5. Animation Using Python Program

5.1 Aim

To explore Animation functions in python with the following task objectives

1. Generate 10 frames for animating the shape of the Gaussian Bell function by changing the variance from 2 to 0.2. Also save the frames as tmp_0001, tmp_0002………tmp_0010 as “png file”

2. Create a SINE wave animation “gif” file by Write python program and save it as “sine_wave.gif”

3. Write python program to generate animated “gif” image file for a moving coil effect(Growing Coil animation)

5.2 Background

In this experiment:

(i) Using matplotlib and Numpy modules, import matplotlib.animation as animation.use plt.savefig, animation, FuncAnimation, save, matplotlib.animation.PillowWriter functions of matplotlib

5.3 Software Used

1. Anaconda Navigator
2. Jupyter Notebook

5.4 Pre Lab Questions

1. List all the functions used for animation in matplotlib and its uses.
2. Discuss the syntax and arguments of plt.savefig animation.FuncAnimation, save, matplotlib.animation.PillowWriter functions of matplotlib.

5.5 Procedure

1. In Jupyter Notebook click on ‘New Launcher’ and then single click on ’Python3’ under Notebook.
2. Type your program to get the desired output.
3. To view the output, click on ‘Run’ or press ‘Shift+Enter’ to execute the program of the selected cell. Note: In case of error, refer to the error message and do the required changes.

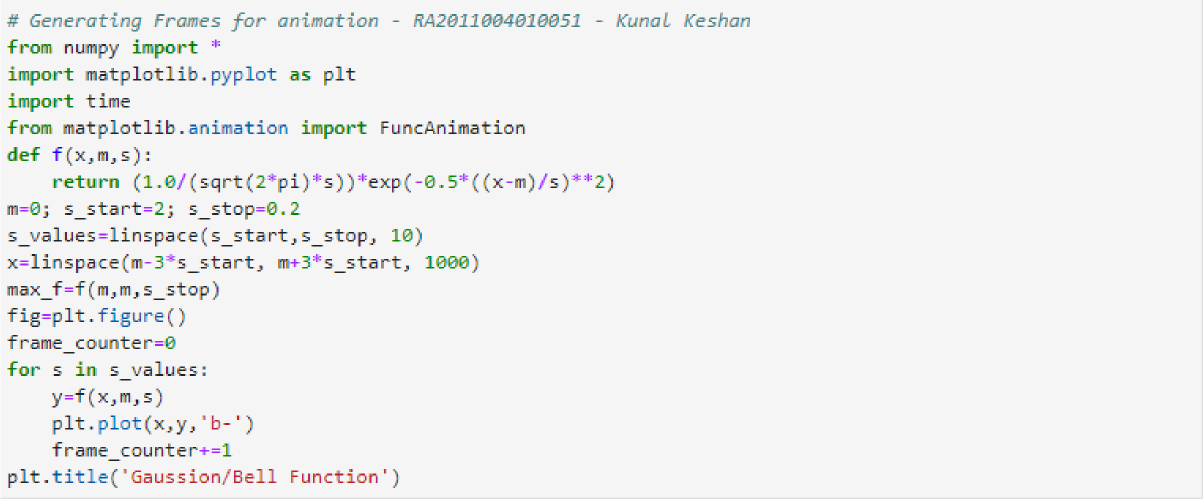
5.6 Program

5.6.1 Generating Frames for animation

Code:

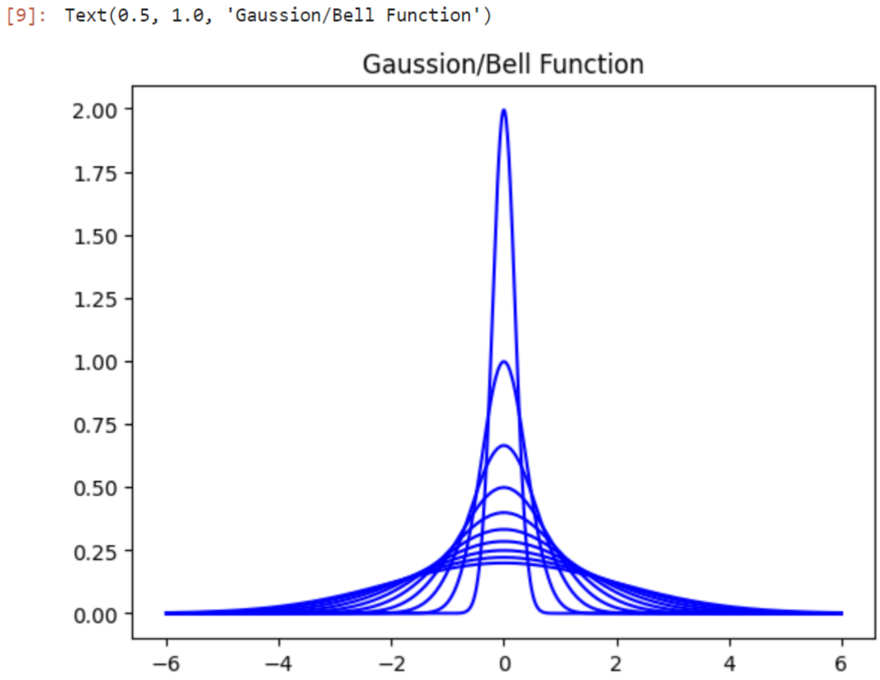
```
# Generating Frames for animation - RA2011004010051 - Kunal Keshan
from numpy import *
import matplotlib.pyplot as plt
import time
from matplotlib.animation import FuncAnimation
def f(x,m,s):
   return (1.0/(sqrt(2*pi)*s))*exp(-0.5*((x-m)/s)**2)
m=0; s_start=2; s_stop=0.2
s_values=linspace(s_start,s_stop, 10)
x=linspace(m-3*s_start, m+3*s_start, 1000)
max_f=f(m,m,s_stop)
fig=plt.figure()
frame_counter=0
for s in s_values:
   y=f(x,m,s)
   plt.plot(x,y,'b-')
   frame_counter+=1
plt.title('Gaussion/Bell Function')
```

Observation:



```
# Generating Frames for animation - RA2011004010051 - Kunal Keshan
from numpy import *
import matplotlib.pyplot as plt
import time
from matplotlib.animation import FuncAnimation
def f(x,m,s):
    return (1.0/(sqrt(2*pi)*s))*exp(-0.5*((x-m)/s)**2)
m=0; s_start=2; s_stop=0.2
s_values=linspace(s_start,s_stop, 10)
x=linspace(m-3*s_start, m+3*s_start, 1000)
max_f=f(m,m,s_stop)
fig=plt.figure()
frame_counter=0
for s in s_values:
    y=f(x,m,s)
    plt.plot(x,y,'b-')
    frame_counter+=1
plt.title('Gaussion/Bell Function')
```

Output:



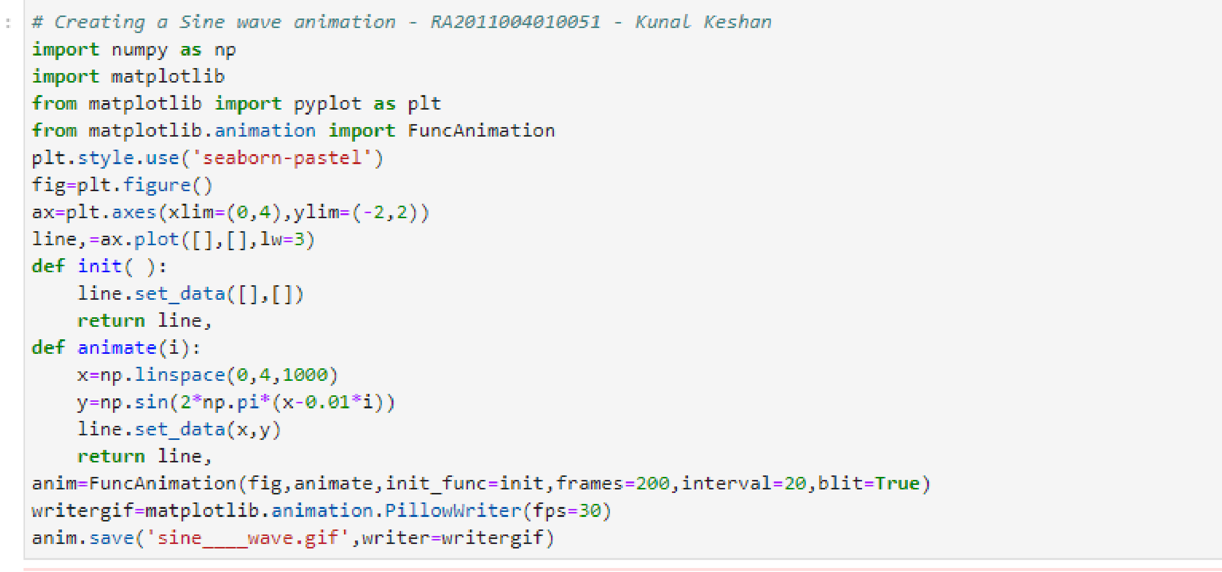
```
[9]: Text(0.5, 1.0, 'Gaussion/Bell Function')
```

5.6.2 Creating a SINE wave animation

Code:

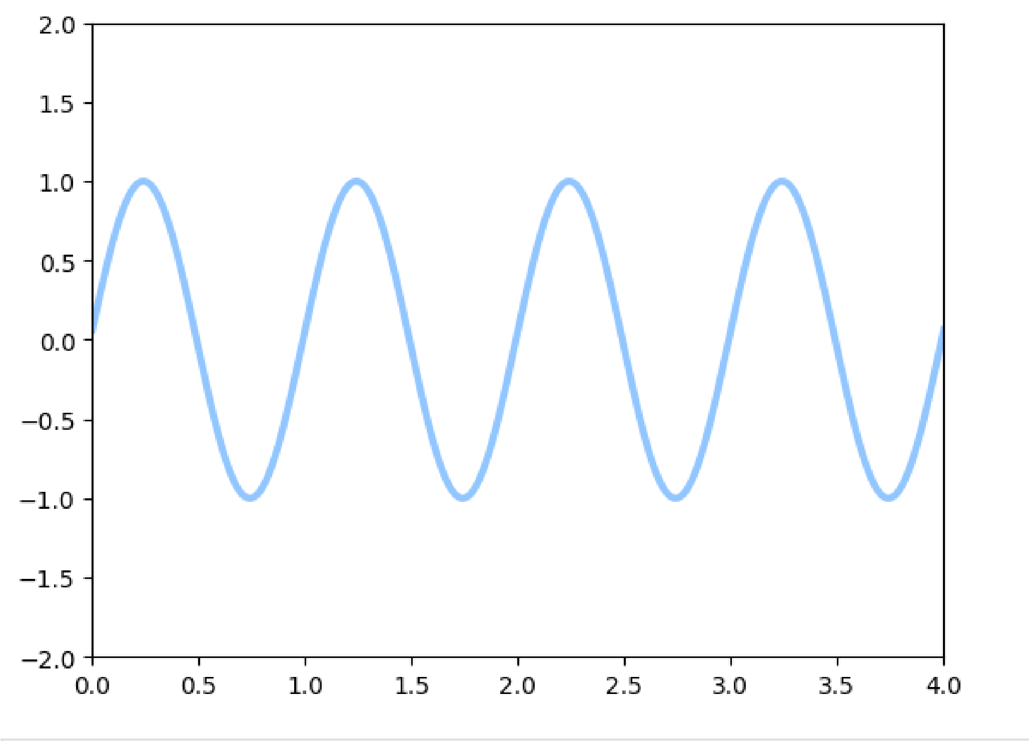
```
# Creating a Sine wave animation - RA2011004010051 - Kunal Keshan
import numpy as np
import matplotlib
from matplotlib import pyplot as plt
from matplotlib.animation import FuncAnimation
plt.style.use('seaborn-pastel')
fig=plt.figure()
ax=plt.axes(xlim=(0,4),ylim=(-2,2))
line,=ax.plot([],[],lw=3)
def init( ):
    line.set_data([],[])
    return line,
def animate(i):
    x=np.linspace(0,4,1000)
    y=np.sin(2*np.pi*(x-0.01*i))
    line.set_data(x,y)
    return line,
anim=FuncAnimation(fig,animate,init_func=init,frames=200,interval=20,blit=True)
writergif=matplotlib.animation.PillowWriter(fps=30)
anim.save('sine____wave.gif',writer=writergif)
```

Observation:



```
# Creating a Sine wave animation - RA2011004010051 - Kunal Keshan
import numpy as np
import matplotlib
from matplotlib import pyplot as plt
from matplotlib.animation import FuncAnimation
plt.style.use('seaborn-pastel')
fig=plt.figure()
ax=plt.axes(xlim=(0,4),ylim=(-2,2))
line,=ax.plot([],[],lw=3)
def init( ):
    line.set_data([],[])
    return line,
def animate(i):
    x=np.linspace(0,4,1000)
    y=np.sin(2*np.pi*(x-0.01*i))
    line.set_data(x,y)
    return line,
anim=FuncAnimation(fig,animate,init_func=init,frames=200,interval=20,blit=True)
writergif=matplotlib.animation.PillowWriter(fps=30)
anim.save('sine____wave.gif',writer=writergif)
```

Output:



5.6.3 Creating Growing Coil animation

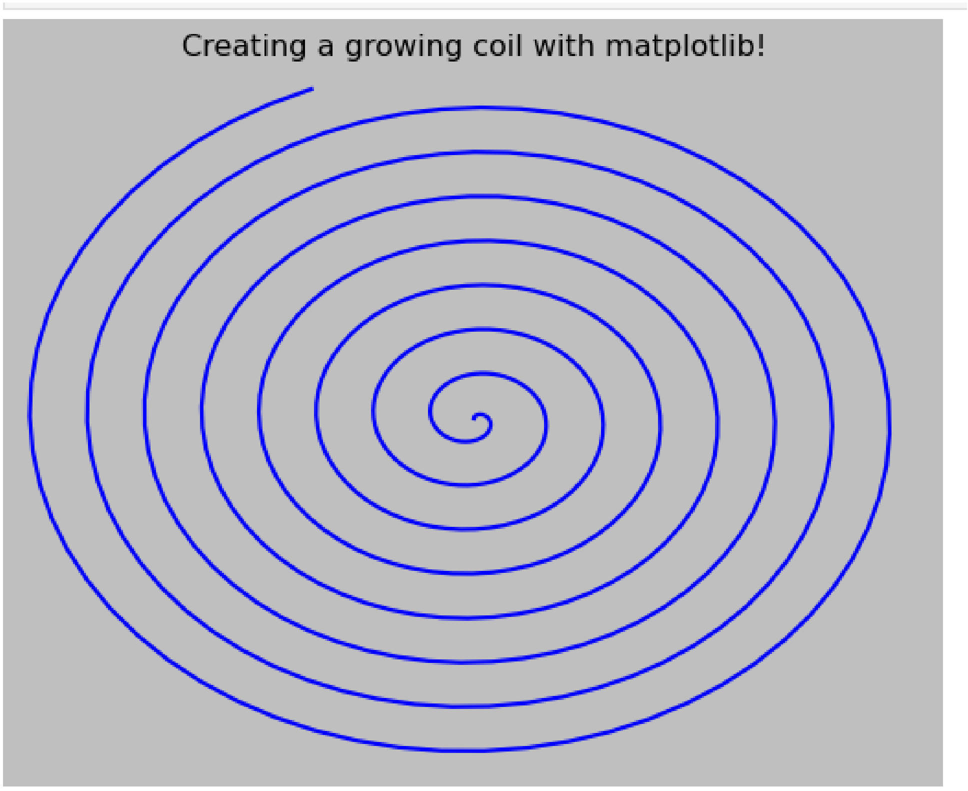
Code:

```
# Creating a Growing Coil Animation - RA2011004010051 - Kunal Keshan
import matplotlib
import matplotlib.pyplot as plt
import matplotlib.animation as animation
import numpy as np
plt.style.use('classic')
fig=plt.figure()
ax=plt.axes(xlim=(-50,50),ylim=(-50,50))
line,=ax.plot([],[],lw=2)
def init():
   line.set_data([],[])
   return line,
xdata,ydata=[],[]
def animate(i):
   t=0.1*i
   x=t*np.sin(t)
   y=t*np.cos(t)
   xdata.append(x)
   ydata.append(y)
   line.set_data(xdata,ydata)
   return line,
plt.title('Creating a growing coil with matplotlib!')
plt.axis('off')
anim=animation.FuncAnimation(fig,animate,init_func=init,frames=500,interval=20,blit=True)
writergif=matplotlib.animation.PillowWriter(fps=30)
anim.save('Desktop/growingcoil2.gif',writer=writergif)
```

Observation:

```
# Creating a Growing Coil Animation - RA2011004010051 - Kunal Keshan
import matplotlib
import matplotlib.pyplot as plt
import matplotlib.animation as animation
import numpy as np
plt.style.use('classic')
fig=plt.figure()
ax=plt.axes(xlim=(-50,50),ylim=(-50,50))
line,=ax.plot([],[],lw=2)
def init():
    line.set_data([],[])
    return line,
xdata,ydata=[],[]
def animate(i):
    t=0.1*i
    x=t*np.sin(t)
    y=t*np.cos(t)
    xdata.append(x)
    ydata.append(y)
    line.set_data(xdata,ydata)
    return line,
plt.title('Creating a growing coil with matplotlib!')
plt.axis('off')
anim=animation.FuncAnimation(fig,animate,init_func=init,frames=500,interval=20,blit=True)
writergif=matplotlib.animation.PillowWriter(fps=30)
anim.save('Desktop/growingcoil2.gif',writer=writergif)
```

Output:



5.7 Post Lab Questions

1. Change the frame number to 700 and interval to 10 ,5 and 40 and give your observation in Creating Growing Coil animation. Save the animated file in "MP4"format
2. Make an animation mp4 or gif file from your choice of image frames. You can decide your frame numbers, animation interval and the parameters (**open ended**)

5.8 Result

Thus Python is written for creating animation effect files using matplotlib functions

Post Lab Answers

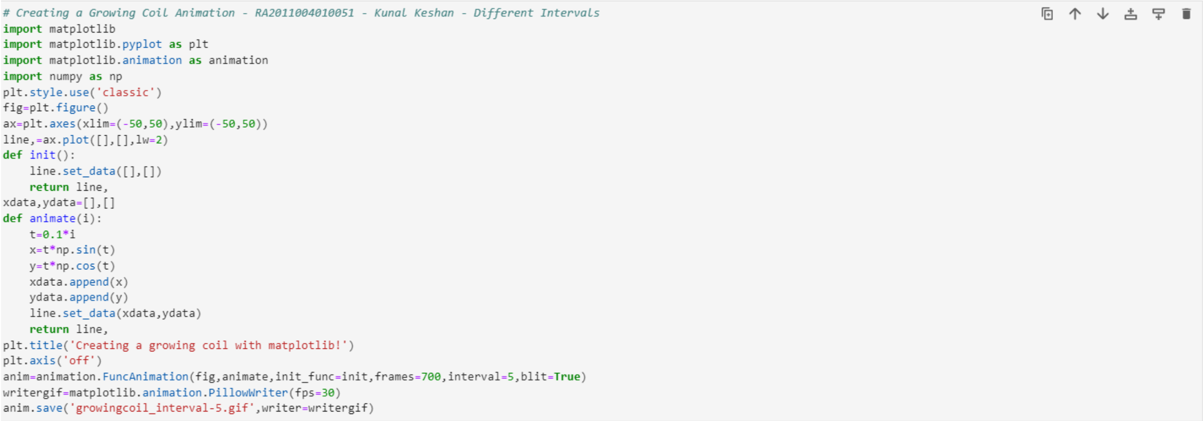
1. Change the frame number to 700 and interval to 10 ,5 and 40 and give your observation in Creating Growing Coil animation. Save the animated file in “MP4”format

Solution:

Code:

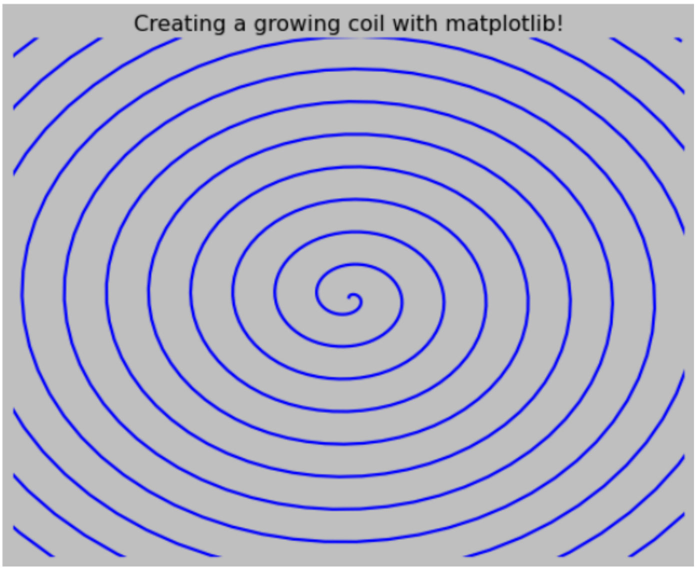
```
# Creating a Growing Coil Animation - RA2011004010051 - Kunal Keshan - Different Intervals
import matplotlib
import matplotlib.pyplot as plt
import matplotlib.animation as animation
import numpy as np
plt.style.use('classic')
fig=plt.figure()
ax=plt.axes(xlim=(-50,50),ylim=(-50,50))
line,=ax.plot([],[],lw=2)
def init():
    line.set_data([],[])
    return line,
xdata,ydata=[],[]
def animate(i):
    t=0.1*i
    x=t*np.sin(t)
    y=t*np.cos(t)
    xdata.append(x)
    ydata.append(y)
    line.set_data(xdata,ydata)
    return line,
plt.title('Creating a growing coil with matplotlib!')
plt.axis('off')
anim=animation.FuncAnimation(fig,animate,init_func=init,frames=700,interval=5,blit=True)
writergif=matplotlib.animation.PillowWriter(fps=30)
anim.save('growingcoil_interval-5.gif',writer=writergif)
```

Observation:



```
# Creating a Growing Coil Animation - RA2011004010051 - Kunal Keshan - Different Intervals
import matplotlib
import matplotlib.pyplot as plt
import matplotlib.animation as animation
import numpy as np
plt.style.use('classic')
fig=plt.figure()
ax=plt.axes(xlim=(-50,50),ylim=(-50,50))
line,=ax.plot([],[],lw=2)
def init():
    line.set_data([],[])
    return line,
xdata,ydata=[],[]
def animate(i):
    t=0.1*i
    x=t*np.sin(t)
    y=t*np.cos(t)
    xdata.append(x)
    ydata.append(y)
    line.set_data(xdata,ydata)
    return line,
plt.title('Creating a growing coil with matplotlib!')
plt.axis('off')
anim=animation.FuncAnimation(fig,animate,init_func=init,frames=700,interval=5,blit=True)
writergif=matplotlib.animation.PillowWriter(fps=30)
anim.save('growingcoil_interval-5.gif',writer=writergif)
```

Output:



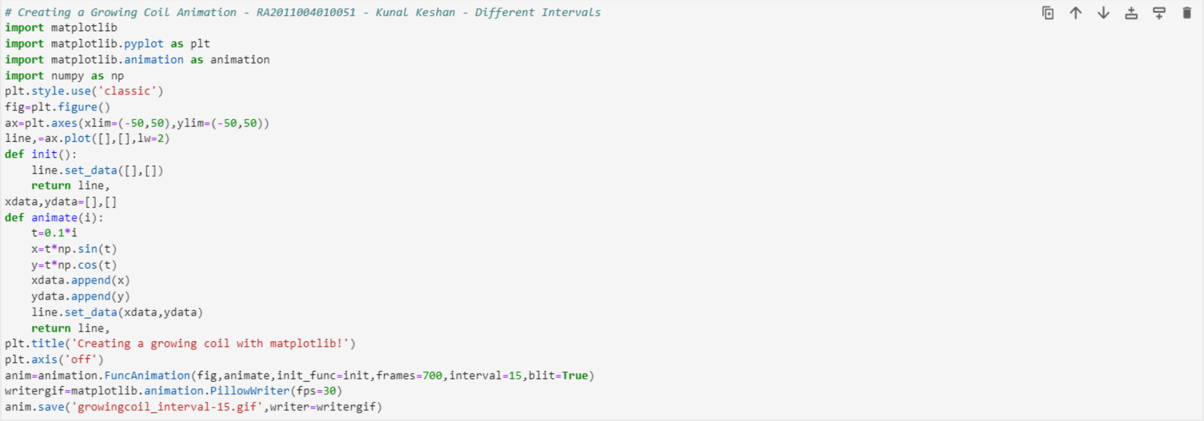
2. Make an animation mp4 or gif file from your choice of image frames. You can decide your frame numbers, animation interval and the parameters (**open ended**)

Solution:

Code:

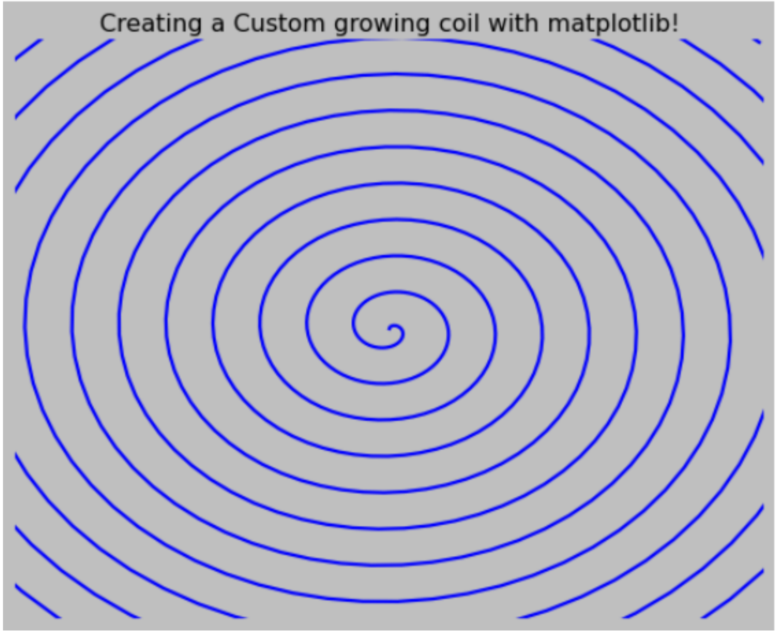
```
# Creating a Growing Coil Animation - RA2011004010051 - Kunal Keshan - Different Intervals
import matplotlib
import matplotlib.pyplot as plt
import matplotlib.animation as animation
import numpy as np
plt.style.use('classic')
fig=plt.figure()
ax=plt.axes(xlim=(-50,50),ylim=(-50,50))
line,=ax.plot([],[],lw=2)
def init():
   line.set_data([],[])
   return line,
xdata,ydata=[],[]
def animate(i):
   t=0.1*i
   x=t*np.sin(t)
   y=t*np.cos(t)
   xdata.append(x)
   ydata.append(y)
   line.set_data(xdata,ydata)
   return line,
plt.title('Creating a growing coil with matplotlib!')
plt.axis('off')
anim=animation.FuncAnimation(fig,animate,init_func=init,frames=700,interval=15,blit=True)
writergif=matplotlib.animation.PillowWriter(fps=30)
anim.save('growingcoil_interval-15.gif',writer=writergif)
```

Observation:



```
# Creating a Growing Coil Animation - RA2011004010051 - Kunal Keshan - Different Intervals
import matplotlib
import matplotlib.pyplot as plt
import matplotlib.animation as animation
import numpy as np
plt.style.use('classic')
fig=plt.figure()
ax=plt.axes(xlim=(-50,50),ylim=(-50,50))
line,=ax.plot([],[],lw=2)
def init():
    line.set_data([],[])
    return line,
xdata,ydata=[],[]
def animate(i):
    t=0.1*i
    x=t*np.sin(t)
    y=t*np.cos(t)
    xdata.append(x)
    ydata.append(y)
    line.set_data(xdata,ydata)
    return line,
plt.title('Creating a growing coil with matplotlib!')
plt.axis('off')
anim=animation.FuncAnimation(fig,animate,init_func=init,frames=700,interval=15,blit=True)
writergif=matplotlib.animation.PillowWriter(fps=30)
anim.save('growingcoil_interval-15.gif',writer=writergif)
```

Output:


Creating a Custom growing coil with matplotlib!