Displaying Graphics In A Separate Window

This function will display a graphic in a separate window for 2 seconds and then close the window. The user will return to the Idle window. Note that tkinter can only display GIF and PGM/PPM images. To use other formats like jpeg, PNG, tiff, bmp, etc., the PIL(low) library must be loaded. This program will:

1) Load the tkinter and PIL libraries. Note that both of these **must** be installed on your computer. The link for the PIL(low) library for Python 3.X is:
   https://pypi.python.org/pypi/Pillow/3.0.0 - choose the correct version
2) Create a tkinter window.
3) Load the graphic into the program
4) Create a canvas object on the window to display the graphic
5) Draw the graphic on the canvas
6) Wait 2 seconds
7) Close the window and return to Idle

Code:

```
from tkinter import *          # Needed for tkinter
from PIL import Image, ImageTk  # tkinter only allows gifs to be used
                               # These libraries allow many other graphic file formats to be used including jpg
import time    # Needed for the delay

def showPic (graphicFile):
    '''graphicFile is the complete name of the graphic file that displayed
    root = Tk ()         # Creates a tkinter window

    image = Image.open (graphicFile)         # Creates a link to the graphic file
    photo = ImageTk.PhotoImage (image)             # Converts the graphic into a tkinter format

    photoHeight = photo.height ()            # Calculates the height of the graphic
    photoWidth = photo.width()               # Calculates the width of the graphic

    # Sizes and places the tkinter window. This must be a single string in the format
    # “heightxwidth + numberOfPixelsFromLeftEdge + numberOfPixelsFromTopEdge
    # In this case, the tk window would be the width and height of the graphic created 100 pixels from the
    # top of the screen and 100 pixels from the left of the screen
    root.geometry (str (photoWidth) + "x" + str (photoHeight) + "+100+100" )

    # Creates a canvas object on the window which displays the graphic
    myCanvas = Canvas (root, width = photoWidth, height = photoHeight, background = 'white')
    myCanvas.pack()

    # Draws the graphic on the canvas. The first two parameters are x and y coordinates of the centre of
    # the graphic
    myCanvas.create_image (photoWidth//2, photoHeight//2, image = photo)

    myCanvas.update()
    time.sleep (2)      # Delays for 2 seconds
    root.destroy()      # Closes the window
```

Note: To have the user close the window when they are finished, replace the last two lines with the command:

```
        root.mainloop ()
```