot
x = df['Date of Release']
y = df['Sales (USD)']
plt.plot(x, y)

# Set labels and title
plt.xlabel('Date of Release')
plt.ylabel('Number of Sales')
plt.title('Product Sales Based on Release Date')

# Rotate x-axis labels for better visibility
plt.xticks(rotation=45, ha='right')
plt.tight_layout()

# Show the plot
plt.show()
```



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

- Line Style dotted and Line-color should be red
- Show legend at the lower right location.
- X label name = Year
- Y label name = Sales
- Add a circle marker.
- Line marker color as read
- Line width should be 3

```
# Extract date of release from the .csv file
df['Date of Release']

# Extract the year from 'Date of Release'
df['Year'] = pd.to_datetime(df['Date of Release']).dt.year

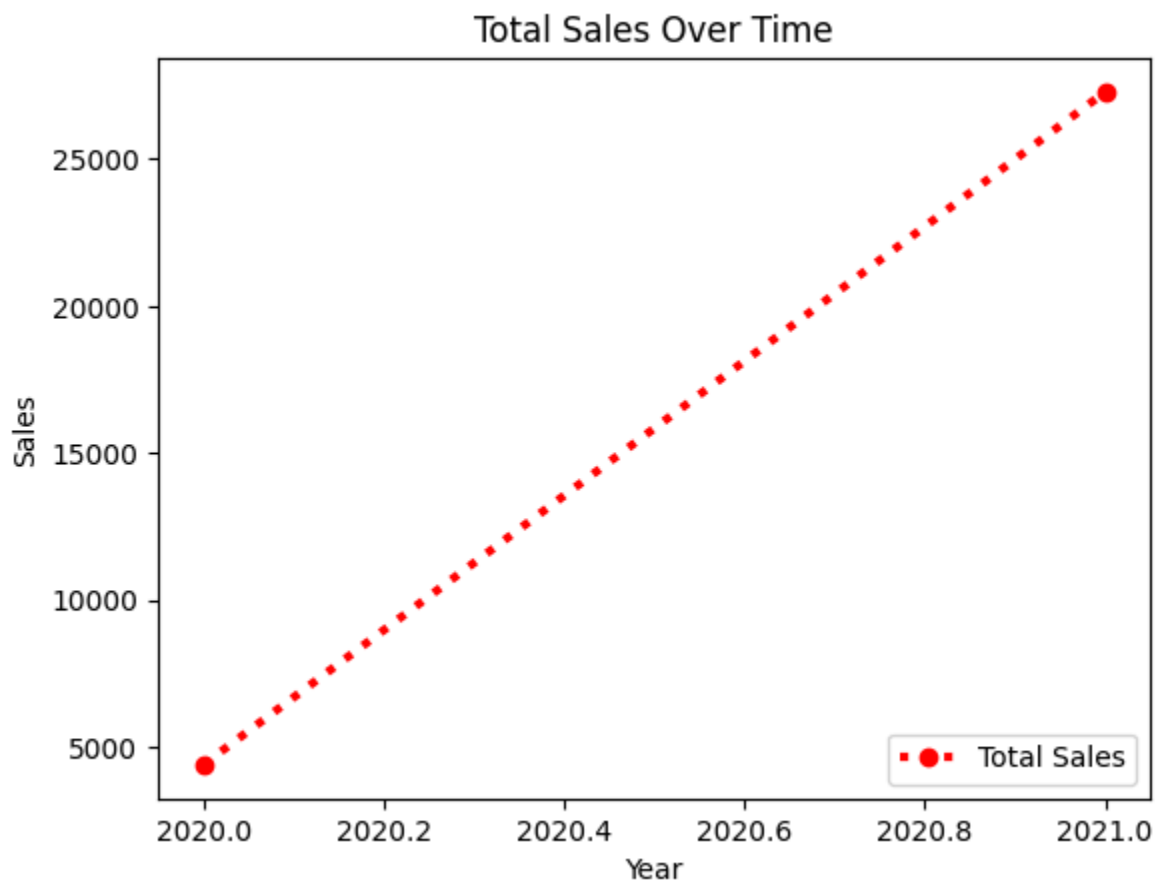
# Group by year and sum sales
total_sales_by_year = df.groupby('Year')['Sales
(USD)'].sum().reset_index()
```

```
# Create the line plot
plt.plot(total_sales_by_year['Year'], total_sales_by_year['Sales (USD)'],
         linestyle='dotted', color='red', marker='o',
         markerfacecolor='red', linewidth=3, label='Total Sales')

# Set labels and title
plt.xlabel('Year')
plt.ylabel('Sales')
plt.title('Total Sales Over Time')

# Show legend at the lower right
plt.legend(loc='lower right')

# Show the plot
plt.show()
```



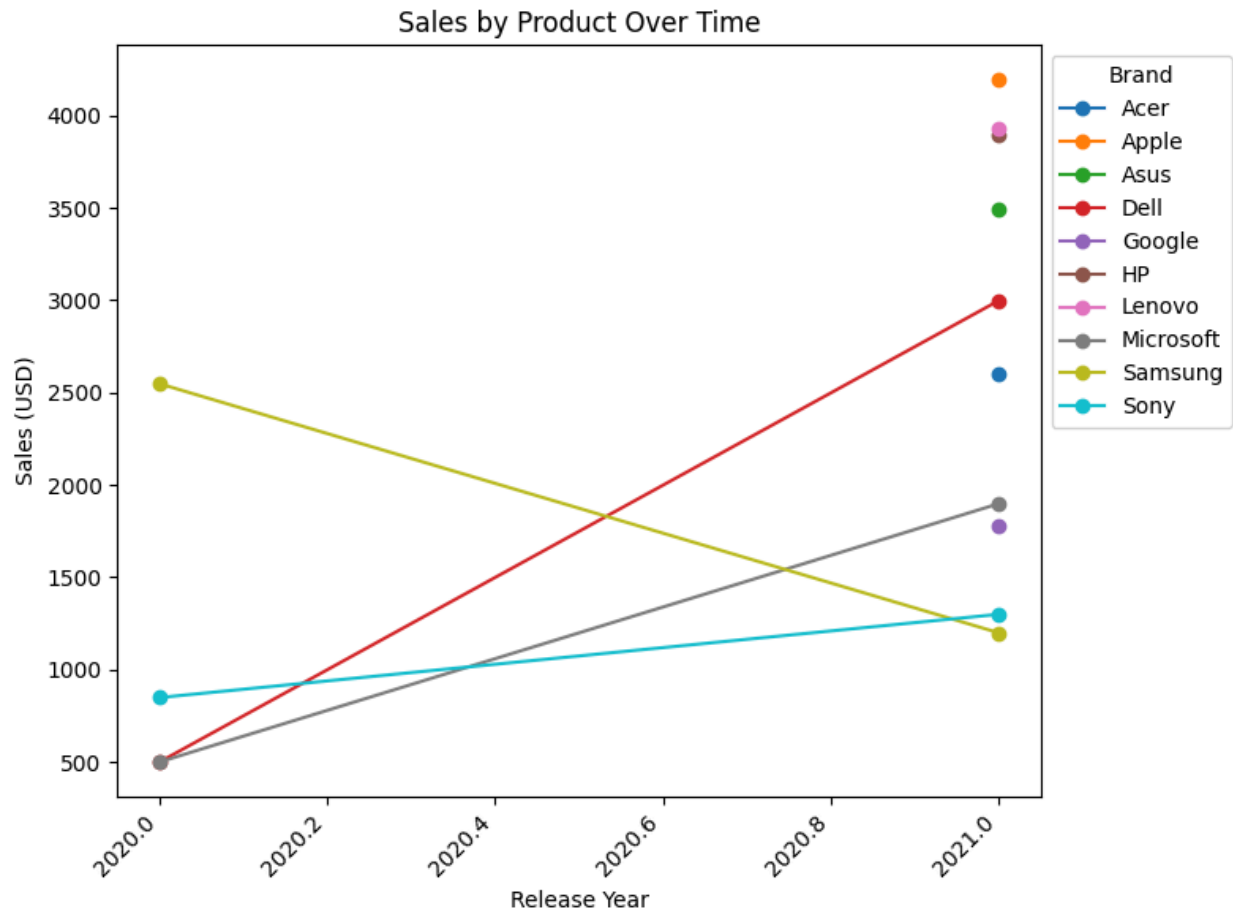
✓ c. Read all Product sales data and show it using a multiline plot. Label them.

```
# Group data and sum sales, unstack for plotting
sales_by_year = df.groupby(['Brand', 'Release Year'])['Sales
(USD)'].sum().unstack(level=0)

# Create the multiline plot from the grouped data
sales_by_year.plot(kind='line', figsize=(8, 6), marker='o')

# Customize the plot
plt.xlabel('Release Year')
plt.ylabel('Sales (USD)')
plt.title('Sales by Product Over Time')
plt.legend(title='Brand', bbox_to_anchor=(1, 1))
plt.xticks(rotation=45, ha='right')
plt.tight_layout()

# Show the plot
plt.show()
```



▼ d. Read each brand sales data and show it using the bar chart. Label them.

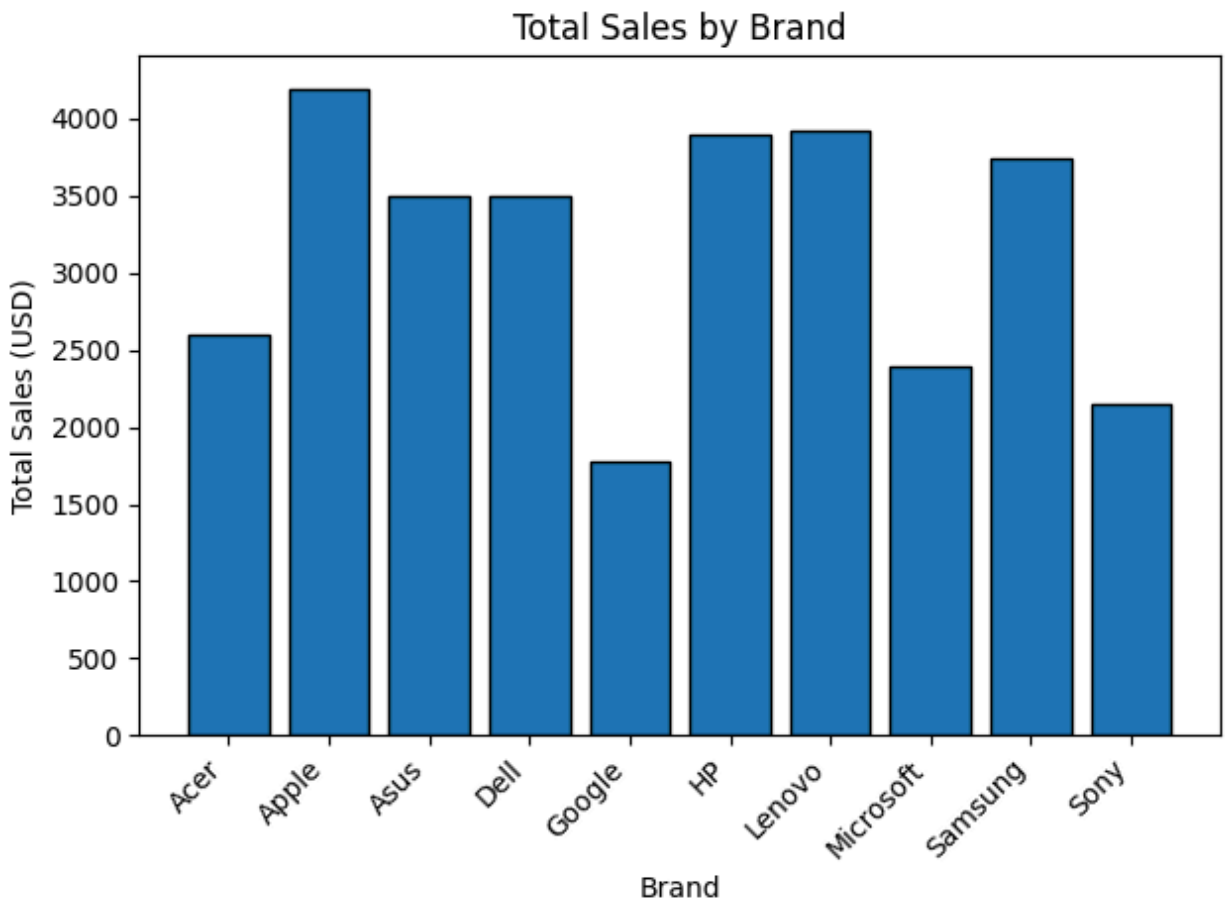
```
# Group data by brand and sum sales
brand_sales = df.groupby('Brand')['Sales (USD)'].sum().reset_index()

# Create the bar chart
plt.bar(brand_sales['Brand'], brand_sales['Sales (USD)'], edgecolor =
'black')

# Showing the labels
plt.xlabel('Brand')
plt.ylabel('Total Sales (USD)')
plt.title('Total Sales by Brand')
plt.xticks(rotation=45, ha='right') # Rotate x-axis labels for better
visibility
plt.tight_layout()
```



```
# Show the chart
plt.show()
```



e. Read each Product and show it using the histogram to see the most common Country of Origin. Label them.

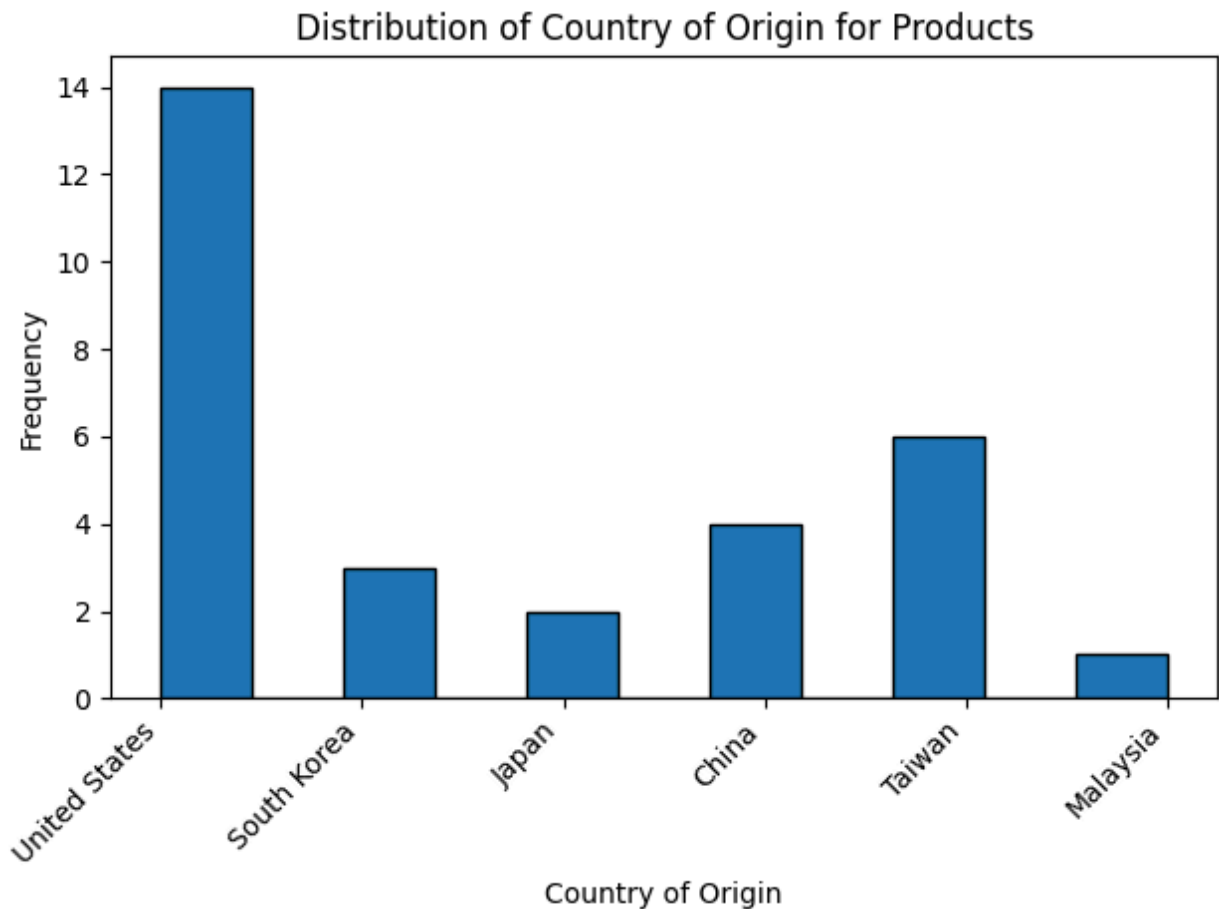
```
# Group data by product and country of origin, counting occurrences
product_country_counts = df.groupby(['Brand', 'Country of Origin'])['Brand'].count().reset_index(name='Count')

# Create a histogram
plt.hist(df['Country of Origin'], bins = 11, edgecolor = 'black')

# Set labels and title
plt.xlabel('Country of Origin')
plt.ylabel('Frequency')
plt.title('Distribution of Country of Origin for Products')
```

```
# Rotate x-axis labels for better visibility
plt.xticks(rotation=45, ha='right')
plt.tight_layout()

# Show the histogram
plt.show()
```



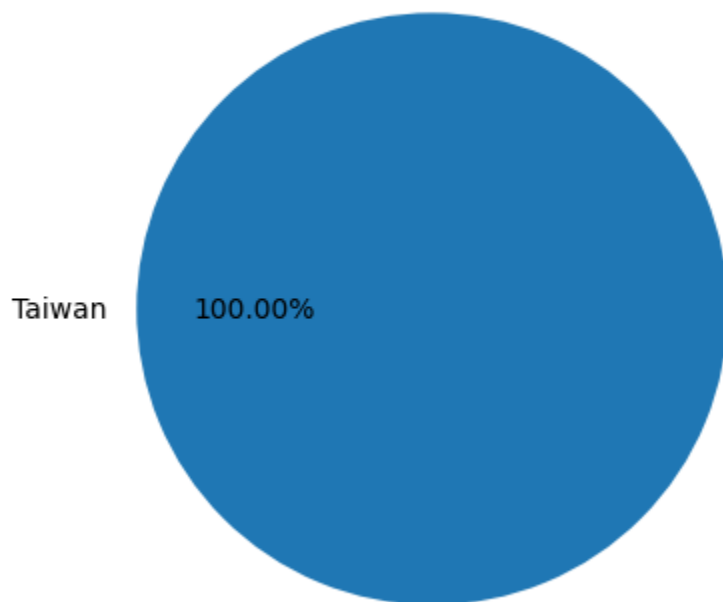
▼ f. Create a pie chart that shows each brand Country of Origin. Label them.

```
# Iterate through each brand and its data group and count occurrences of
each country
for brand, group in df.groupby('Brand'):
    country_counts = group['Country of Origin'].value_counts()

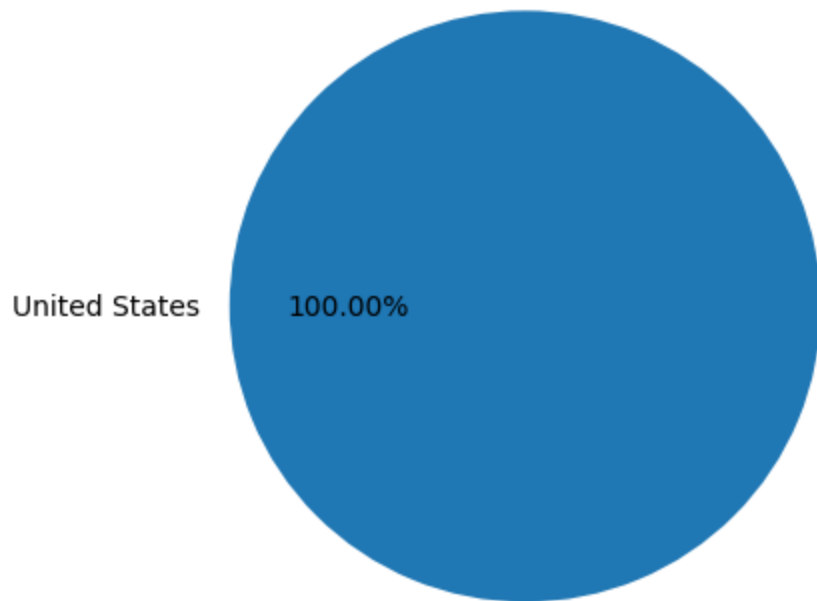
    # Create a pie chart for the current brand
    plt.pie(country_counts,
```

```
labels = country_counts.index,  
autopct = '%1.2f%%')  
  
plt.title(f'Country of Origin for {brand}')  
plt.show()
```

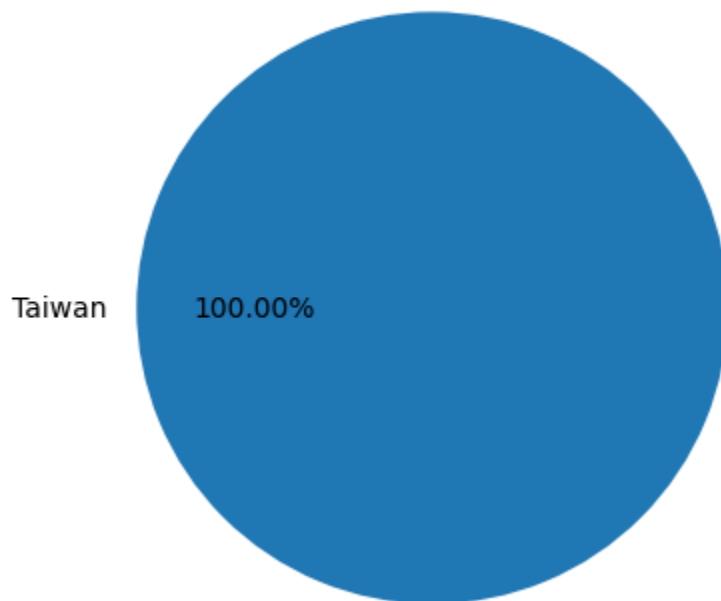
Country of Origin for Acer



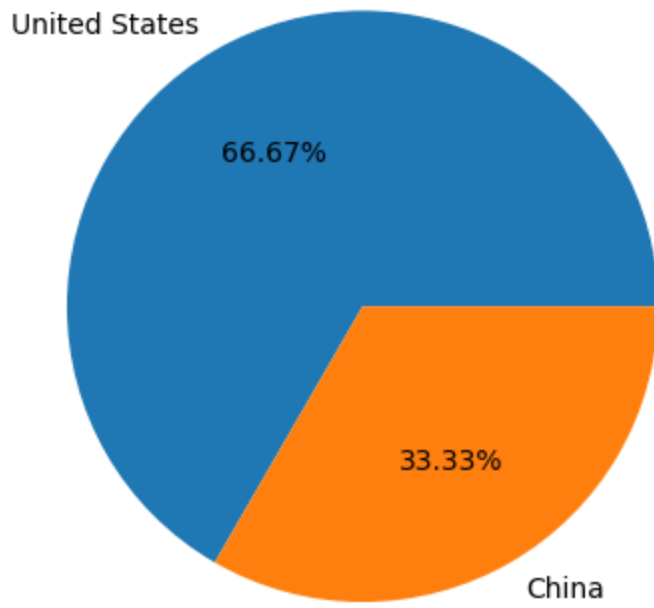
Country of Origin for Apple



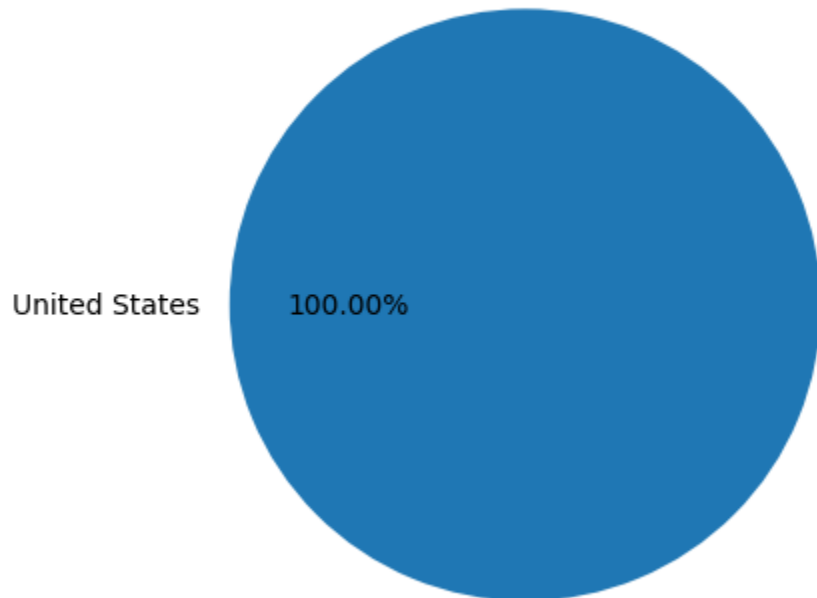
Country of Origin for Asus



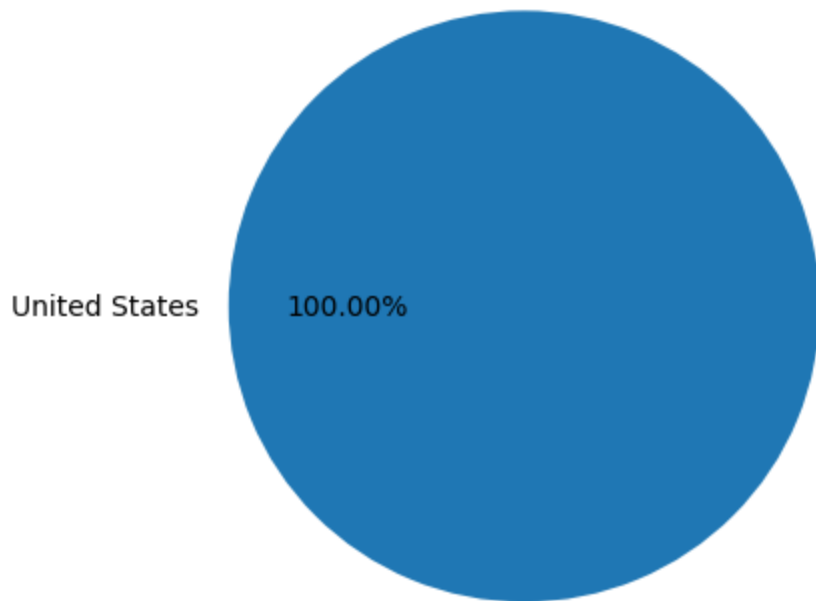
Country of Origin for Dell



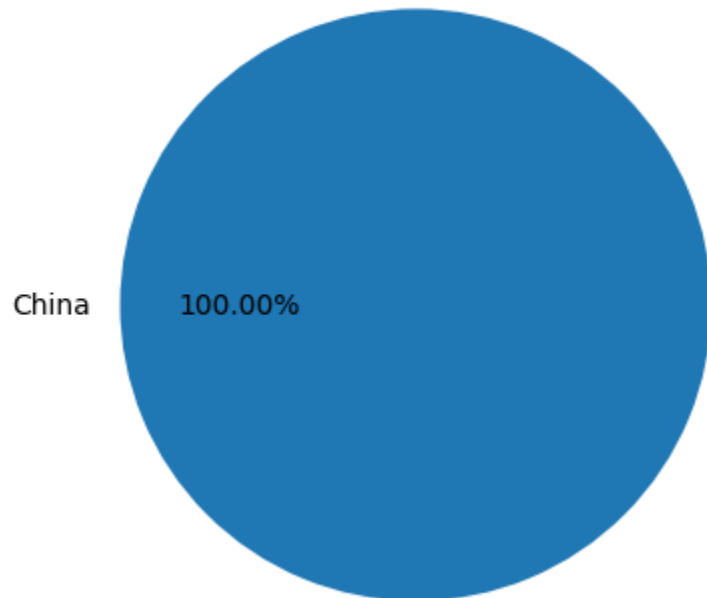
Country of Origin for Google



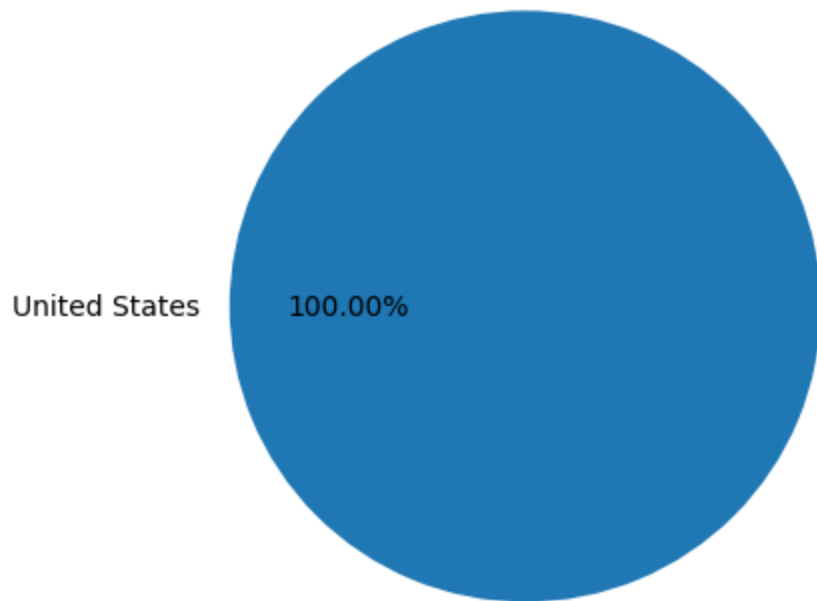
Country of Origin for HP



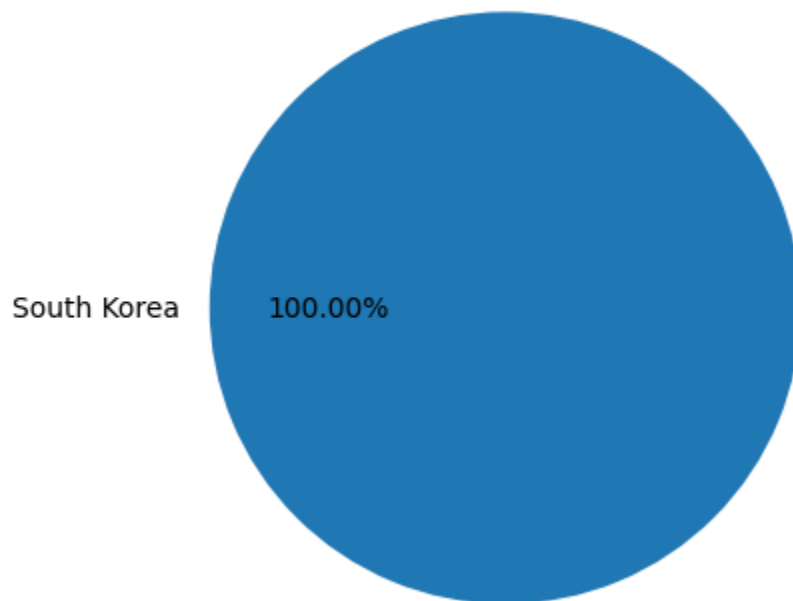
Country of Origin for Lenovo



Country of Origin for Microsoft



Country of Origin for Samsung



Country of Origin for Sony

