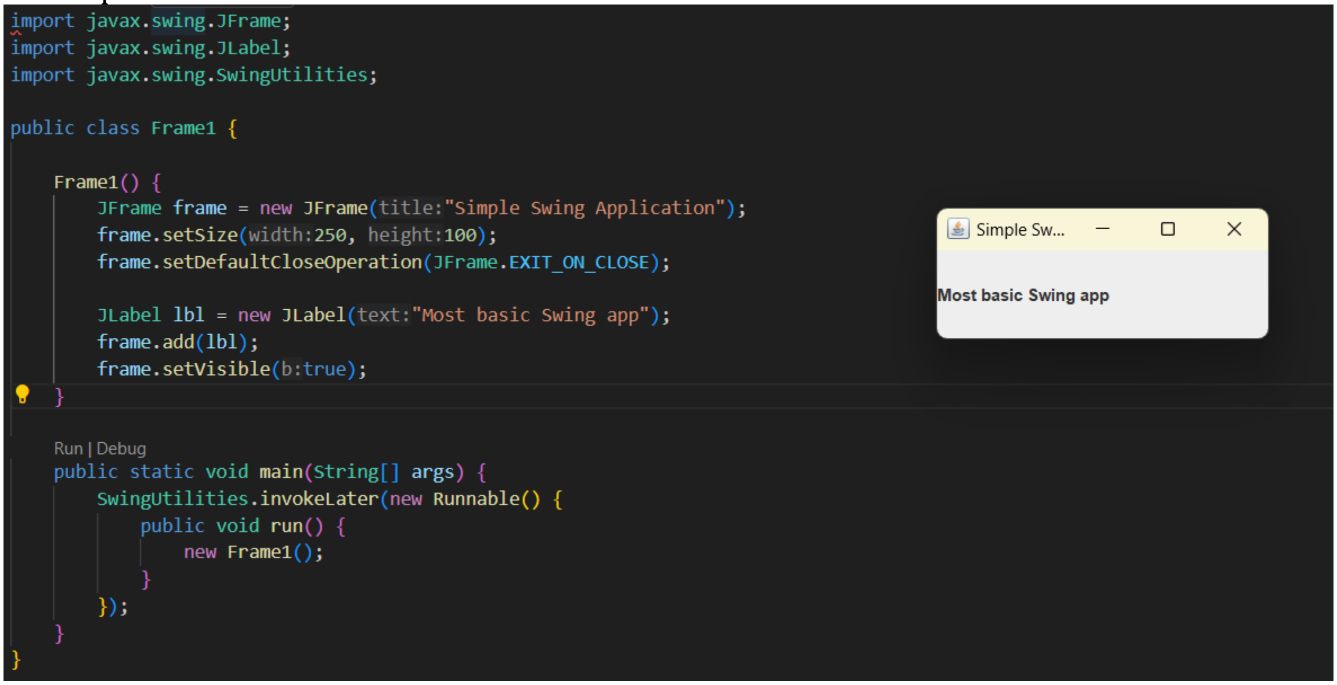
Lab Assignments – IX
MCA Semester III
CG and Java Lab (CS3307)

Before starting this assignment, let's look into Swing a bit. Observe the following code and the output.

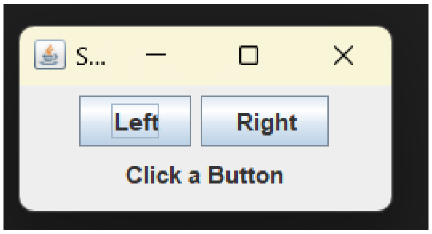


```
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.SwingUtilities;

public class Frame1 {

    Frame1() {
        JFrame frame = new JFrame(title:"Simple Swing Application");
        frame.setSize(width:250, height:100);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        JLabel lbl = new JLabel(text:"Most basic Swing app");
        frame.add(lbl);
        frame.setVisible(b:true);
    }

    Run | Debug
    public static void main(String[] args) {
        SwingUtilities.invokeLater(new Runnable() {
            public void run() {
                new Frame1();
            }
        });
    }
}
```

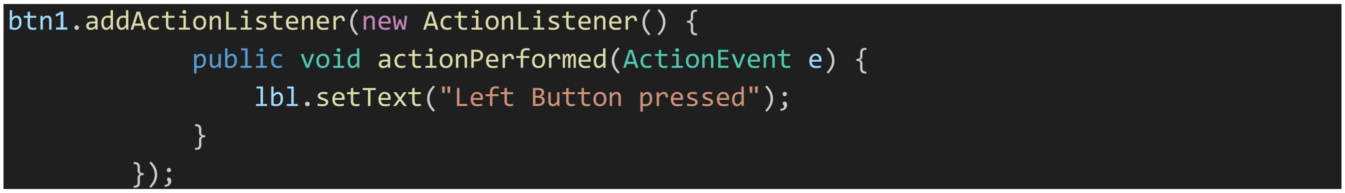
Most basic Swing app

A starting Swing app, where a JFrame is being initialized with a title of "Simple Swing Application", it's size is being set, it is being told that when the cross will be clicked the application will exit. Then, a label stating "Most basic Swing app" is being added to this frame, and then the frame is being shown. The output you can see on the right hand side as a separate floating window. We will go into details of how this works, in this and upcoming week's theory class.

1. To add multiple components, we need some layout to set. As most basic, set the layout to FlowLayout by "frame.setLayout(new FlowLayout());". Then add two buttons (JButton) stating Left and Right respectively and a label stating "Click a Button" [Hint: modify the size of frame to get this output. Otherwise, in natural flow, you may see the label on the right hand side of the Right button]



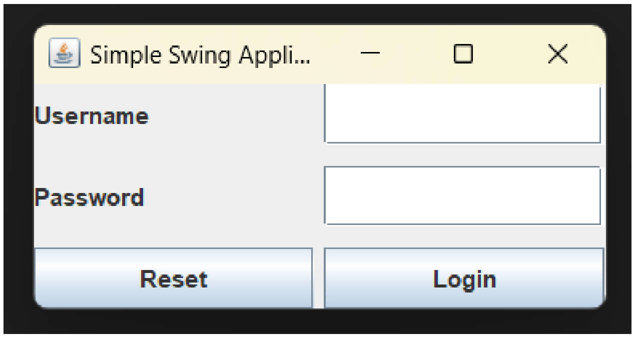
2. Now, we will add event listeners to it. The click listener to a button is added via addActionListener(ActionListener al), where ActionListener is an interface and you need to implement its actionPerformed(ActionEvent e) method. Let's look into one example, where clicking on Left Button will put "Left Button pressed" on the label [btn1 is the left button instance and lbl is the Label instance]



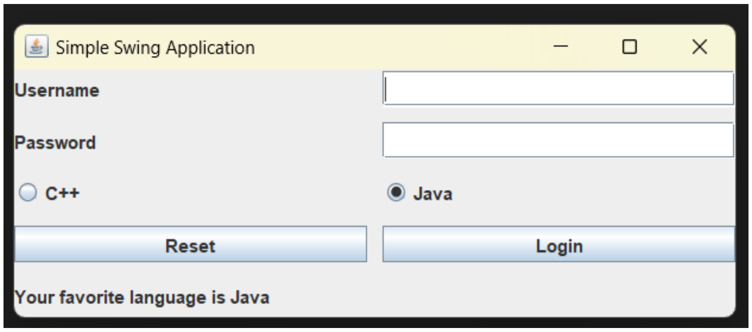
```
btn1.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                lbl.setText("Left Button pressed");
            }
        });
```

Observe the Anonymous inner class. Swing is full of it. Implement the action listener for Right Button similarly, and test your code.

3. Now, create a form as follows. The textboxes are JTextField. You need to use GridLayout now with 3 columns and 2 rows, there will be 5 pixels of horizontal gap and 10 pixels of vertical gap between the components. Clicking on Reset will clear the textboxes whereas clicking on login will print the name and password from the textbox contents, to the console.



4. Next, add a group of two radio buttons (JRadioButton) "C++" and "Java", and a label. The "Java" one will be selected by default on first launch. The label will show the statement as "Your favorite language is " followed by the text of the radio button. At a time, only one can remain selected [Use ButtonGroup]. On clicking a radio button, the label will change [add ItemListener to the radio buttons].



○ C++ ● Java

Your favorite language is Java