**Practical Sheet : Event handling**

**Creating and displaying Swing components (2 buttons)**

1. Type the following program

```java
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.JButton;

public class ButtonsFrame extends JFrame {
    public ButtonsFrame() {
        JPanel panel = new JPanel();
        getContentPane().add(panel);

        JButton buttonOne = new JButton("Say Aw ras!");
        panel.add(buttonOne);

        JButton buttonTwo = new JButton("Say Caw ras!");
        panel.add(buttonTwo);

        setSize(250, 63);
        setVisible(true);
    }
}
```

In the main method of a Launcher class add

```java
ButtonsFrame myButtonsFrame = new ButtonsFrame();
```

**Event Handling**

1. Define a new class MyTextField which extends JTextField.
   In ButtonsFrame add a MyTextField instance to the panel setting its size to 20. Execute to see the text field together with the two buttons.

2. In MyTextField implement the ActionListener interface and in the implementation of the actionPerformed(ActionEvent e) set the text of the text field to "Received message". In ButtonsFrame register buttonOne and buttonTwo as ActionListener to MyTextField:

   ```java
   buttonOne.addActionListener(myTextField);
   buttonTwo.addActionListener(myTextField);
   ```

   Execute and click on the buttons. What happens in the text field?

**Distinguishing between event sources**

3. In ButtonsFrame do:

   ```java
   buttonOne.setActionCommand("one");
   buttonTwo.setActionCommand("two");
   ```
In `MyTextField.actionPerformed()` use the passed event to distinguish between the two sources and thus to display appropriate messages for each button.

**Adapters**

After the window is closed (for the example above) the program still runs. In the following class we shall close the system after the window is closed.

Define a class `FrameTerminator` which implements the `WindowListener` interface. Rather than implementing the `WindowListener` interface directly, subclass the `FrameTerminator` from `WindowAdapter`.

Override the following method:

```java
public void windowClosing(WindowEvent e) {
    System.exit(0);
}
```

Create a `FrameTerminator` instance in a `Launcher` class and register it as a listener of windowing events with `ButtonsFrame`. 

---

*Event handling* ©