

```
import javax.swing.*;
import java.awt.event.*;

public class GUIchef
{
    // Variablendeklaration
    private ZEICHENFENSTER hintergrund;
    private JButton knopf1;
    private JButton knopf2;
    private JLabel anzeigel;
    private JLabel anzeige2;
    private boolean komponente1;
    private boolean komponente2;
    private boolean programmlauf;
    private boolean ablauf;

    /**
     * Konstruktor für einen GUI-Hintergrund mit fester Größe
     */
    public GUIchef()
    {
        // Variableninitialisierung
        hintergrund = new ZEICHENFENSTER("GUI", 800, 500);
        programmlauf = true;
        ablauf = false;
        komponente1 = false;
        komponente2 = false;

        knopf1 = new JButton("Einzelaktion");
        knopf2 = new JButton("Verlauf");

        hintergrund.komponenteHinzufuegen(knopf1, "rechts");
        hintergrund.komponenteHinzufuegen(knopf2, "rechts");

        anzeigel = new JLabel(" ");
        hintergrund.komponenteHinzufuegen(anzeigel, "unten");
        anzeigel.setText("Drücke einen der Buttons rechts am Rand! ");

        anzeige2 = new JLabel(" ");
        hintergrund.komponenteHinzufuegen(anzeige2, "unten");

        knopf1.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent e)
            {
                machwas();
                schreibeStatus("Einzelaktion erledigt!");
                komponente1 = true;
            }
        });

        knopf2.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent e)
            {
                ablauf = true;
            }
        });
    }
}
```

```
    }  
    };  
}  
  
/**  
 * Methode zur Textausgabe in einem JLabel  
 *  
 * @param text Text (String), der angezeigt werden soll  
 */  
public void schreibeStatus(String text)  
{  
    anzeige2.setText(text);  
}  
  
/**  
 * Methode zur Durchführung einer Einzelaktion  
 *  
 * @param irgendwas  
 */  
public void machwas()  
{  
    // hier steht die zugehörige Sequenz  
}  
  
/**  
 * Methode zur Durchführung eines kontinuierlichen Vorgangs  
 *  
 * @param irgendwas  
 */  
public void durchfuehren()  
{  
    //Sequenz des Ablaufs  
    schreibeStatus("Jetzt passiert was!");  
}  
  
/**  
 * Hauptmethode - diese muss zuerst aufgerufen werden, bevor das Programm in seiner  
 * gewünschten Form mit den Buttons gesteuert werden kann  
 *  
 * @param programmmlauf solange der Wert TRUE ist, läuft das Programm in einer  
 * Dauerschleife (boolean)  
 * @param ablauf löst mit Wert TRUE, der über den Button 2 gesetzt wird eine  
 * kontinuierliche Aktion aus - wird anschließend auf FALSE zurückgesetzt (boolean)  
 */  
public void main()  
{  
    while (programmmlauf)  
    {  
        if (ablauf) {  
            durchfuehren();  
            schreibeStatus("Aktion läuft!");  
            ablauf = false;  
        }  
    }  
}  
}
```