

```
// structure/BigBlob.java -- Everything in one file.
// Fred Swartz - December 2004
// Paolo Bucci - May 2012 (modified)

//      A common style of programming is to put all processing
//      in the GUI. This works Ok as long as the "model", the
//      logic, is so small that it isn't worth putting into
//      a separate class.
//
//      However, mixing model with presentation usually makes the program hard
//      to read, and the inevitable growth of the program leads to a mess.
//
//      This fails the simple Interface Independence test.
//      Could the model easily work with a command line or web interface? No.
//
//      It also fails the Model Independence test.
//      Could we easily change the model, eg, to BigDecimal? No.

import java.awt.FlowLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.math.BigInteger;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JTextField;

public class BigBlob {

    public static void main(String[] args) {
        JFrame window = new BigBlobGUI();
        window.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        window.setTitle("Simple Calc");
        window.setVisible(true);
    }

    @SuppressWarnings("serial")
    private static class BigBlobGUI extends JFrame {
        // ... Constants
        private static final String INITIAL_VALUE = "1";

        // ... Components
        private JTextField m_totalTf = new JTextField(10);
        private JTextField m_userInputTf = new JTextField(10);
        private JButton m_multiplyBtn = new JButton("Multiply");
        private JButton m_clearBtn = new JButton("Clear");

        private BigInteger m_total; // The total current value state.

        /** Constructor */
        public BigBlobGUI() {
            // ... Initialize components and model
            this.m_total = new BigInteger(INITIAL_VALUE);
            this.m_totalTf.setText(INITIAL_VALUE);
            this.m_totalTf.setEditable(false);

            // ... Layout the components.
            JPanel content = new JPanel();
            content.setLayout(new FlowLayout());
            content.add(new JLabel("Input"));
            content.add(this.m_userInputTf);
        }
    }
}
```

```
content.add(this.m_multiplyBtn);
content.add(new JLabel("Total"));
content.add(this.m_totalTf);
content.add(this.m_clearBtn);

// ... finalize layout
this.setContentPane(content);
this.pack();

// ... Listener to do multiplication
this.m_multiplyBtn.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        try {
            BigBlobGUI.this.m_total = BigBlobGUI.this.m_total
                .multiply(new BigInteger(
                    BigBlobGUI.this.m_userInputTf.getText()));
            BigBlobGUI.this.m_totalTf
                .setText(BigBlobGUI.this.m_total.toString());
        } catch (NumberFormatException nex) {
            JOptionPane.showMessageDialog(BigBlobGUI.this,
                "Bad Number");
        }
    }
});

// ... Listener to clear.
this.m_clearBtn.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        BigBlobGUI.this.m_total = new BigInteger(INITIAL_VALUE);
        BigBlobGUI.this.m_totalTf.setText(INITIAL_VALUE);
    }
});
}
```