CSE1341 - Lab 8 Assignment

Overview

In Lab 7 you added inheritance and File I/O to Slots game. In this lab, you will make additional changes to the game by adding a graphical user interface.

Notes and Pre-Lab

The focus of this lab is to construct the GUI around your existing Lab 7 code. For this reason, there will be no pre-lab! Instead, please use the included code changes and instructions on the following pages.

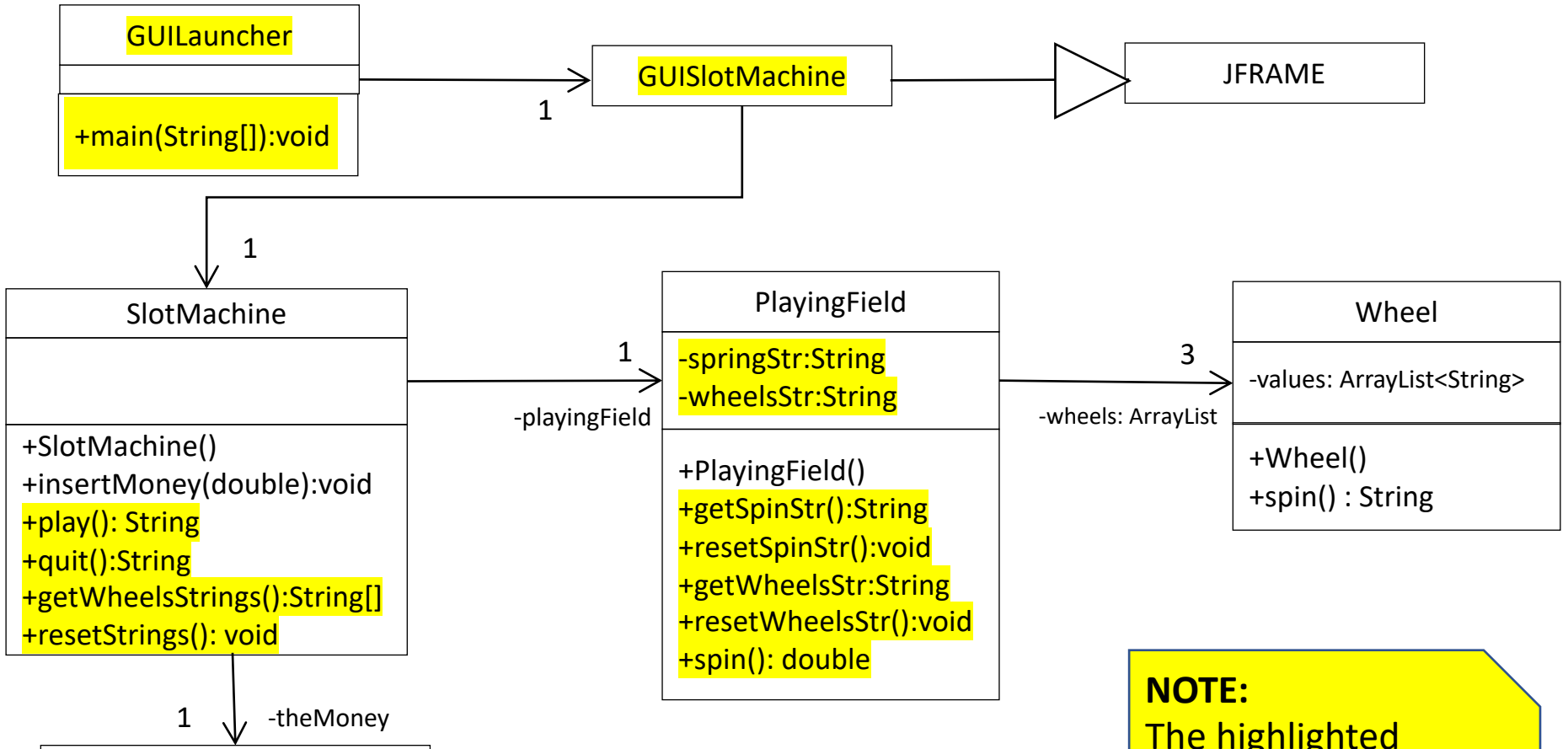
NOTES:

Each program should include comments that explain what each block of code is doing. Additionally, the programs should compile without errors, and run with the results described in the exercise. The following deductions will be made from each exercise if any of the following is incorrect or missing:

Proper formatting [5 points]
Proper names for classes and variables [5 points]
Comments [5 points per class]
Program doesn't compile [5 points for each minor error up to 5 errors provided that after fixing the errors the program compiles. If the program does not compiler after the 5 errors are fixed, partial credit will be given not to exceed 50 points]
Source code (java file) missing [10 points each]
Executable (class file) missing [10 points each]
Missing array where an array was required [5 points each]
Missing class from the design provided [10 points each]
Missing method from the design provided [5 points each]
Missing GUI component [5 points each]
Missing GUI functionality [5 points each]

Submit the *java* and *class* files via Canvas (as a single zip-file). Include a comment block at the top of each *Java* file that includes your name, student id number, and "Lab 8-Fall 2022".

This Lab is due Tuesday Nov 29 at 6:00am.



NOTE:
The highlighted classes, methods and attributes will be added or revised for this lab assignment

Lab (100 Points)

Change your Lab 7 solution with the additions and changes described on the following pages. Your output should conform to the output shown on the last page.

Code Changes

You will need to make the following code changes to allow the GUI to function as required.

Replace the Launcher with the following class:

```
import javax.swing.JFrame;

public class GUILauncher {
    public static void main(String[] args) {
        GUISlotMachine theGame = new GUISlotMachine();
        theGame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        theGame.setSize( width: 650, height: 450);
        theGame.setVisible(true);
    }
}
```

Code Changes

You will need to make the following code changes to allow the GUI to function as required

MoneyHandler.java

- Add a private data field (instance variable) called dispenseStr
- Inside the dispense() method, concatenate the String you originally displayed using the printf to dispenseStr (you can comment out or remove the printf afterwards)
- Add a get method for dispenseStr

PlayingField.java

- Add two private data fields (instance variables) called spinStr and wheelsStr
- Add two get methods: one for spinStr and one for wheelsStr
- Add a method called resetSpinStr() which will assign the empty String to spinStr
- Add a method called resetWheelsStr() which will assign the empty String to wheelsStr
- Inside the spin() method:
    - Concatenate the three Strings returned by calling spin() on each of the Wheel objects; make sure you also concatenate two dashes "-" between the Strings; assign the concatenation to wheelsStr
    - Concatenate the Strings you originally displayed using the printf and println<u>s</u> into spinStr (you can comment out or remove the println<u>s</u> afterwards)

Code Changes

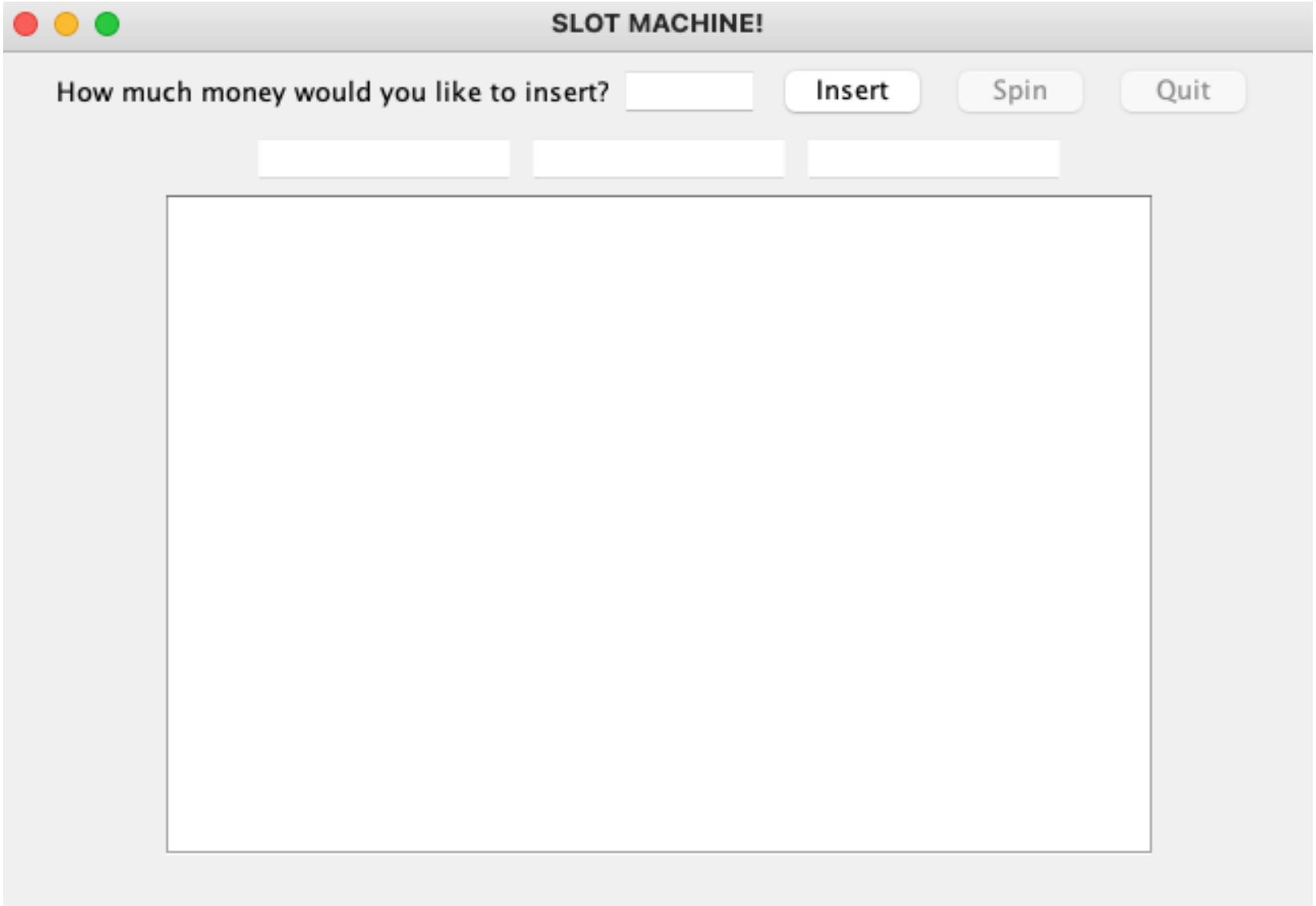
SlotMachine.java

- Modify the play() method as follows
  - Change the return type to String
  - Add a String called returnStr at the beginning of the play method
  - Concatenate the String you originally displayed using the printf to returnStr (you can comment out or remove the printf afterwards)
  - Add another String called spinStr and assign to it the String value returned by calling getSpinStr() which you added to PlayingField.java
  - Concatenate spinStr to returnStr
  - Return the returnStr
  - Everything else remains the same in the method
- Add a new method called getWheelsStrings() (see UML diagram)
  - The method returns a String[] which will contain the result of splitting on "-" the String value returned by calling the getWheelsStr() method you added to PlayingField.java
- Add a new method called resetStrings()
  - The method calls the resetSpinStr() and resetWheelsStr() you added to PlayingField.java
- Modify the quit() method as follows
  - Change the return type to String
  - Return the String returned by calling the getDispenseStr() method you added to MoneyHandler.java

Code Changes

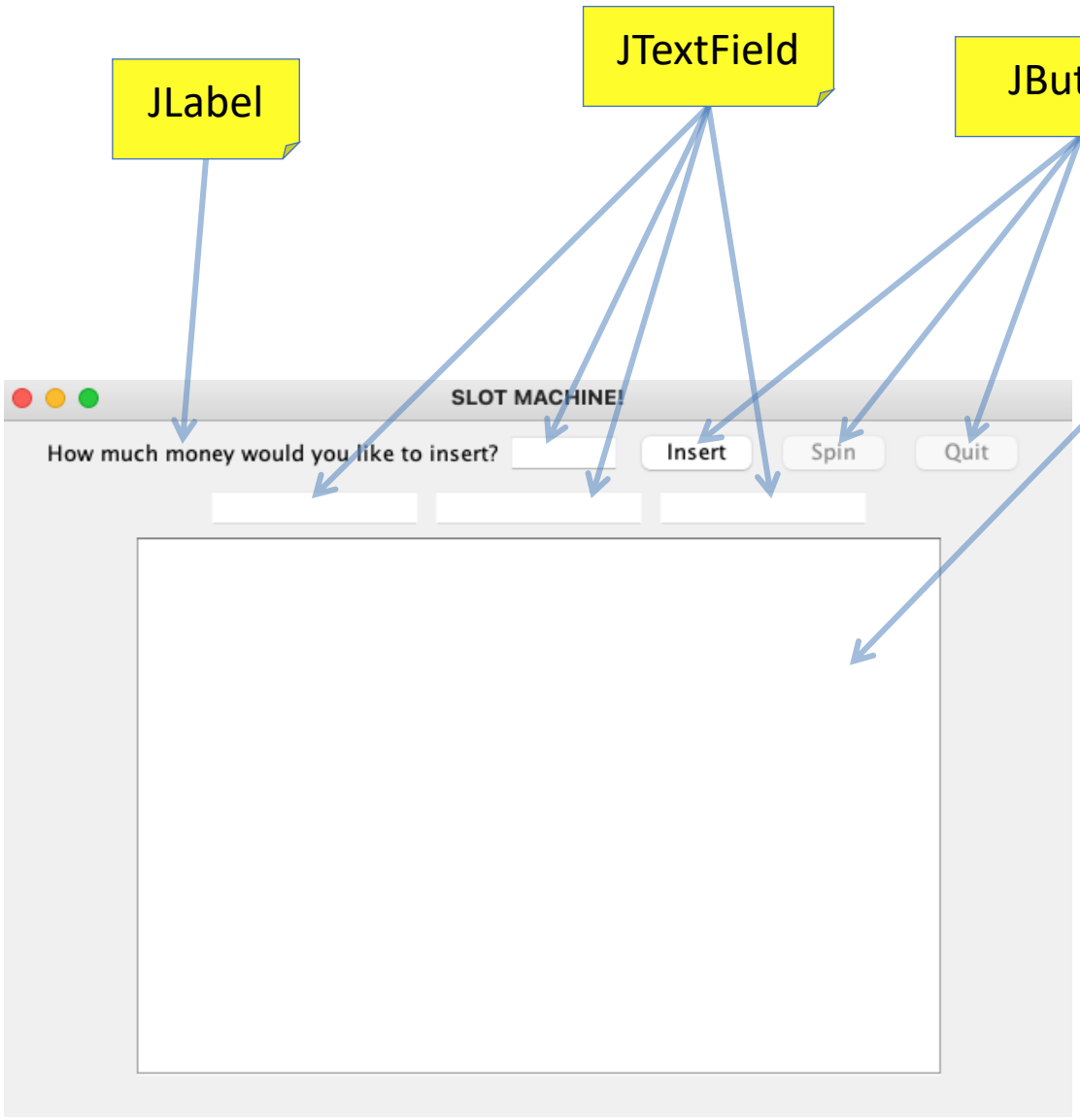
Create **GUISlotMachine.java**

In this class, you will create an instance of **SlotMachine** called slot and call the *play()* method when the user clicks the Spin button. Create the instance of **SlotMachine** with the rest of the GUI components which are shown below



GUI

Create a **JFrame** subclass named **GUISlotMachine**. Use a **FlowLayout** with recommended sizing in the instructions so your Frame matches the diagram. The **JComponents** required are depicted in the annotations below:

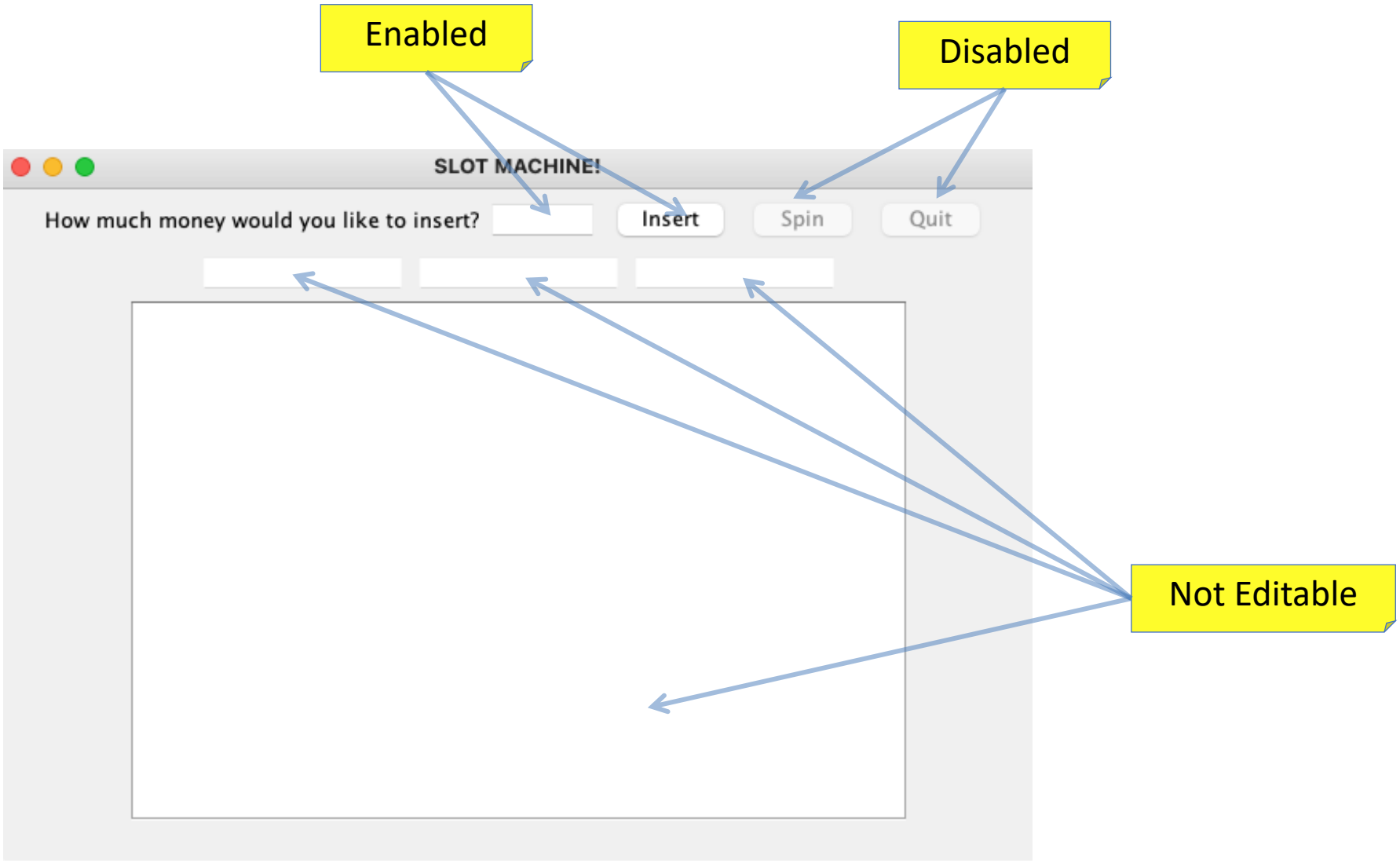


JTextArea Help

```
//Declare JTextArea and JScrollPane
//above the Constructor
JTextArea output;
JScrollPane scrollPane;

//Create and add widgets in the Constructor
output= new JTextArea(20,40);
scrollPane = new JScrollPane(output);
add(scrollPane);

//Disable editing
output.setEditable(false);

//Elsewhere – as needed
//Erase contents and replace with text
output.setText("text");
//Append contents with additional text
output.append("text");
```

GUI Behavior

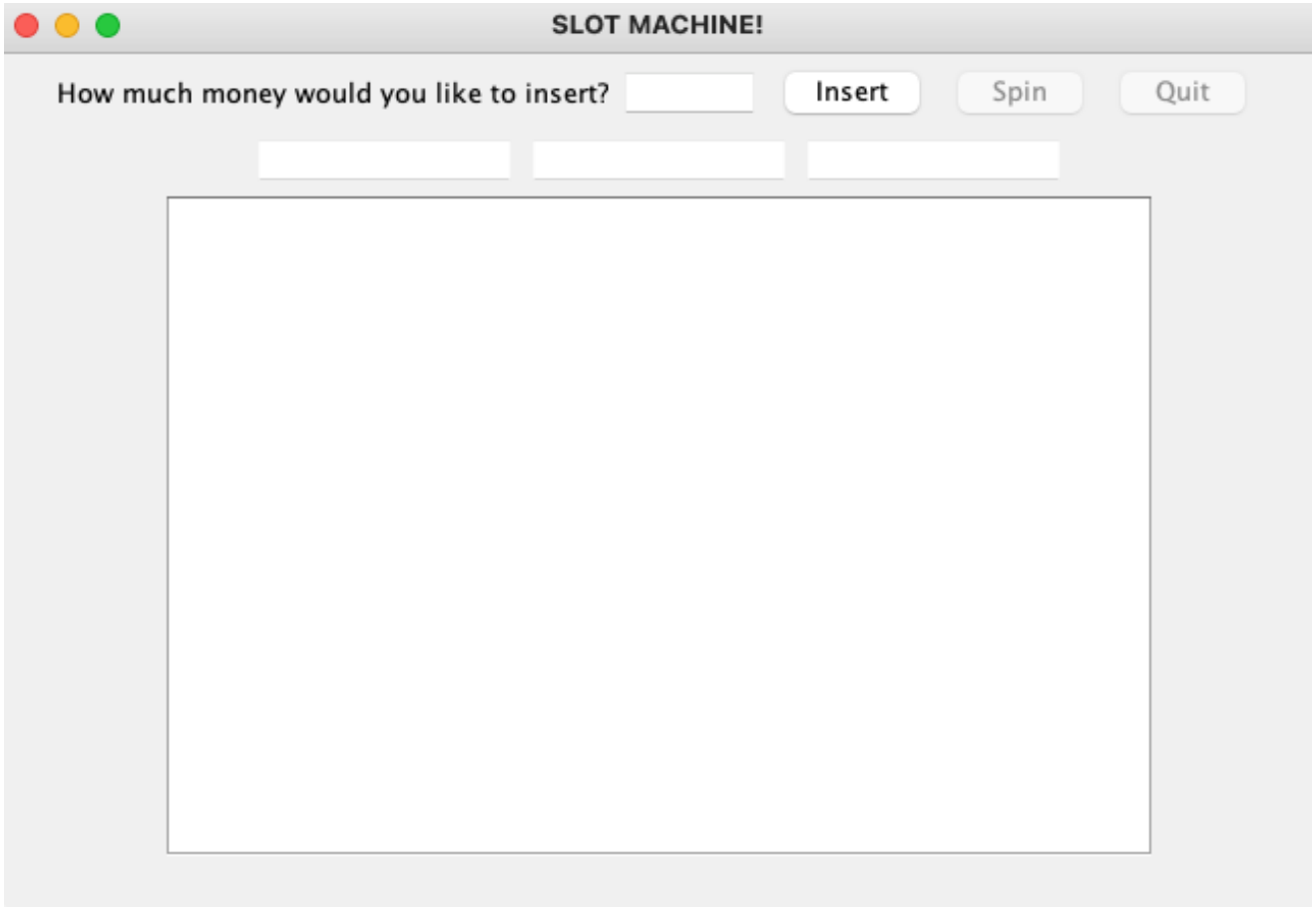
On startup, the **Spin and Quit** buttons whould be disabled, the three textfields below the buttons and the **JTextArea** should not be editable, the other widgets should be enabled and editable.



GUI Behavior

The user enters the monetary amount in the money textfield and clicks the Insert button. When the insert button is clicked,

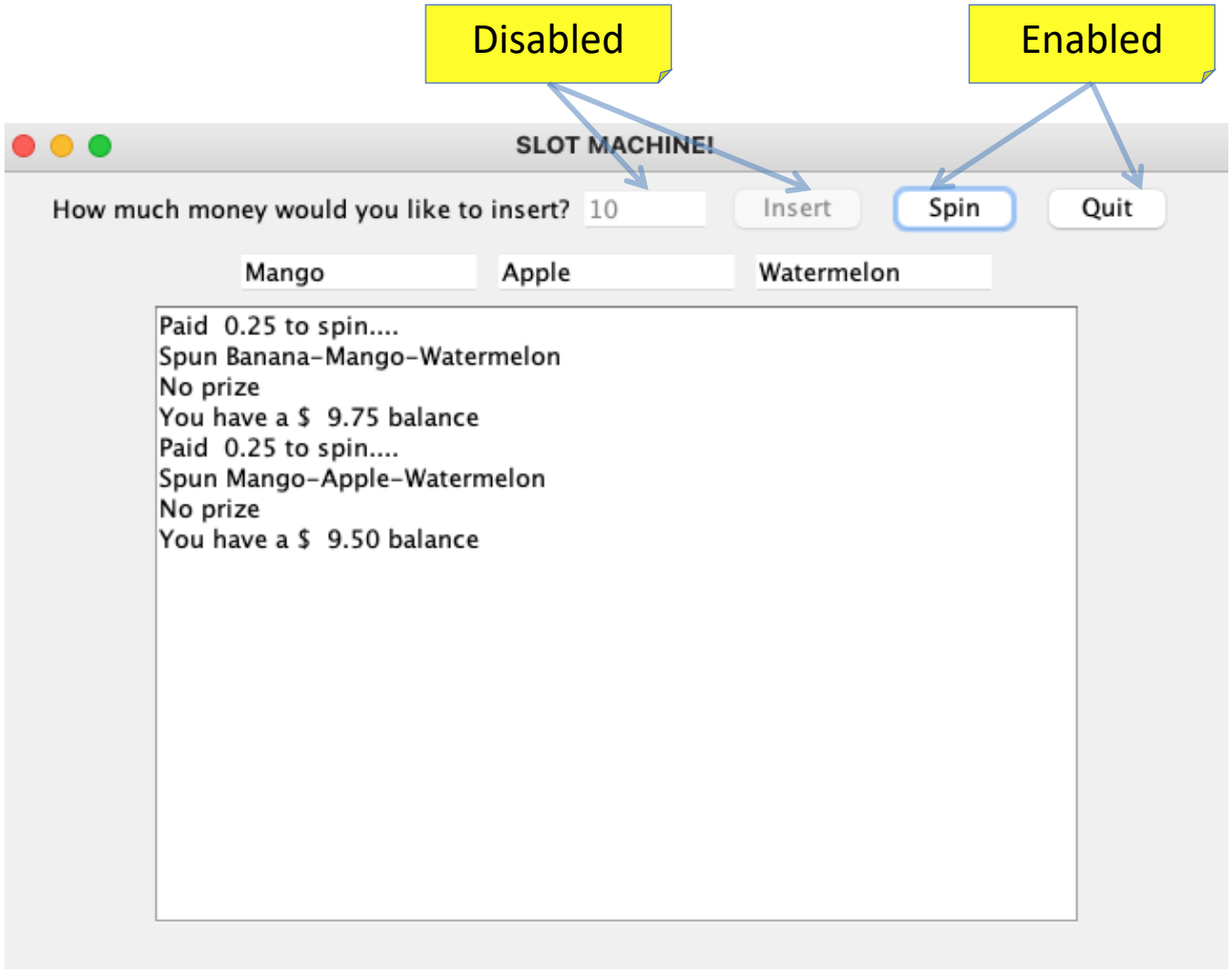
- Call the insertMoney() method on the slot instance passing it the amount entered in the money textfield
- Disable the money textfield and the Insert button
- Enable the Spin and Quit buttons



GUI Behavior (continued)

When the user click the Spin button,

- Call the play() method of the slot instance and append the returned String to the textarea
- Call the getWheelsStrings() method of the slot instance
- Assign the returned String[] array to a String[] array variable
- Set the textfields to the Strings in the String[] array one by one
- Call the resetStrings() method of the slot instance



GUI Behavior (continued)

When the user click the Quit button,

- Call the quit() method of the slot instance and append the returned String to the textarea
- Disable all the enabled GUI widgets (components)