Assignment 5 – Spring 2020

In this assignment we utilize Java's FX technology and object serialization to add functionalities to an existing application.

- ✓ ***Please do your own work, sharing and/or copying code and/or solution ideas with/from others will result in a grade of 0 and disciplinary actions for all involved parties. If you run into any problems and have done your best to solve them, please see me before/after class or e-mail me.***
- ✓ ***If your submission is late, you will incur a 20% deduction of your score for each late day.***

Problem Description:

Before you begin working on this assignment, download the three accompanying files ($Main.java, SampleController.java,$ and $Sample.fxml$). Crate a new $JavaFx$ project using the usual defaults as shown in class using Eclipse. Remember to set the language to $FXML$; the Project's name can be set to anything you like (See Figure 1). If you use a different IDE, please adjust your project's parameters accordingly. The package name should be $application$, the controller's name is $SampleController$, and the driver class is $Main$.

Once done correctly, copy the provided files and overwrite the ones generated by your IDE. Next launch the application, it should look like Figure 2. Do not proceed unless you complete this step and you fully test the application.

Next add the following features:

I. Title: make the title of the window $Assignment\ 5 - < your\ name >$
II. Shape: add a shape Titled Pane with two options (Circle and Square)
   - ✓ Circle is the default
   - ✓ When Square is selected, squared shapes are drawn on the canvas. To do this, use JavaFX's Rectangle class $javafx.scene.shape.Rectangle$. Use the $radius$ variable for the square's $length$ and $width$. Ensure that the color and sizes of the squares are accounted for. See Figure 3.

III. Serialize button:
   - ✓ Red background with white foreground – Figure 3
   - ✓ Click Event: for this assignment, the button will serialize the $Circle$ objects drawn on the drawing canvas as described below.

Serializing the Circle objects to disk in XML format:

- ✓ Prompt the user to enter a filename – Figure 4.
- ✓ Using Regular Expressions, ensure that the filename is valid. If the filename prompt is cancelled or the typed name does not pass the regular expression, display a message – Figure 5.
- ✓ Once the filename is accepted, write every $Circle$ object to the XML file. For this assignment, skip the $Rectangle$ objects. One way to achieve this process is as follows:
   i. Create a POJO class (e.g. $A5Shape$) with three members for $x - coordinates,$ y – coordinates, and $radius$. This is similar to the class on slide 52 in Chapter 15.
   ii. Create a container class to hold objects of type $A5Shape$ in an $ArrayList$. See slide 59 of Chapter 15.
   iii. For every drawn circle on the panel, create an object of type $A5Shape$ and add it to the $ArrayList$.
   iv. Serialize the objects to the XML file.
   v. Once the file is written, show a confirmation prompt – Figure 6
   vi. If serialization fails, print the stack trace. There is no need to display a GUI message

Grades:

Item	Points
Title	5
Shape titled pane	
GUI Component	5
Draws squares	5
Colors	5
Serialize	
GUI Component	5
Colors	5
Sizes	5
Added to canvas	5
File prompt and regex validation	25
XML Serialization	25
Confirmation prompt	10
	100

Screenshots

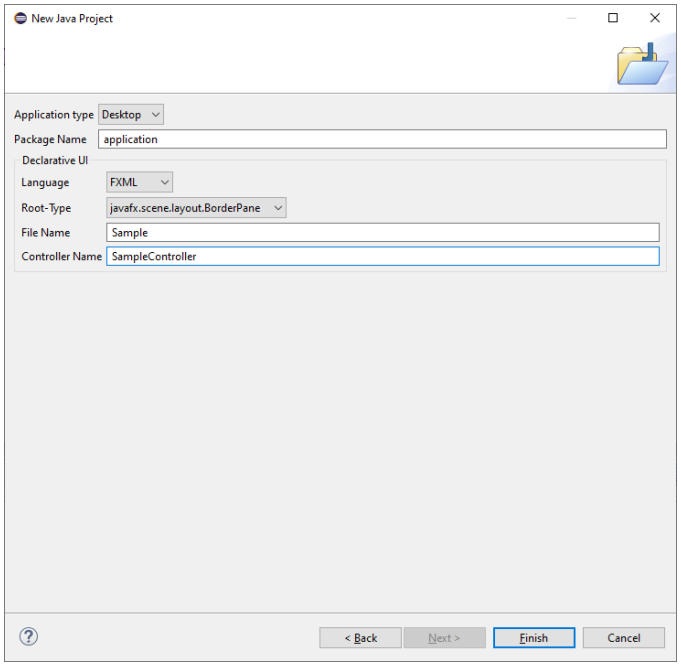


Figure 1: Eclipse JavaFX Project Wizard

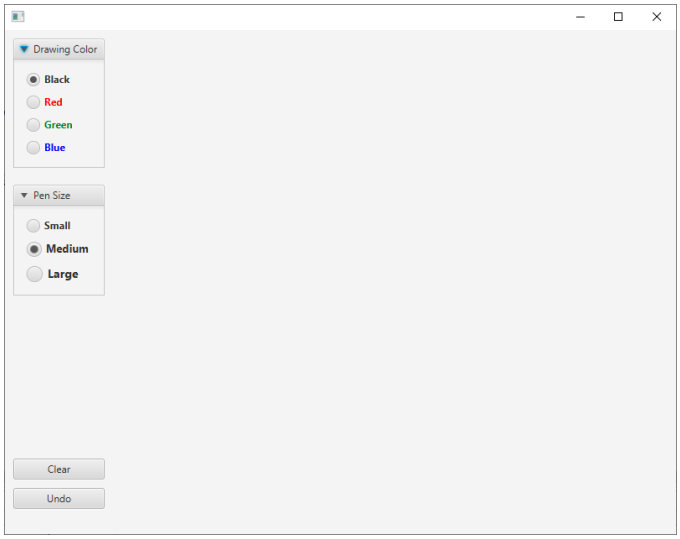
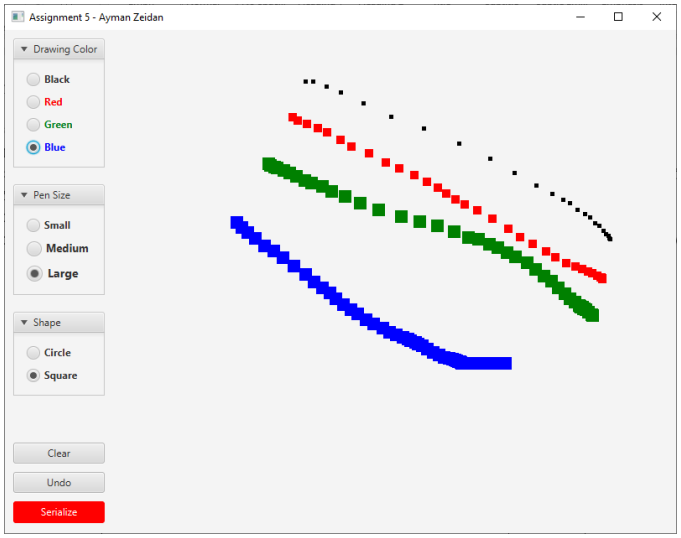


Figure 2: Initial Window



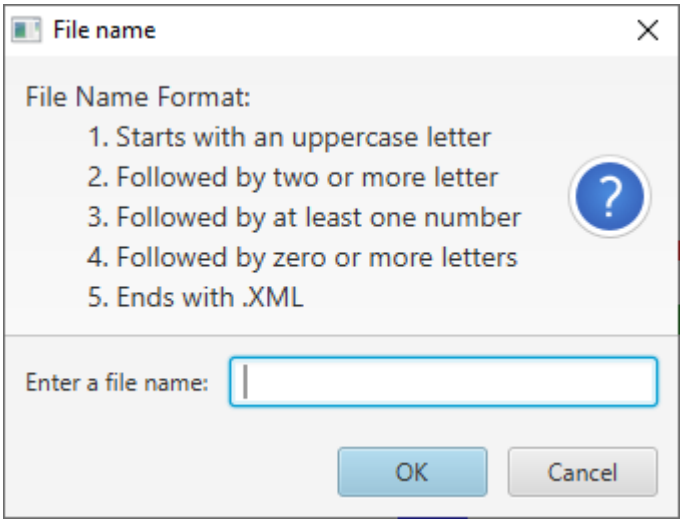


Figure 3: Different Square sizes and colors

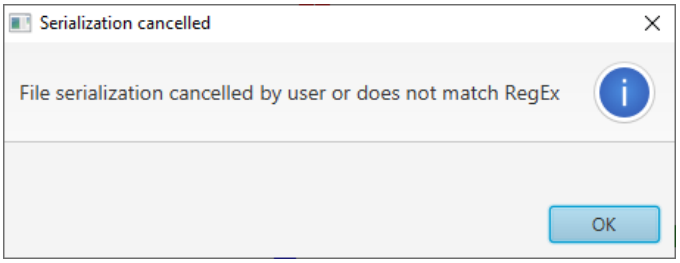


Figure 5: Invalid (or cancelled) filename prompt

Figure 4: Prompt for a file name

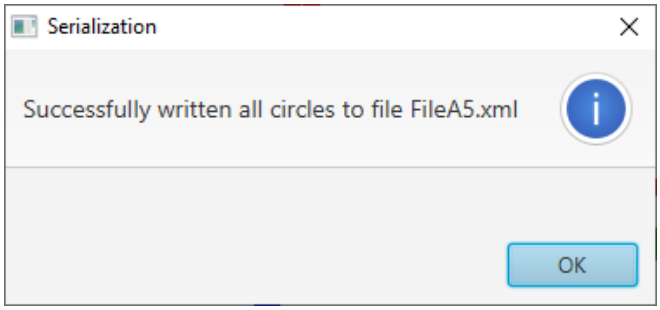


Figure 6: Confirmation message. Note that the file name is the one entered in the prompt and conforms to RegEx