Introduction to animating plots with Matplotlib

Prerequisite skills:

1. Lists
2. Matplotlib
3. Making figures

Objectives:

1. Learn how to animate figures in Python using matplotlib

Steps:

1. Here is some code I have provided to learn: https://replit.com/@marygracealbrig/matplotlibanimation
2. This is a description of what is happening in this Python script:

This code uses Matplotlib to create an animated bar chart representing the GDP (Gross Domestic Product) of six countries over time (****note: these are not correct GDPs, they are just examples). As we go through the code step by step, copy it over to your own replit file and see if it will run at the end!

1. **Import libraries:**

```python
import matplotlib.pyplot as plt
from matplotlib.animation import FuncAnimation
```

The code imports the necessary libraries for creating plots and animations.

2. **Set up figure and axes:**

```python
fig = plt.figure(figsize=(7, 5))
axes = fig.add_subplot(1, 1, 1)
```

The code creates a figure with a size of 7 inches by 5 inches and adds a subplot to it. The subplot is a single plot in a grid of 1 row and 1 column.

3. **Set Y-axis limits:**

```python
axes.set_ylim(0, 300)
```

The Y-axis of the plot is set to range from 0 to 300.

4. **Define color palette:**

```python
palette = ['blue', 'red', 'green', 'darkorange', 'maroon', 'black']
```

A list of colors is defined to be used for the bar plots.

5. **Assign empty lists for data:**

```python
india = []
china = []
germany = []
usa = []
canada = []
uk = []
```

Empty lists are created to store the GDP values for each country.

6. Define animation function:

```python
def animation_function(i):
    india = i
    china = 5 * i
    germany = 3 * i
    usa = 2 * i
    canada = 6 * i
    uk = 3 * i

    plt.xlabel("Country")
    plt.ylabel("GDP of Country")

    plt.bar(["India", "China", "Germany", "USA", "Canada", "UK"],
            [india, china, germany, usa, canada, uk],
            color=palette)
```

The animation_function() takes parameter “i”, which represents the frame number of the animation. The GDP values for each country are calculated based on this frame number. The bar chart is then updated with the new values for each country.

7. **Set title:**

```python
plt.title("Bar Chart Animation")
```

You can set this to be whatever you would like!

8. **Create animation:**

```python
animation = FuncAnimation(fig, animation_function, interval=50)
```

The FuncAnimation class is used to create the animation. For arguments, it takes the figure, the animation function, and the interval between frames in milliseconds. In this case, frames are updated every 50 milliseconds.

9. **Show plot:**

```python
plt.show()
```

10. Once you are able to successfully run this code, see what else you can come up with! Can you play around with the colors or the speed that the bars move? Is there another animated bar plot that you can come up with? Be creative, but you're also welcome to use Google as a reference ☺