CSE 219 : Recitation 7

The Factory Pattern

Notify your Recitation TA once you have completed this recitation such that they may verify your work.

Recitation Goal

In this recitation you will implement the factory pattern for a familiar class of object, but from scratch.

Recitation Setup

- For this recitation, you will only need your IDE.

Recitation Requirements

1. First, create a *Shape* interface which has only one method called `draw()` with return type `void`.
2. Then create concrete classes *Rectangle*, *Square* and *Circle* such that any instance of these concrete types has the apparent type of a *Shape*. Your `draw()` method in these classes don't have to actually draw or render anything. A simple print statement saying something like, "Drawing a rectangle now." is sufficient.
3. Next, create an enumerable type called *ShapeType*. This `enum` should contain all the concrete shape classes.
4. Create a factory class called *ShapeFactory* with a method called `getShape(ShapeType st)`.
5. Finally, write a separate class called *FactoryPatternDemo*. This class is just going to contain the `main` method. Within this main method, create an instance of the factory class, and use that to in turn create a rectangle, a circle, and a square. Invoke the `draw()` method on each concrete instance to verify that the real type is correct.

When you are done, show your work to the TAs. The TAs will verify your work against the solution provided to them separately.