Assignment 3 Starter Code

Preparation

Please run the following in the **terminal** prior to running the program

```
gradle Maker --args="img/Pineapple-Upside-down-cake.jpg 2"
gradle Maker --args="img/To-Funny-For-Words1.png 3"
```

Grid Image Maker Usage

Terminal

```
gradle Maker --args="<image to slice> <size>"
```

GUI Usage

Code

1. Create an instance of the GUI

   ```
   ClientGui main = new ClientGui();
   ```

2. Create a new game and give it a grid dimension

   ```
   // the pineapple example is 2, but choose whatever dimension of grid you want
   // you can change the dimension to see how the grid changes size
   main.newGame(2);
   ```

Depending on how you want to run the system, 3 and 4 can be run how you want

3. Insert image

   ```
   // the filename is the path to an image
   // the first coordinate(0) is the row to insert in to
   // the second coordinate(1) is the column to insert in to
   // you can change coordinates to see the image move around the box
   main.insertImage("img/Pineapple-Upside-down-cake_0_1.jpg", 0, 1);
   ```

4. Show GUI

   ```
   // true makes the dialog modal meaning that all interaction allowed is
   //   in the windows methods.
   // false makes the dialog a pop-up which allows the background program
   //   that spawned it to continue and process in the background.
   main.show(true);
   ```

Terminal

```
gradle Gui
```

Files

GridMaker.java

Summary

> This takes in an image and a dimension and makes a grid of the image

ClientGui.java

Summary

> This is the main GUI to display the picture grid.

Methods

- show(boolean modal) : Shows the GUI frame with the current state
    - NOTE: modal means that it opens the GUI and suspends background processes. Processing still happens in the GUI If it is desired to continue processing in the background, set modal to false.
- newGame(int dimension) : Start a new game with a grid of dimension x dimension size
- insertImage(String filename, int row, int col) : Inserts an image into the grid
- appendOutput(String message) : Appends text to the output panel
- submitClicked() : Button handler for the submit button in the output panel

PicturePanel.java

Summary

> This is the image grid

Methods

- newGame(int dimension) : Reset the board and set grid size to dimension x dimension
- insertImage(String fname, int row, int col) : Insert an image at (col, row)

OutputPanel.java

Summary

> This is the input box, submit button, and output text area panel

Methods

- getInputText() : Get the input text box text
- setInputText(String newText) : Set the input text box text
- addEventHandlers(EventHandlers handlerObj) : Add event listeners
- appendOutput(String message) : Add message to output text