Joseph Julian
05 February 2023

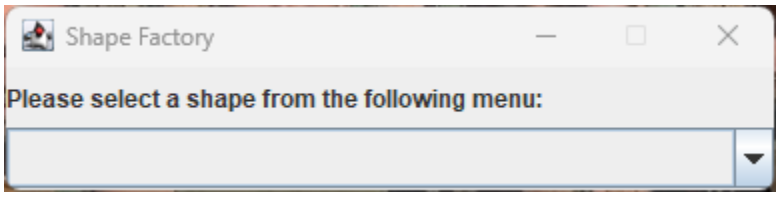
Week 4 Project 2 - "Shape Factory"

Project Description: Building off of my first project, ***Shapes***, I designed a Java Swing GUI that replaced the console-based menu, allowing users to select a desired shape from a combo box, enter the appropriate dimensions, and then have the shape drawn inside of a JFrame. As recommended in the project rubric, I elected to display an image of the respective 3D shape, while still providing the calculated volume. This project was created using Intellij IDEA.

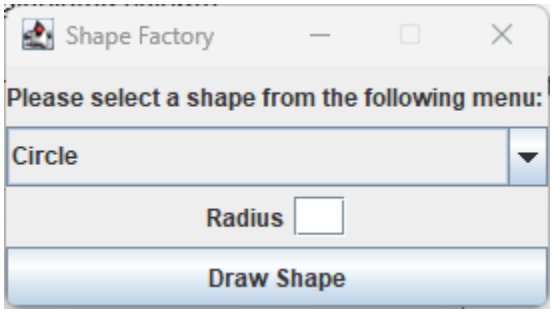
User's Guide: To use **Shape Factory**, please navigate to the project directory and use ***Main.class*** as the entry point for the program. **Follow the directions** on the GUI, and be sure to read any error messages that arise in order to achieve the desired results. Images are stored locally within the project directory, under ***ShapeFactory/src/img***.

Here are some screenshots of my program running, and notes that described the intended and actual outcomes.

Example 1:

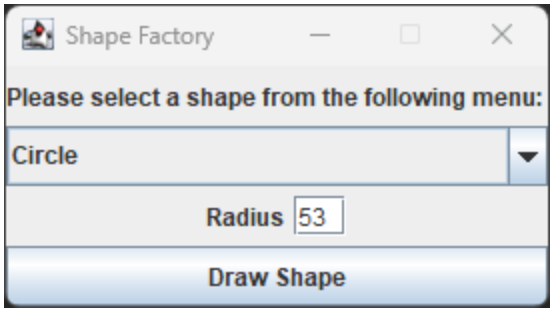


*This is the main GUI for my program, titled **Shape Factory**. From this menu, the user is first expected to select a shape from the combo box (see below):*

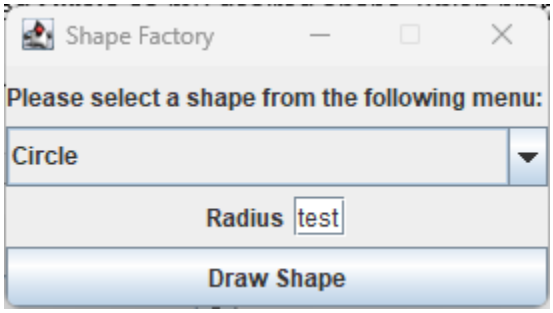


*In this example, I selected **Circle** as my desired shape, which prompted the other GUI components to become visible. From here, I can enter a preferred radius for my circle, then click **Draw Shape** to construct it within a new frame.*

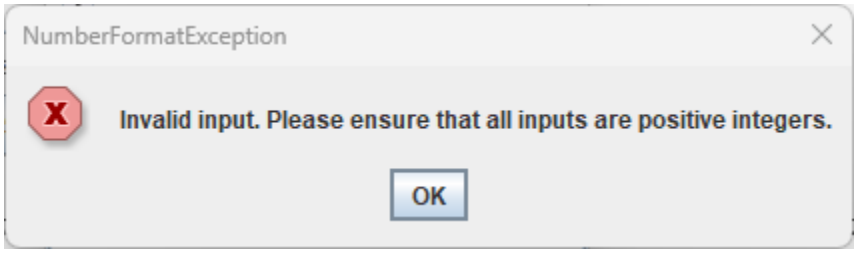
Joseph Julian
05 February 2023



*For this example, I entered the integer **"53"** as my desired radius for my circle. Please note, my program will attempt to catch invalid input, such as non-positive / non-number entries.*

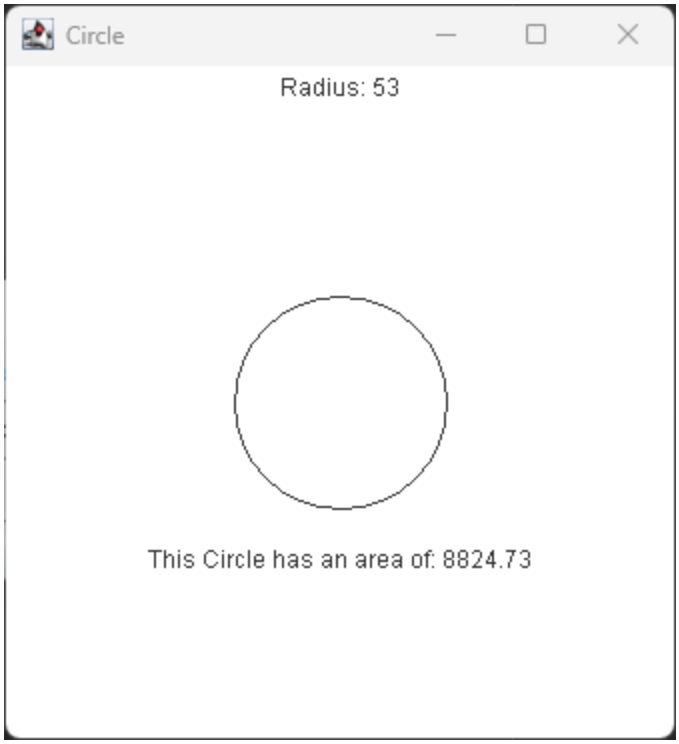


*When attempting to enter **"test"** as my radius value, I was prompted with the following message, and returned to the main GUI.*



*Upon entering the correct radius format, and clicking **Draw Shape**, a new frame appears on top of my main GUI, detailing the radius for my circle, the calculated area, and a painted shape.*

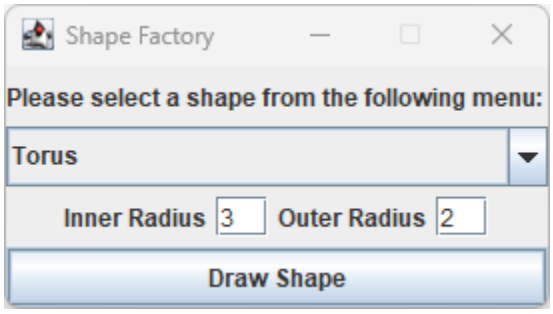
Joseph Julian
05 February 2023



*To return to the main GUI, I can simply click the **X** in the top-right of the window, triggering the default **DISPOSE_ON_CLOSE** operation.*

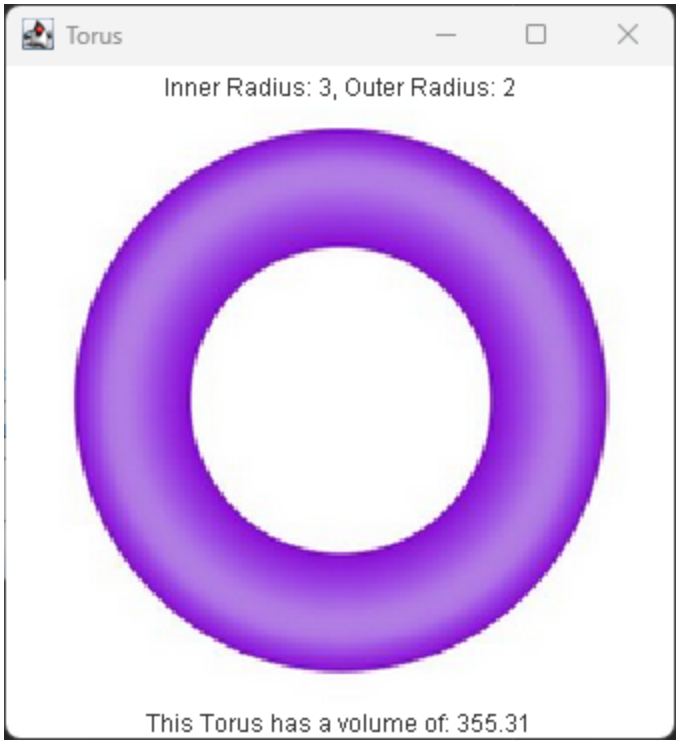
Example 2:

*For my next example, I chose the 3D shape Torus from the menu, and entered an appropriate inner and outer radius of **"3"** and **"2"**, respectively.*



Clicking Draw Shape for this example displayed a local image of a torus, along with the entered dimensions and calculated volume.

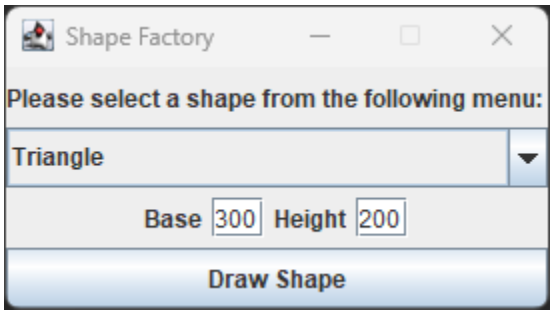
Joseph Julian
05 February 2023



As expected, these calculations were correct, and the appropriate image was displayed inside of the frame.

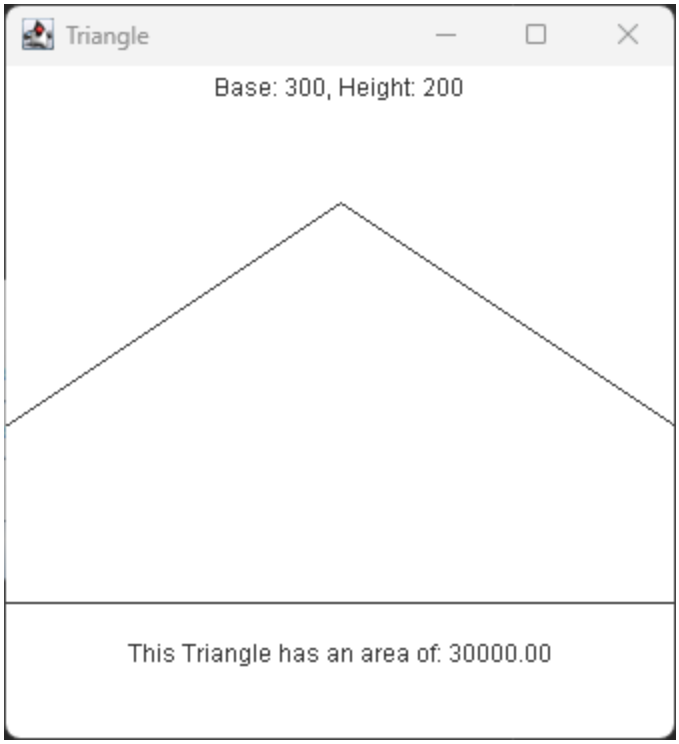
Example 3:

*For my last example, I chose **Triangle** as my desired shape, in order to demonstrate the **Area** calculations of my program.*



*I purposefully entered a rather large **Base** and **Height** for my triangle, wanting to see how it would be drawn in the frame, knowing it would be too large to display by default.*

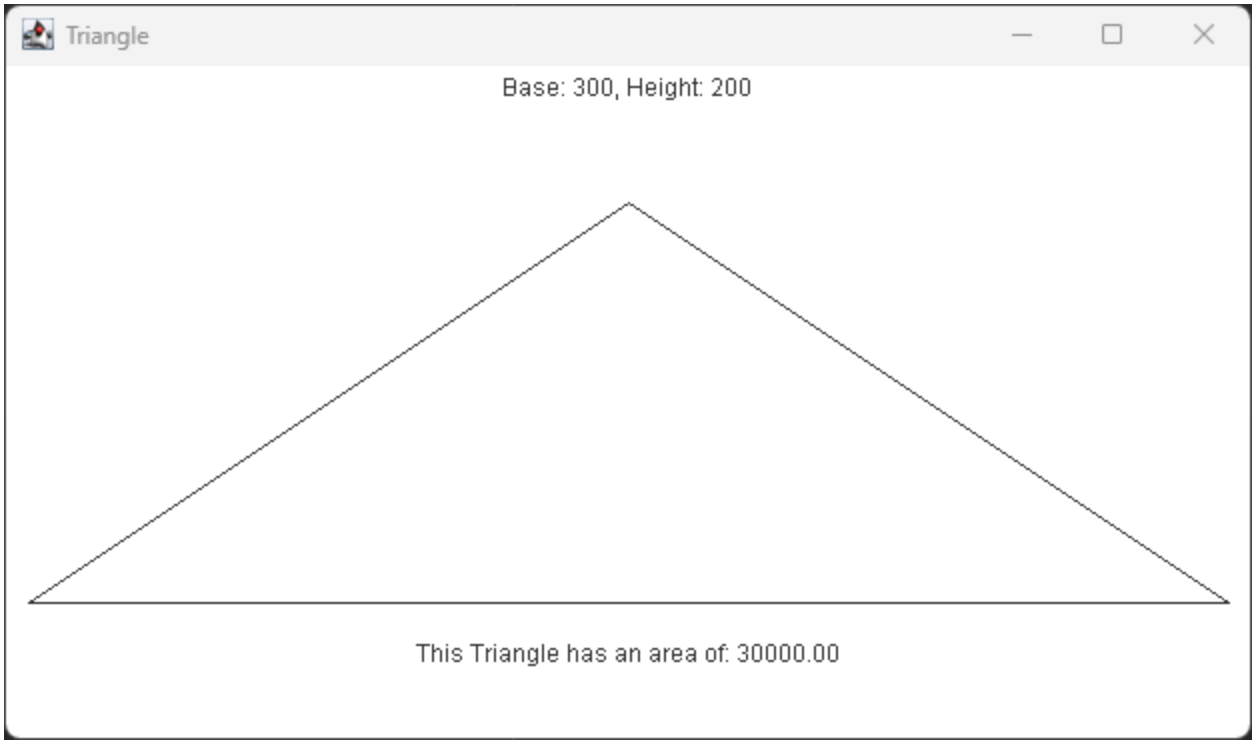
Joseph Julian
05 February 2023



As expected, the dimensions and area of my triangle are calculated correctly - however - the triangle was not fully displayed.

Expanding the window manually by dragging the sides produced the following result, correctly painting the full size of my triangle.

Joseph Julian
05 February 2023



It's worth noting that maximizing the window would not allow me to capture my screen using ***Windows Snipping Tool****, due to my code explicitly keeping this window on top of all others.*

Lessons Learned:

This project was a little difficult to overcome for a few reasons. I wanted to reuse as much code from my first project as possible, however, I was struggling to maintain proper inheritance when implementing the GUI for this program. I ended up keeping my ***Main*** *class, which essentially initialized the major component of this program,* ***Shape Factory****. Additionally, I struggled a lot with the overall design of my GUI. I wanted to improve the functionality by dynamically resizing windows, but I elected to maintain the simplicity of my program. As I become more familiar with Java and all of its GUI components, I hope to master all of the minor details in order to make my program as interactive as possible and provide the best user experience.*