**Lab 5: 100 pts total**

One of our eventual goals in this class will be for you to write a GUI-based game using JavaFx. We do have not covered enough JavaFx yet to start working on a game program, but you can start thinking about ideas (the game will be your choice).

In this lab, I will ask you to complete four “game setup” problems, for three different game types. You aren’t limited to these types for later lab(s), but I think these may be helpful for getting started.

Three types: **(1) Grid Click Location**. For example: minesweeper, battleship; **(2) Grid Piece Movement.** For example: checkers, chess, stratego; **(3) Draw or Roll.** For example: blackjack, craps (the dice game), basic high-low card draw.

1. (50pts) Please name this file **lastname\_lab5a.java**. The purpose of this problem is to demonstrate how one might set up a **Grid Click Location board**. Please implement the following in one JavaFx application file **(one application/file, but make it create two stages):**
   1. Stage 1: Display a 10x10 grid of buttons with no labels. Buttons should all be of same size, centered neatly in a grid pattern.
   2. Stage 2: Display a 10x10 grid of randomly-generated 0’s and 1’s. Each number should be centered in a text field (use TextField’s setText() method).
2. (50pts) You may name this lab file whatever you want – I will ask you to **submit the entire src directory containing source code and image files (create a zip file).**

The purpose of this problem is to show how one might set up a **Grid Piece Movement board** and/or display images for a **Draw or Roll game**. We have not covered the materials needed to make our game pieces movable yet. But we can place some image-based game pieces on a board. Please implement the following in one JavaFx application file **(one application/file, but make it create two stages):**

* 1. Stage 1: Create and display an 8x8 checkerboard (chess board) grid. Add one chess piece image to any of the squares on the board, but make sure it is properly sized and is centered within its square. For public domain images, see <http://www.pdclipart.org/> (or similar site – do a search for “public domain chess piece images”, for example). Feel free to make your own images if you’d prefer.
  2. Stage 2: There are 61 Egyptian Hieroglyphic images here: <https://www.pdclipart.org/thumbnails.php?album=45>

Write a program that will randomly choose and display three of these side-by-side. There are several ways to do this, think about how you might either create or choose filenames based on randomly-generated numbers.