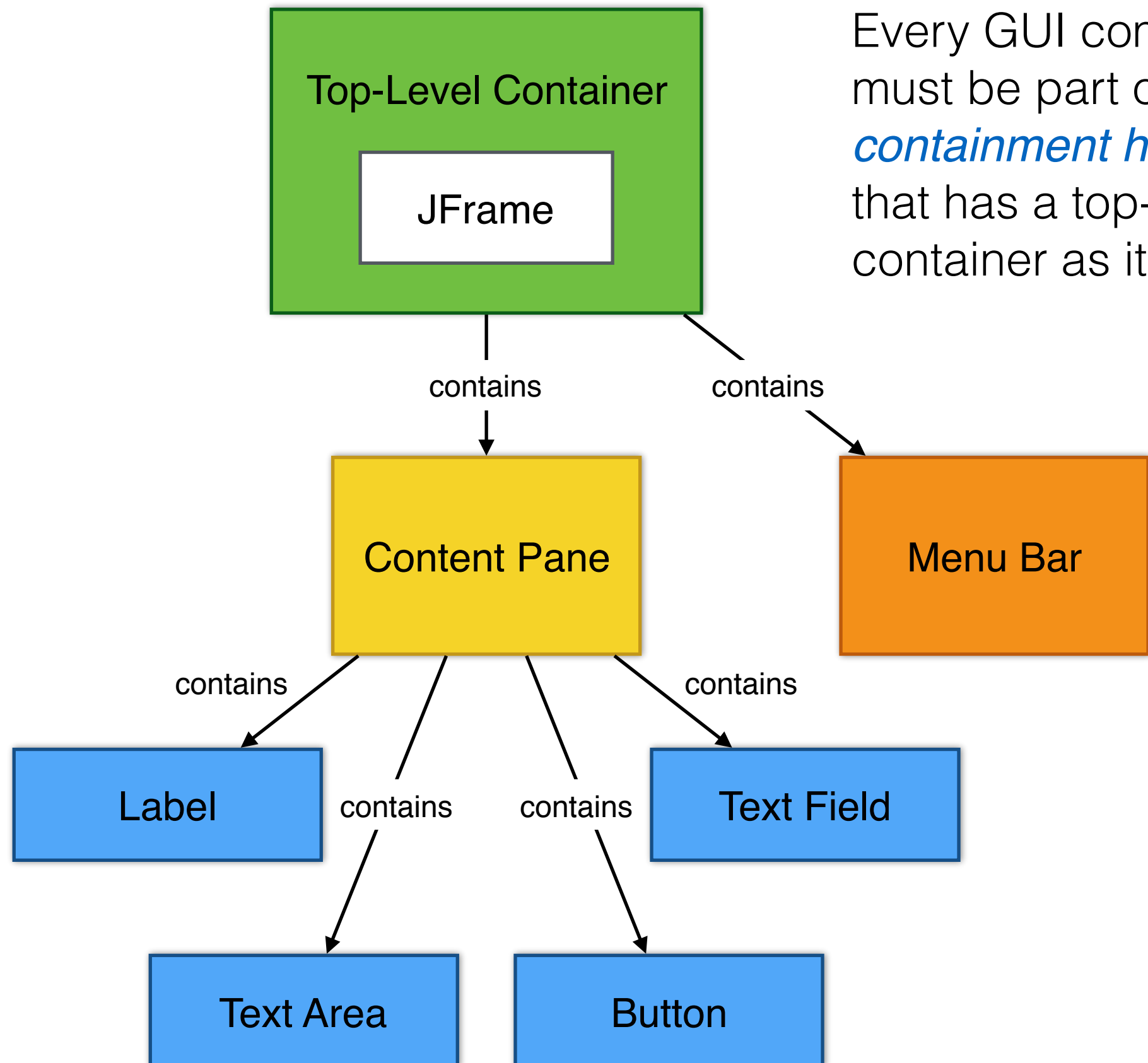
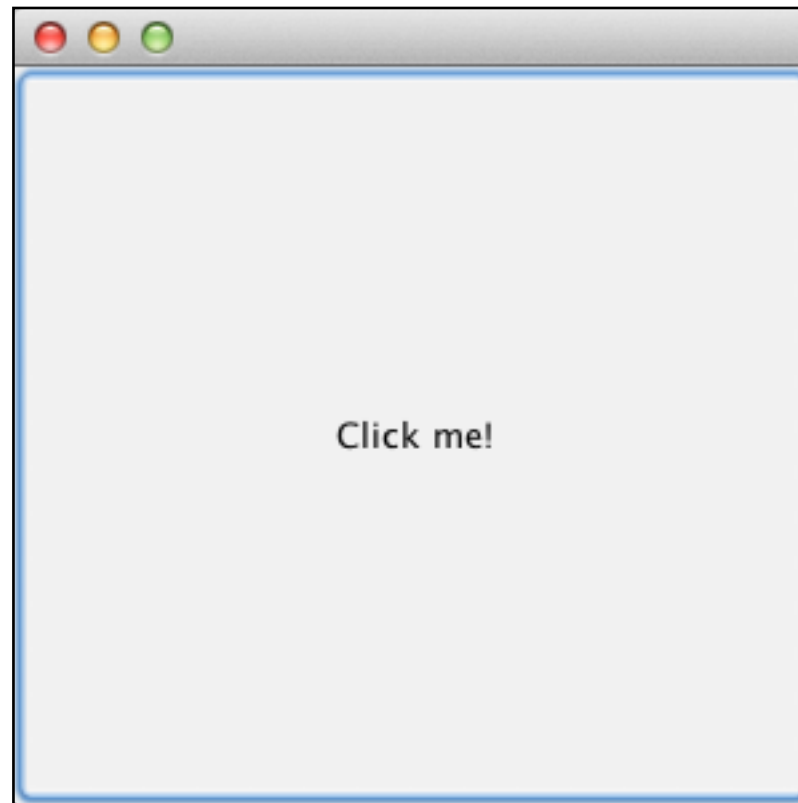


Java GUI Development



Every GUI component must be part of a *containment hierarchy* that has a top-level container as its root

ButtonDemo




Just a big button
that says "Click me!"

When you click it,
nothing happens

Head-First Java Style

Add button to
content pane

```
public class ButtonDemo {  
  
    public static void main(String[] args) {  
        JFrame frame = new JFrame();  
        JButton button = new JButton("Click me!");  
  
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        frame.getContentPane().add(button);  
        frame.setSize(300, 300);  
        frame.setVisible(true);  
    }  
}
```



Java Tutorial Style

```
public class ButtonDemo extends JPanel {  
    JButton button;
```

```
    public ButtonDemo() {  
        button = new JButton("Click me!");  
        add(button);  
    }
```

Add button to new instance, which becomes the new content pane

```
    private static void createAndShowGUI() {  
        JFrame frame = new JFrame("ButtonDemo");  
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        JComponent newContentPane = new ButtonDemo();  
        newContentPane.setOpaque(true);  
        frame.setContentPane(newContentPane);  
        frame.pack();  
        frame.setVisible(true);  
    }
```

pack or setSize

```
    public static void main(String[] args) {  
        javax.swing.SwingUtilities.invokeLater(new Runnable() {  
            public void run() {  
                createAndShowGUI();  
            }  
        });  
    }  
}
```

boilerplate

Listen for and Respond to a Button Click

```
public class ButtonDemo extends JPanel implements ActionListener {
```

```
    JButton button;
```

```
    public ButtonDemo() {
```

```
        button = new JButton("Click me!");
```

```
        button.addActionListener(this);
```

← this listens to button

```
        add(button);
```

```
    }
```

```
    public void actionPerformed(ActionEvent e) {
```

```
        button.setText("Ouch!");
```

← event response

```
    }
```