Object Oriented Programming
Assignment 02 - Drawing and Storing Shapes

March 15, 2023

1 Assignment Goal

The purpose of this assignment is to understand Polymorphism and proper usage of data structures like queues and stacks with objects.

You can access Java Documentation at this link.

For all the graphics, you will need to read the Java Documentation at this link.

2 The Task

In this assignment, you will create a simple drawing program that draws shapes on screen. You will make use of abstraction, encapsulation, inheritance and polymorphism as governed by Object-Oriented Programming to achieve this task. This task is however distributed into the following subtasks that you must fulfill:

1. You will be able to draw a Rectangle, A Circle and a Triangle
2. All these shapes will inherit from a single Shape class
3. You will press 1, 2 and 3 to select between the three classes when drawing
4. There should be text written in the bottom left of the screen that lets us know which shape has been selected to be drawn.
5. You will click and drag to draw a Rectangle and a Circle
6. You should be able to view the Circle or the Rectangle as it is being made to judge how big it should be
7. The triangle will be made by clicking three times on the canvas. As you click, a line will be drawn from the point to the position of your mouse till you click again. On third click, the triangle will be made.
8. All drawn shapes will be filled with random colors, they do not need to have any stroke value
9. All drawn shapes will be stored in a Stack which is created by you
10. When you press right click, the latest drawn figure is popped from the Stack and stored in the redo Queue. Hence you will need to implement the Queue
11. When you press the middle mouse button, the shapes are de-queued from the Queue and Pushed into the Stack. Keep in mind that the Node that you will define will store a Shape class, thus you do not need to make stacks and queues for all the different shapes. Thus you will be able to undo and redo shapes.
12. If the user undos shapes and then draws something on the canvas, it will end up purging the entire redo queue.
13. When you will close the program, it will store the state of all the shapes in a text file.
14. When you will open the program, it will read from the same text file and repopulate the shapes as they were when quitting the program

A very basic UML diagram is given below to let you know the inheritance among classes. Keep in mind that Shape is going to be an abstract class, so be careful in deciding which functions need to be abstract and which do not. We will cover this topic in Week06.

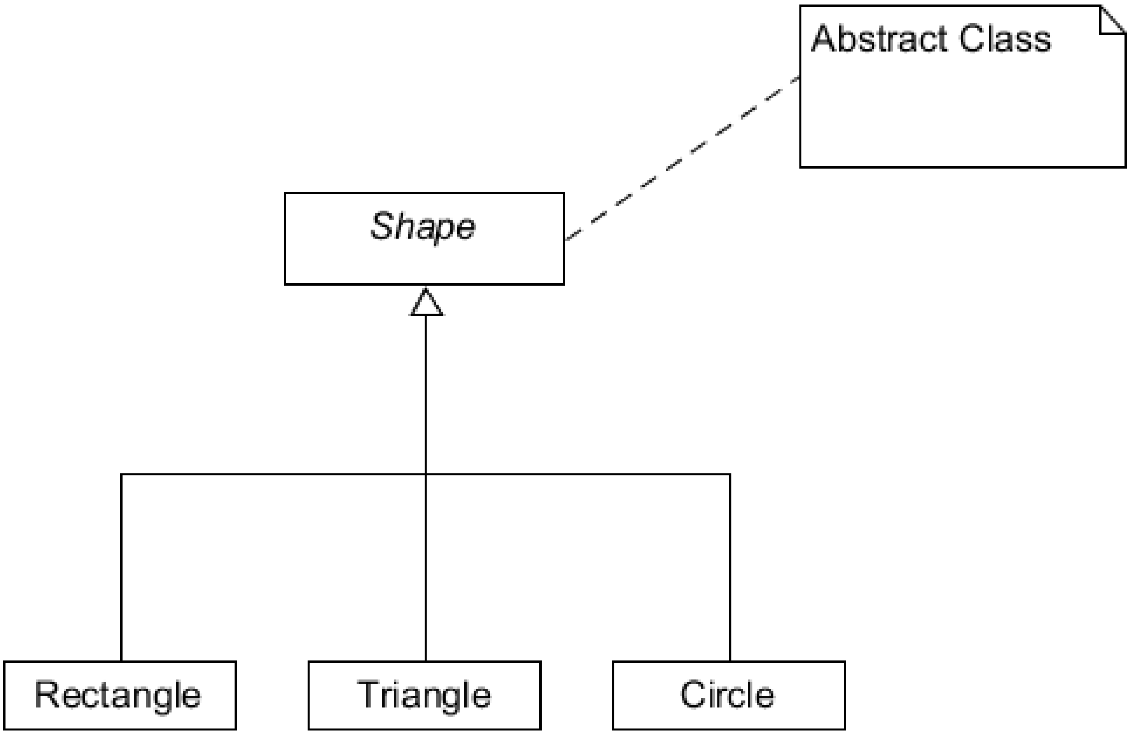

Figure 1: UML for Inheritance

3 Critical Points

Make use of the **DrawingProgram - Sample.zip** uploaded on LMS to write your code. You can make use of all the code that you created in labs or given to you in the lectures/labs and build on top of it. It is very important that you work on this assignment daily as it will prepare you for the upcoming midterm. For queues and stacks, implement and use your own data structures.

4 Submission

The deadline of the assignment is 22nd of March at 11:55 pm. There will be no extensions and no late submissions will be accepted. You should complete the assignment at least a day before the submission deadline.

You will zip the source folder and name it as your name ERP.zip. Suppose your name is John Doe and your ERP is 12345, then the name of your file should be John_Doe_12345.zip. You will then upload it to LMS before the deadline.

- If you upload the wrong file, you will not get any marks.
- If at the last moment LMS does not work, you will not get marks. LMS tends to fail at the last moment as everyone is uploading files. The load makes it crash so avoid uploading the files at the last moment. If you choose to take that risk, then it will be your choice.
- Do not email me your assignments or upload them on dropbox, they will not be checked
- If the size of the files is too big, you are uploading files that are not required so remove those files from the zip file and then upload them.
- If your files are not in a zip file, they will not be checked.

Keep in mind that any attempt at plagiarism or collusion will cause you to automatically fail the course, hence write your own code.