

## Step 11 – More stuff and Large Task 2

So, second large task. As promised, the task will have some customization, but not as two distinct tasks. The task itself is large, and you can choose functionality to implement. The important thing is to have a well functioning program that gives you the opportunity to show the knowledge you have acquired so far.

Goals for this step:

- I am familiar with extended exception handling such as multi-catch, exception-chaining and try-with-resources.
- I know the life cycle of an object and how Garbage Collector cleans up objects.
- I understand the concepts of *high cohesion* and *low coupling*.
- I understand what is meant by the terms *pass by value* and *pass by reference*.

Relevant chapters in the book:

- Chapter 19: Pass by Value vs. Pass by Reference
- Chapter 20: Garbage Collection

[Here](#) is the questionnaire for step 11 where you can tell how it went.

### Task

You should try to create a program that is based on the geometric shapes we have worked with before. Feel free to use the code in Canvas or on the git repo as a starting point (or use the code you have written yourself). If you choose to use the Canvas code as a starting point, you can run the program. Then the expected result is:

*A Circle with radius = 2.0 and center=x is 0.0, y is 0.0, which is a subclass of A Shape with color of java.awt.Color[r=64,g=64,b=64] and not filled.*

The program should read information about geometric shapes stored on file (the file is in Canvas). You can change the format of the file if you wish. The figures must be created and placed in a suitable data structure.

All figures must be "Drawable", ie they must be able to draw themselves. We simulate this by printing (SOUT) all the information about the figure (since drawing them properly is challenging).

Once the information about the figures has been read in (and the objects created), the program should display a menu to the user. Here is a description of the basic functionality:

- Draw all the shapes
- Get information on the total area of all the squares.
- Exit

The program can be expanded with more functionality. Here are some suggestions:

- Add a shape (circle, rectangle, square).
- Move a shape.
- Write information about all the shapes (including the newly added ones) to file.

You are encouraged to come up with more relevant ideas about shape handling, if you are done with these requirements.