7. GUI Programming :: The *JButton* Class

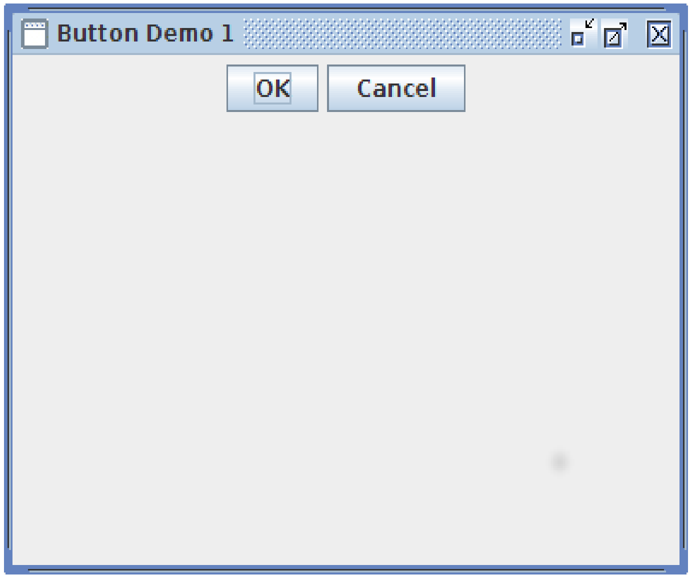
To add a button to a panel we use the *javax.swing.JButton* class. To create a *JButton* with a text label:

```
JButton butOk = new JButton("OK");
JButton butCancel = new JButton("Cancel");
```

*JButton*s are added to a *JPanel* just like *JLabel*s:

```
panel.add(butOk);
panel.add(butCancel);
```

Resulting in:



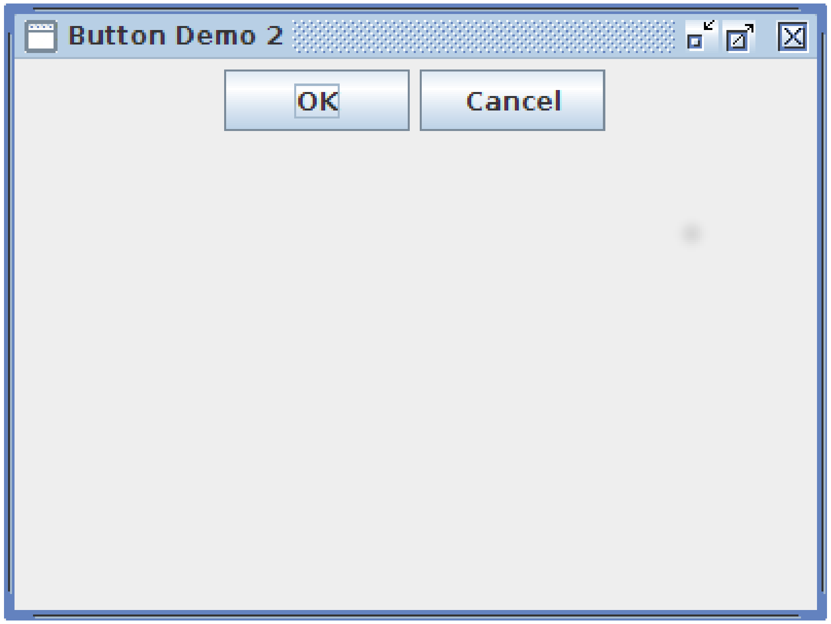
See *ButtonDemo1.java*. Note that the default layout manager (*FlowLayout*) is being used with *panel* and that the sizes of the buttons are based on the length of the text string displayed on the button face.

7. GUI Programming :: The *JButton* Class (continued)

To change the button sizes we call the *setPreferredSize*(*Dimension*) method on each *JButton*.

```
// ButtonDemo2.java
import java.awt.Dimension;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JPanel;
public class ButtonDemo2 {
  public static void main(String[] args) { new ButtonDemo2().run(); }
  public void run() {
    JFrame.setDefaultLookAndFeelDecorated(true);
    JPanel panel = new JPanel();
    JButton butOk = newButton("OK", 90, 30); panel.add(butOk);
    JButton butCancel = newButton("Cancel", 90, 30); panel.add(butCancel);
    JFrame frame = new JFrame("Button Demo 2");
    frame.setSize(400, 300);
    frame.add(panel);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    frame.setVisible(true);
  }
  private JButton newButton(String pText, int pWidth, int pHeight) {
    JButton button = new JButton(pText);
    button.setPreferredSize(new Dimension(pWidth, pHeight));
    button.setAlignmentX(java.awt.Component.CENTER_ALIGNMENT);
    return button;
  }
}
```

7. GUI Programming :: The JButton Class (continued)


Button Demo 2
OK
Cancel