

HW11-Christmas Tree V2

10 points

Assignment: Write a Python program that draws a tree, and then draws a number of ornaments on the tree. Use inheritance to create your various ornament classes as described below. Add some of each of the ornament types to your tree.

Due: Friday 11/22, 5PM

Turn In: Submit your sources and pictures to BBV. You don't need to submit the images (smallOrnament.gif, tree.gif, star.gif) unless you altered them or added more images.




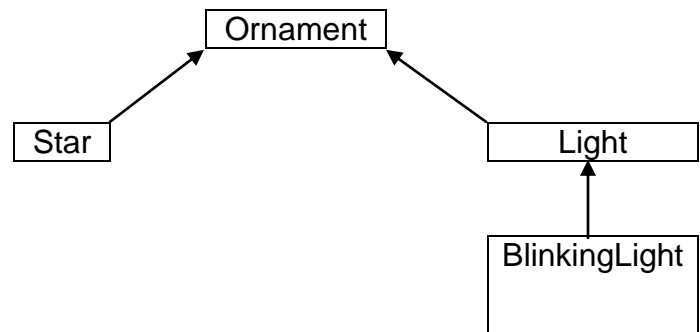
This is an exercise in extending classes to make new classes.

Your main class should create the window, the tree as a background image, and create many instances of the Ornament, Light, and Blinking light classes, and one instance of the Star class, as explained below.

And, the main class is where you'll call `root.mainloop()` as usual.

The Ornament class should create ONE of

the red balls: . The ball should be placed at a random location within the bounds of the tree.



You'll need to pass the canvas to the Ornament creator (the `__init__()` function), so it can create a `canvas_image`.

Here is a function that can select a random point inside of a triangle. The pixel locations of the tree are: top (87, 20), bottom left (12, 270), bottom right (162, 270). You can use this in your Ornament class.

```

def choosePointInTriangle(self, pA, pB, pC):
    a = random.random()
    b = random.random()
    if a+b > 1:
        a = 1-a
        b = 1-b
    c = 1-a-b
    x = int(a*pA.x + b*pB.x + c*pC.x)
    y = int(a*pA.y + b*pB.y + c*pC.y)
    return Point(x,y)
  
```

Break your Ornament class into several different functions, in preparation for extending the Ornament class when making the rest of the classes.

Next, make a Light class that creates a small circle (canvas oval), choosing a color of either "red" or "green" at random. Light should extend Ornament, and inherit as much as possible (certainly the x,y location).

Then, make a Star class, that also extends Ornament. It should draw the star.gif image, and it should be at the top of the tree (not at the randomly selected x,y from Ornament).

Finally, make a BlinkingLight class that extends Light (not Ornament directly!). This will be very similar to the Light class, so you'll be able to inherit and use nearly everything that Light provides. But in addition, create a thread that makes your lights blink on and off. Make the on and off times random, so that all of the BlinkingLights won't flash simultaneously.

All of your classes should be as small as possible. Use inheritance to do as much work for you as possible.