Name:_____ **Partners:** _____

Python Activity 25: Plotting Data

Plotting data is useful, but first we have to get data into the right format.

Learning Objectives
Students will be able to:
Content:
- Describe what is needed in order to plot data
- Predict what **matplotlib** code will do

Process:
- Write code that rearranges data for plotting
- Write code that plots data with appropriate labels

Prior Knowledge
- Python concepts: import, x&y axes, lists, loops

Critical Thinking Questions:

1. Examine the sample code below, which uses data on Lickety Split's yearly ice cream sales of their flavor, Purple Cow, and the output from this code:

Set-up Code

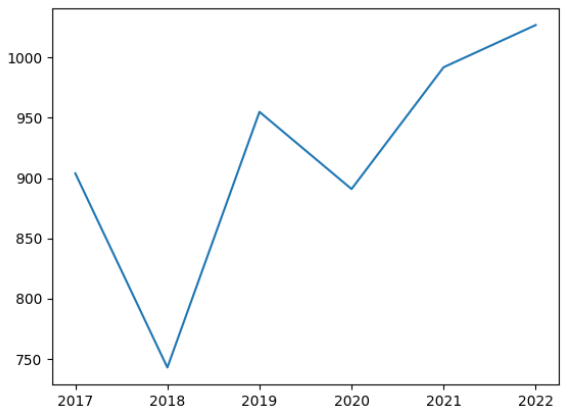
```
import matplotlib.pyplot as plt

years = [2022, 2021, 2020, 2019, 2018, 2017]
num_sales = [1027, 992, 891, 955, 743, 904]

x_values = years
y_values = num_sales

plt.plot(x_values, y_values)
plt.show()
```

Sample Output



a. Circle the code concepts that are new to us.

b. What *type* of variable is `num_sales`? _____

What *type* of values are the *elements* of `num_sales`? _____

What does the data in `num_sales` represent? _____

What is the *smallest* value of the elements in `num_sales`? _____

What is the *largest* value of the elements in `num_sales`? _____

What is the *smallest* value of the X-axis in the sample output? _____

What is the *largest* value of the X-axis in the sample output? _____

How might the data in `years` relate to the values in the X-axis of the output?

_____

c. What *type* of variable is `num_sales`? _____

What *type* of variable are the *elements* of `num_sales`? _____

What does the data in `num_sales` represent? _____

What is the *smallest* value of the elements in `num_sales`? _____

What is the *largest* value of the elements in `num_sales`? _____

What is the *smallest* value of the Y-axis in the sample output? _____

What is the *largest* value of the Y-axis in the sample output? _____

How might the data in `num_sales` relate to the values in the Y-axis of the output? _____

d. What might the lines with, `plt.plot(x,y)` & `plt.show()`, be doing?

_____

e. What might happen to the output, if we switched the `x_values` and `y_values` where they're assigned?

_____

f. What might the line, `import matplotlib.pyplot as plt`, do?

_____

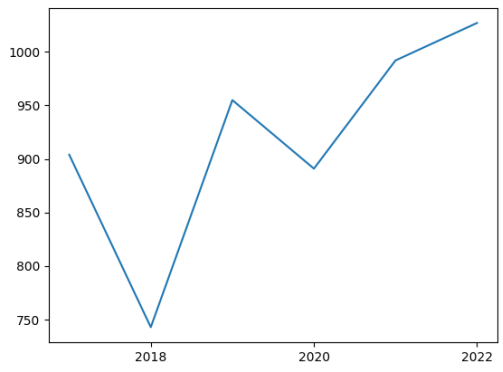
FYI: When we import a module, we can use the **as** keyword to specify a shorter name that we can refer to that module as in our code. For very common modules, such as **matplotlib**, this is very common practice.

2. Examine the sample code and output below, which continues from Question 1:

Sample Code, continued

```
plt.plot(years, num_sales)
plt.xticks(years[::2])
plt.show()
```

Sample Output



a. Circle the code that is different from Question1.

b. Circle what is different in this sample output, compared to Question 1's output.

c. Which of this new code might be responsible for the changes we see in the sample output? _____

d. What might the `plt.xticks(..)` method do?

_____

e. If you had to guess, what might a `plt.yticks(..)` method do?

_____

f. If we replaced line 13 with `plt.xticks(years,['Y1', 'Y2', 'Y3', 'Y4', 'Y5', 'Y6'])` the X-tick on our plot that currently says 2018 would be replaced with `Y2` , 2020 with `Y4`, and 2022 with `Y6`. Why might that be?

_____

What might the second parameter of `plt.xticks(..)` represent?

_____

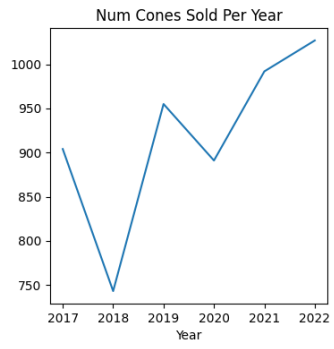
3. Examine the sample code and output below, which continues from Questions 1 & 2:

Sample Code, continued

```
plt.figure(figsize=(4, 4))
plt.plot(years, num_sales)
plt.xticks(years)

plt.xlabel("Year")
plt.ylabel("Num Cones Sold")
plt.title("Num Cones Sold Per Year")
plt.show()
```

Sample Output



a. Circle the code that is different from Question 2.

b. Circle what is different in this sample output, compared to Question 2's output.

c. Match the function below, on the left, to what you think it might do, on the right:

`plt.figure(figsize=(4, 4))`	Specifies which values to show on the X-axis
`plt.plot(x_vals, y_vals)`	The values to show on the Y-axis, and their labels
`plt.xticks(listXticks)`	Specifies a top caption for the plot
`plt.yticks(l1,l2)`	Specifies the X-axis label
`plt.xlabel(a_string)`	Specifies the size of the plot
`plt.ylabel(a_string)`	Specifies the Y-axis label
`plt.title(a_string)`	Displays the the completed plot
`plt.show()`	Makes the plot

4. Examine the sample code and output below:

Sample Code

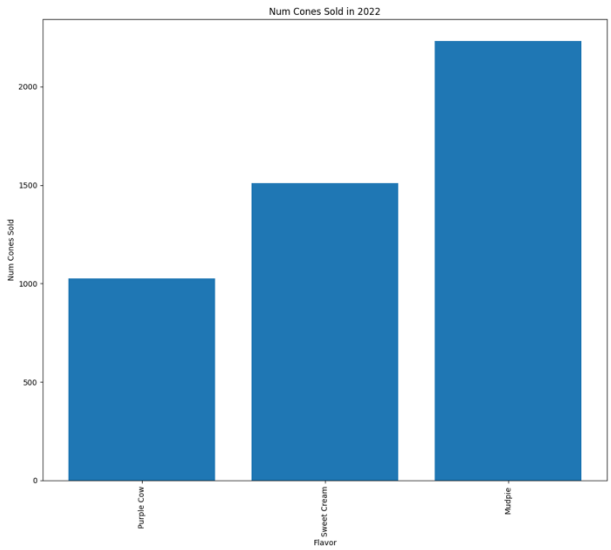
```
import matplotlib.pyplot as plt

flavors = ["Purple Cow", "Sweet Cream", "Mudpie"]
num_sales = [1027, 1509, 2231]

# Create a new figure:
plt.figure()
# Create a bar chart
plt.bar(flavors, num_sales)
plt.xticks(flavors, flavors, rotation=90)

# axis labels and title
plt.xlabel("Flavor")
plt.ylabel("Num Cones Sold")
plt.title("Num Cones Sold in 2022")
plt.show()
```

Sample Output



a. Circle the code that is different from Question 3.

b. Circle what is different in this sample output, compared to Question 1's output.

What *kind* of chart did we make in Question 1-3? _____

What *kind* of chart did we make in this Question 4? _____

Which of the code we circled in (a) might be responsible for the change in chart type?

_____

c. What variable represents the X-values in this example? _____

What variable represents the X-values in Question 1-3? _____

How do these two variables differ? _____

d. What variable represents the Y-values in this example? _____

What variable represents the Y-values in Question 1-3? _____

How do these two variables differ? _____

e. What *type* of object is `flavors`? _____ of _____

What does `flavors` represent in the above code? _____

What *type* of object is `num_sales`? _____ of _____

What does `num_sales` represent in the above code? _____

f. If you had to guess, what might the `rotation=90` argument value do to our X-ticks?