Sean Skinner
1/20/2022
CptS 487
Assignment 1

Language: Java

1.
   ```
   Public class Rectangle
   {
   private int verticalCenterPos;
   private int horizontalCenterPos;
   private int height;
   private int width;
   public int getArea()
   {
       return height * width;
   }
   }

   Public class Circle
   {
   private int verticalCenterPos;
   private int horizontalCenterPos;
   private int radius;
   public int getArea()
   {
       return pi*radius*radius;
   }
   }
   ```

2.
   ```
   Public abstract class Shape
   {
   private int verticalCenterPos;
   private int horizontalCenterPos;
   public int getArea()
   }
   ```

   The name of the relationship between these classes is a form of inheritance, The circle and rectangle classes are subclasses of the shape class. To correctly incorporate the shape class, I need to define methods and variables that all shape classes have in common and then extend the circle and rectangle classes, so they inherit from the shape class. We want to have a shape class so it is simple to create new shape classes that inherit from a common framework as the circle and rectangle classes.

3. No, the shape class should not be able to allow the user to create an unspecified shape, instead it should be an abstract class which can not be instantiated but can be subclassed.

4. In my previous code you would need to have a style be declared in the abstract class shape and would need to be part of the constructor of the specific shape classes such as Rectangle and Circle So that when those objects are instantiated it asks for a style class object to be applied to the shape being drawn.

5. Yes, style should be its own class. I believe the relationship would be that of composition as the aggregate class (shape) has a specified style that will be deleted if the parent class is deleted.

6.
   ```
   Import Shape
   Public class Canvas
   {
   public void draw(Shape s)
   {
       // Some code that would “draw” the given shape object on the canvas
       // The position the shape would appear in is specified in the shape class.
   }
   }
   ```

7. The Canvas should have an association with the shape class where 1 canvas can have 0 to n many shapes. This is different from the relationship between shape and style as the style is tied to the lifetime of the shape and can not exist without it. However, with the canvas and shape classes, shapes need to be able to exist outside of the canvas class before the draw function is called to apply them to the canvas and the canvas does not own these shapes. So even if the canvas were to be deleted the shapes data would still exist, they just wouldn’t be applied to the canvas, while when a shape is deleted the style instances would also be deleted as they are tied to the shape instance’s life cycle. This is all because the user must first declare a shape and then have it be applied to the canvas, so the shape exists outside the life cycle of the canvas.

Note to Professor/Grader:

Were the answer here in depth enough? I saw that the assignment was asking for pseudocode and I assumed I didn’t have to go into too much detail about how this would all work on a technical level and that most of what was being asked here was about the higher level approach to building a class structure so please let me know if I should go into further detail on future assignments. Thanks!