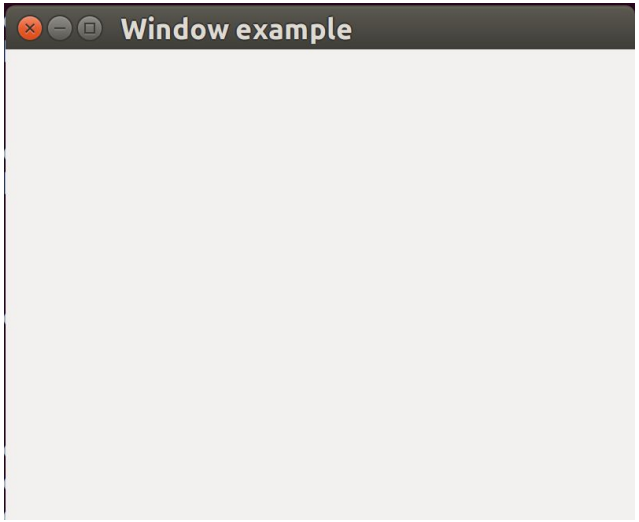
Swing - JButton

Adding buttons to the main window

An empty JFrame is not very useful

```
// In some GUI class:
```




```
window = new JFrame("Window example");

window.setSize(800,600);

window.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

window.setVisible(true);
```

Swing class rough layout

```
public class WindowExample {
  private JFrame window;

  public WindowExample() {
    initComponents();
  }

  private void initComponents() {
    window = new JFrame("Window example");
    window.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    window.setSize(800, 600);
  }

  public void show() {
    window.setVisible(true);
  }
}
```

Adding two buttons

```
public class WindowExample {
  // Our class has a JFrame and two JButtons
  private JFrame window;
  private JButton closeButton;
  private JButton toggleButton;

  public WindowExample() {
    initComponents();
  }

  private void initComponents() {
    window = new JFrame("Window example");
    window.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    window.setSize(800, 600);
    closeButton = new JButton("Close");  // Button text
    toggleButton = new JButton("Toggle");
  }

//...
```

Let's add a layoutComponents() method

```
public class WindowExample {
  private JFrame window;
  private JButton closeButton;
  private JButton toggleButton;

  public WindowExample() {
    initComponents();
    layoutComponents(); // called from the constructor
  }

  private void layoutComponents() {
    // Put the buttons somewhere in the JFrame...
  }

  //...
```

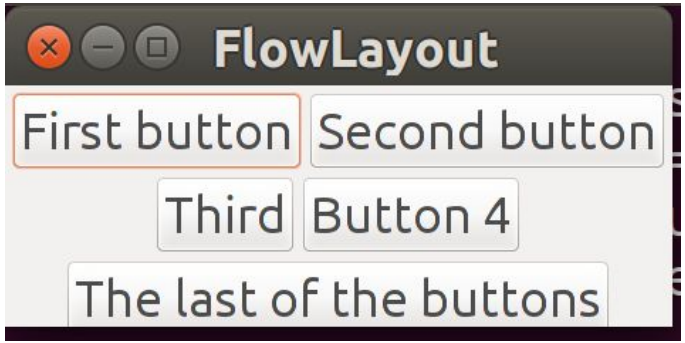
The JFrame needs a layout manager

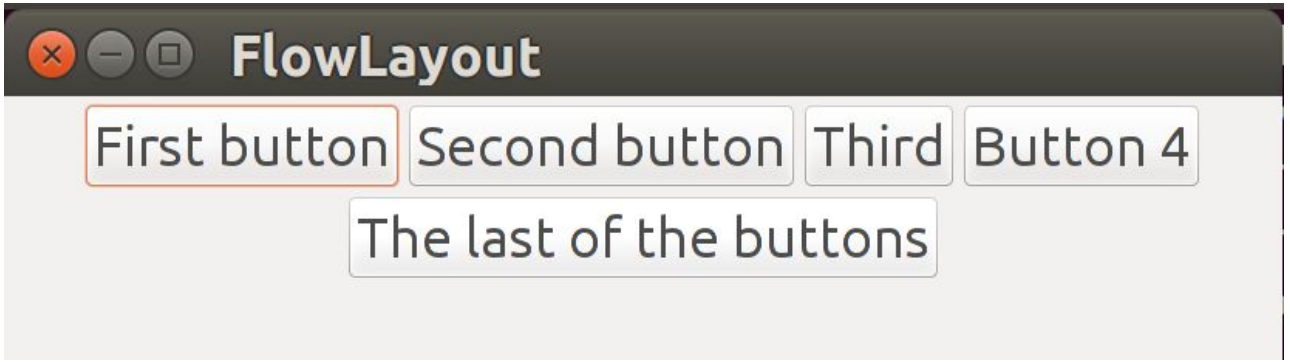
```
public class WindowExample {
  private JFrame window;
  private JButton closeButton;
  private JButton toggleButton;

  public WindowExample() {
    initComponents();
    layoutComponents(); // called from the constructor
  }

  private void layoutComponents() {
    window.setLayout(new FlowLayout()); // Will put components in sequence
    window.add(closeButton);
    window.add(toggleButton);
  }

  //...
```

LayoutManagers - FlowLayout

components

- keep their preferred size
- are placed row-by-row
- wrap if they don’t fit on a single row





The two buttons show up but don't do anything

- Actionable components (like buttons, menus, menu items etc) need an ActionListener object to encapsulate the action to perform when acted on
- ActionListener is an interface in java.awt.event
  - and has only one method declared - public void actionPerformed(ActionEvent ae) { }
- You put the code for the action in actionPerformed() using the addActionListener() method:

```
private void addListeners() { // new method, called from the constructor
  closeButton.addActionListener(e -> window.dispose());
  /* Or:
     closeButton.addActionListener(new ActionListener() {
       public void actionPerformed(ActionEvent ae) {
         window.dispose(); // closes the main window
       }
     });
   */
}
```

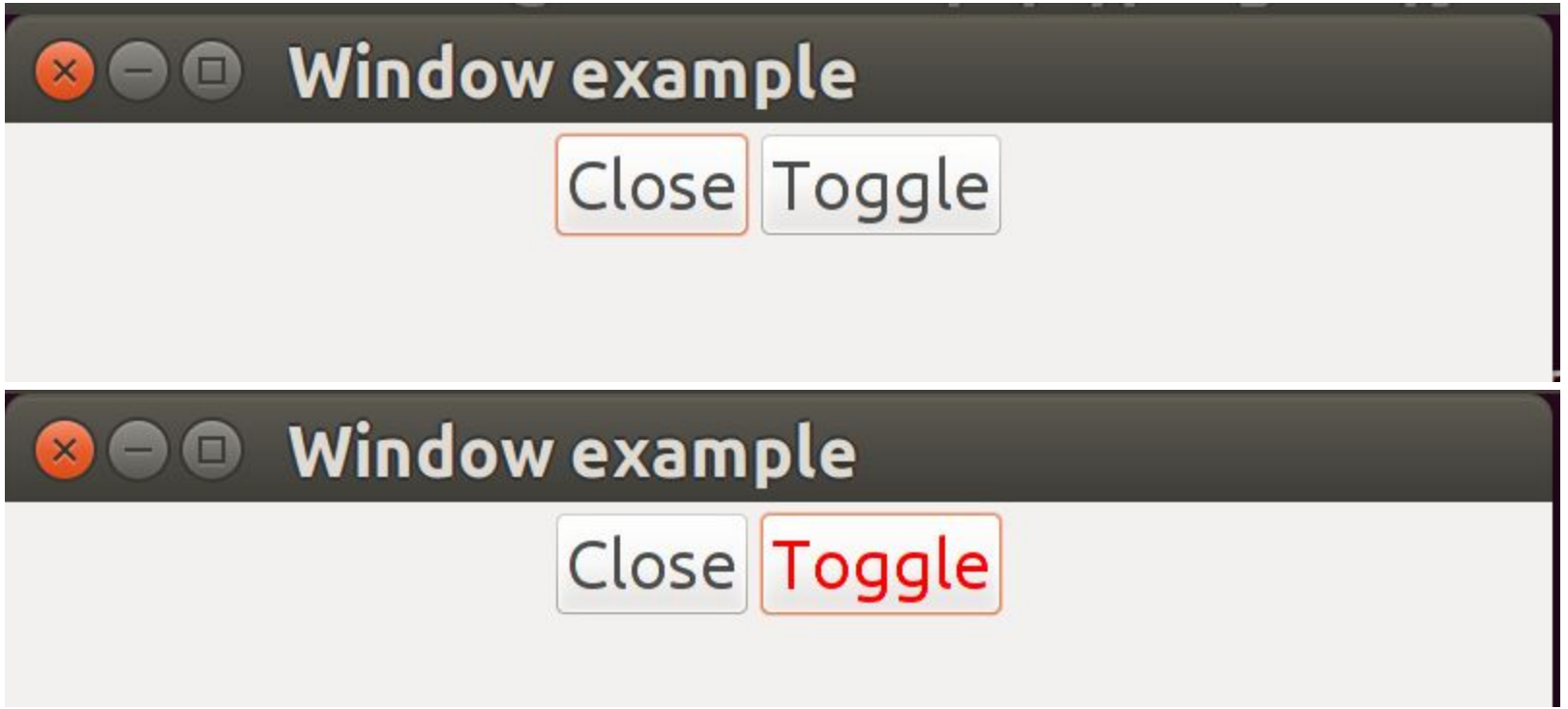
The constructor now looks like this

```
public ButtonExample() {
  initComponents();
  layoutComponents();
  addListeners();
}
```

Adding an action to the toggle button

```
// Make the toggle button change its text color back and forth
  private void addListeners() {

    closeButton.addActionListener(e -> window.dispose());

    toggleButton.addActionListener(e -> {
        if (toggleButton.getForeground() != RED) {
          toggleButton.setForeground(RED);
        } else {
          toggleButton.setForeground(BLACK);
        }
      });
  }
```

Result



Source code

```
import javax.swing.*;
import java.awt.FlowLayout;
import java.awt.event.*;
import static java.awt.Color.RED;
import static java.awt.Color.BLACK;

public class ButtonExample {

  private JFrame window;
  private JButton closeButton;
  private JButton toggleButton;

  public ButtonExample() {
    initComponents();
    layoutComponents();
    addListeners();
  }
```

Source code

```
private void initComponents() {
  window = new JFrame("Window example");
  window.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
  window.setSize(800, 600);
  closeButton = new JButton("Close");
  toggleButton = new JButton("Toggle");
}

private void layoutComponents() {
  window.setLayout(new FlowLayout());
  window.add(closeButton);
  window.add(toggleButton);
}
```

Source code

```
  private void addListeners() {
    closeButton.addActionListener(e -> window.dispose());
    toggleButton.addActionListener(e -> {
        if (toggleButton.getForeground() != RED) {
          toggleButton.setForeground(RED);
        } else {
          toggleButton.setForeground(BLACK);
        }
      });
  }

  public void show() {
    window.setVisible(true);
  }

} // end of class declaration
```

Further reading

- https://docs.oracle.com/javase/tutorial/uiswing/components/button.html
- https://docs.oracle.com/javase/tutorial/uiswing/layout/flow.html
- https://docs.oracle.com/javase/tutorial/uiswing/events/intro.html
- https://docs.oracle.com/javase/tutorial/uiswing/events/actionlistener.html