COM6516
Object Oriented Programming and Software Design

The contents of this module has been developed by Adam Funk, Kirill Bogdanov, Mark Stevenson, Richard Clayton and Heidi Christensen

Practical 8

Calculator GUI

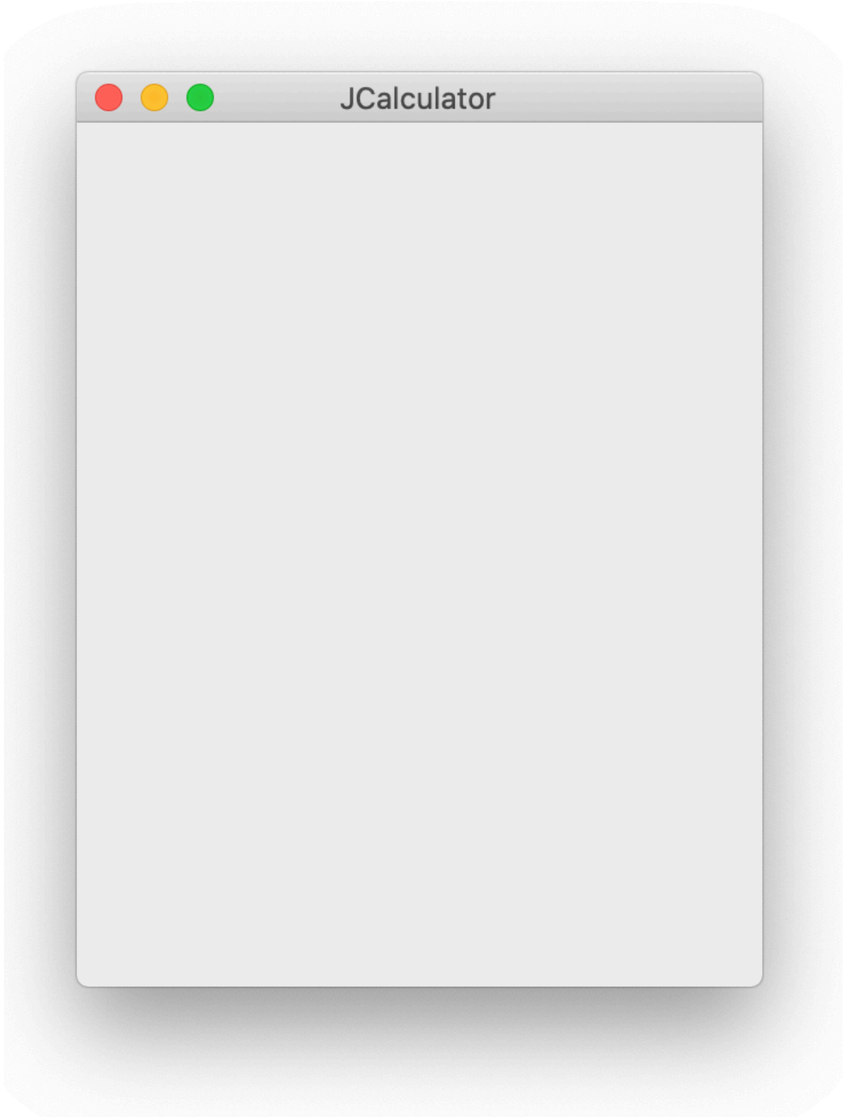
- Creating a frame
- Creating buttons
- Link buttons to action

Creating a frame

A 300x400 window:

```
public class JCalculator extends JFrame {
    public JCalculator() {
        setTitle("JCalculator");
        Toolkit tk = Toolkit.getDefaultToolkit();
        Dimension dim = tk.getScreenSize();
        setSize(300, 400);
        setLocation(new Point(dim.width/4, dim.height/4));
        Container contentPane = this.getContentPane();









    }
    …
}
```

Creating a frame

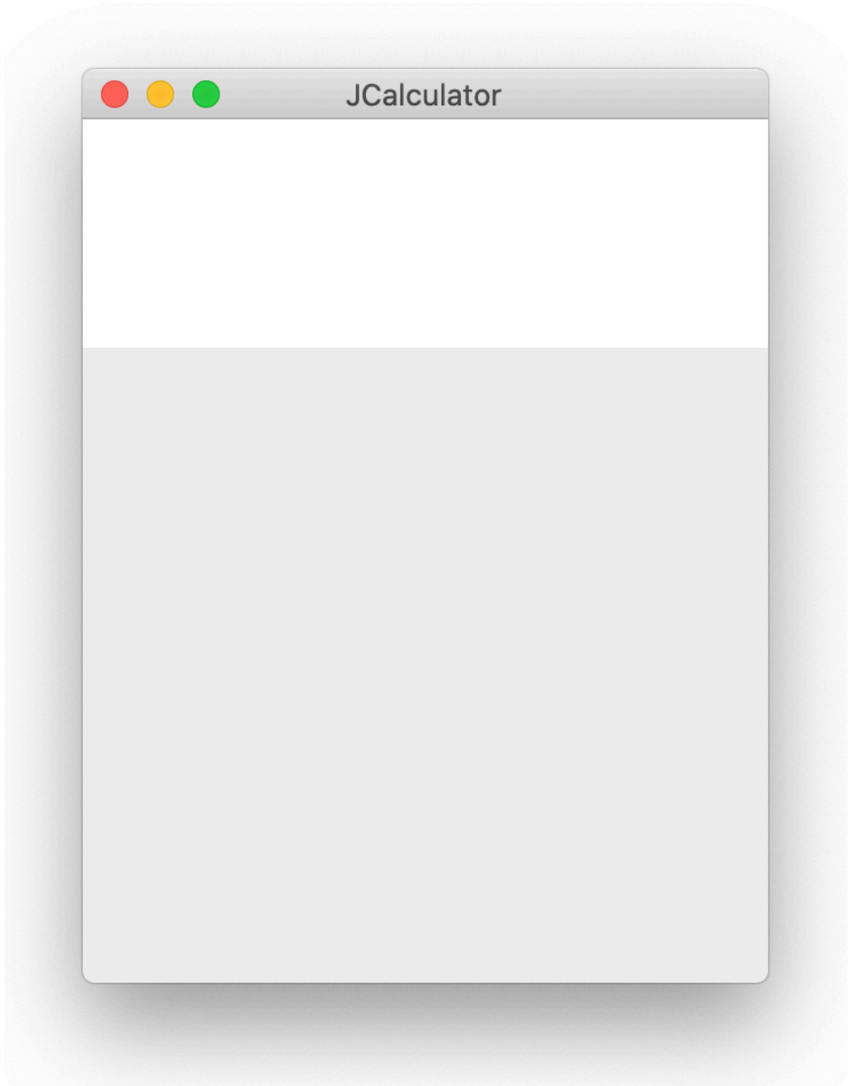


Creating a frame

Top (NORTH) panel for display:

```
public class JCalculator extends JFrame {
    public JCalculator() {
        setTitle("JCalculator");
        Toolkit tk = Toolkit.getDefaultToolkit();
        Dimension dim = tk.getScreenSize();
        setSize(300, 400);
        setLocation(new Point(dim.width/4, dim.height/4));
        Container contentPane = this.getContentPane();
        JTextArea display = new JTextArea(1, 20);
        display.setFont(new Font("Courier", Font.BOLD, 40));
        display.setEditable(false);
        display.setPreferredSize(new Dimension(300, 100));
        contentPane.add(display, BorderLayout.NORTH);



    }
    …
}
```

Creating a frame



Creating buttons

Centre panel for buttons:

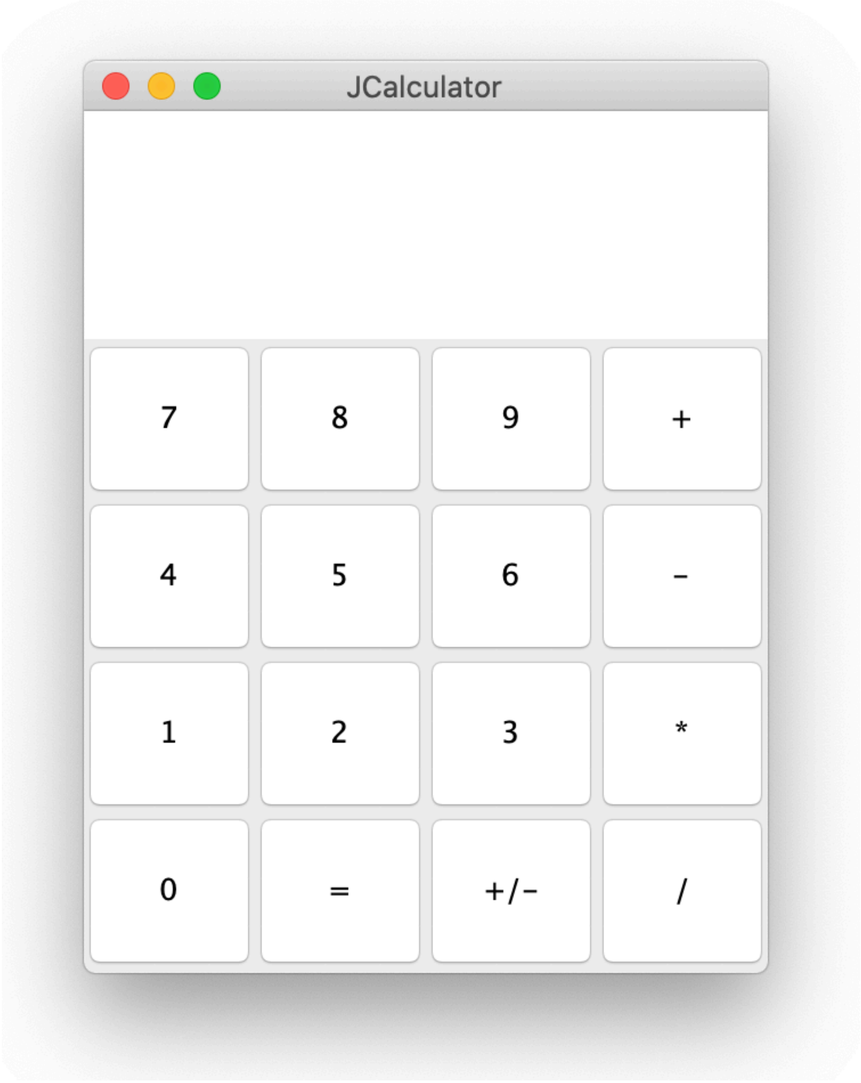
```
public class JCalculator extends JFrame {
    public JCalculator() {
        setTitle("JCalculator");
        Toolkit tk = Toolkit.getDefaultToolkit();
        Dimension dim = tk.getScreenSize();
        setSize(300, 400);
        setLocation(new Point(dim.width/4, dim.height/4));
        Container contentPane = this.getContentPane();
        JTextArea display = new JTextArea(1, 20);
        display.setFont(new Font("Courier", Font.BOLD, 40));
        display.setEditable(false);
        display.setPreferredSize(new Dimension(300, 100));
        contentPane.add(display, BorderLayout.NORTH);
        CalculatorButtons buttons = new CalculatorButtons(display);
        buttons.setLayout(new GridLayout(4, 4));
        contentPane.add(buttons, BorderLayout.CENTER);
    }
    …
}
```

Creating buttons

Centre panel for buttons:

```
class CalculatorButtons extends JPanel {
    private static final String buttonLabels[] =
        {"7", "8", "9", "+", "4", "5", "6", "-", "1", "‟2",
         "3", "*", "0", "=", "+/-", "/"};
    public CalculatorButtons(JTextArea display) {
        for (int i = 0; i < 16; i++) {
            makeButton(this, buttonLabels[i], display);
        }
    }
    private void makeButton(JPanel p, String name, JTextArea display) {
        JButton b = new JButton(name);
        p.add(b);


    }



}
```

Creating buttons



Linking buttons to action

Inner class for button action:

```
class CalculatorButtons extends JPanel {
    private static final String buttonLabels[] =
        {"7", "8", "9", "+", "4", "5", "6", "-", "1", "2",
         "3", "*", "0", "=", "+/-", "/"};
    public CalculatorButtons(JTextArea display) {
        for (int i = 0; i < 16; i++) {
            makeButton(this, buttonLabels[i], display);
        }
    }
    private void makeButton(JPanel p, String name, JTextArea display) {
        JButton b = new JButton(name);
        p.add(b);
        ButtonAction a = new ButtonAction(name, display);
        b.addActionListener(a);
    }
    private class ButtonAction implements ActionListener {
        …
    }
}
```

Linking buttons to action

Inner class for button action:

```
private class ButtonAction implements ActionListener {
    private String theLabel;
    private JTextArea theDisplay;
    public ButtonAction(String name, JTextArea d) {
        theLabel = name;
        theDisplay = d;
    }
    @Override
    public void actionPerformed(ActionEvent actionEvent) {
        // number buttons (0,1,…,9)
        …
        // +/- button
        …
        // = button
        …
        // +,-,*,/ buttons
        …
    }
}
```