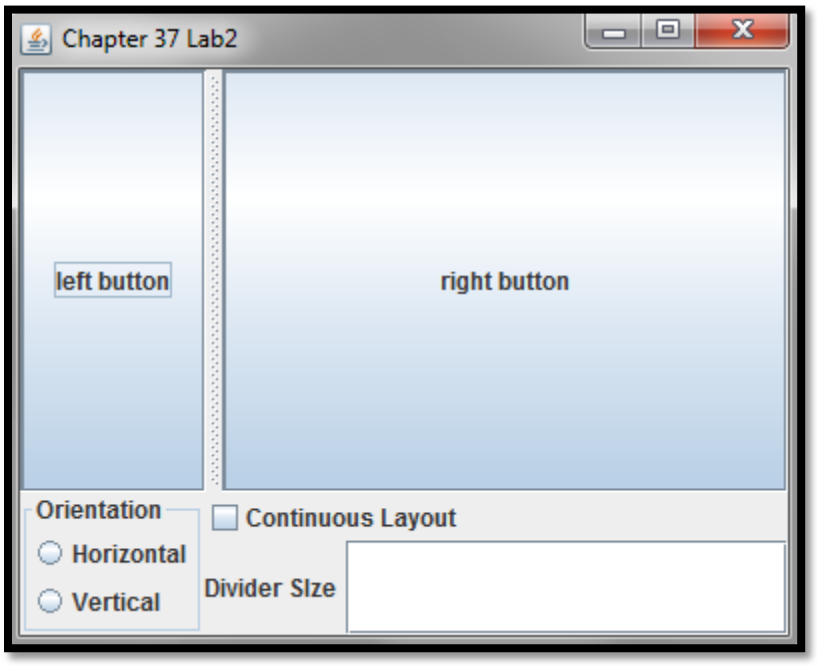
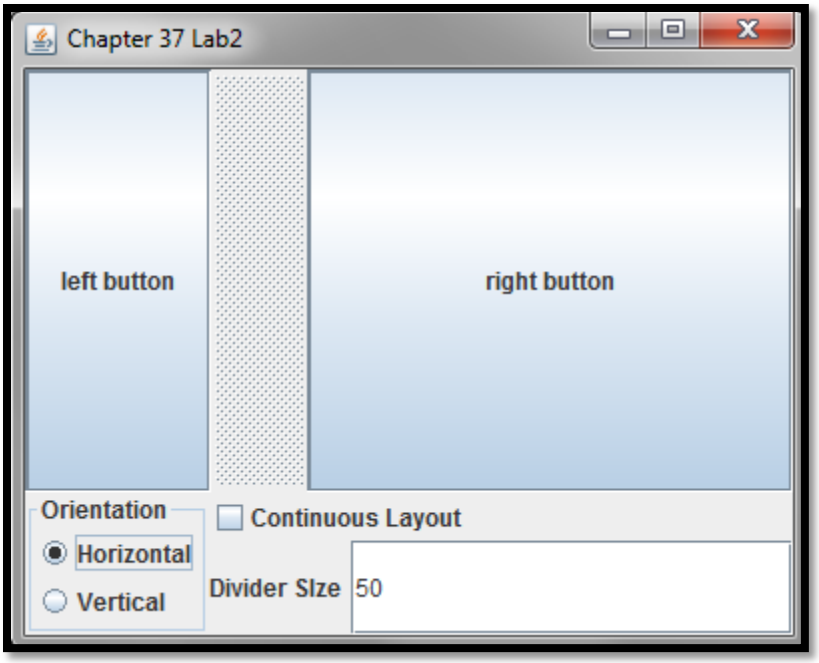
CIT 249: Java II
Chapter 37 Lab2

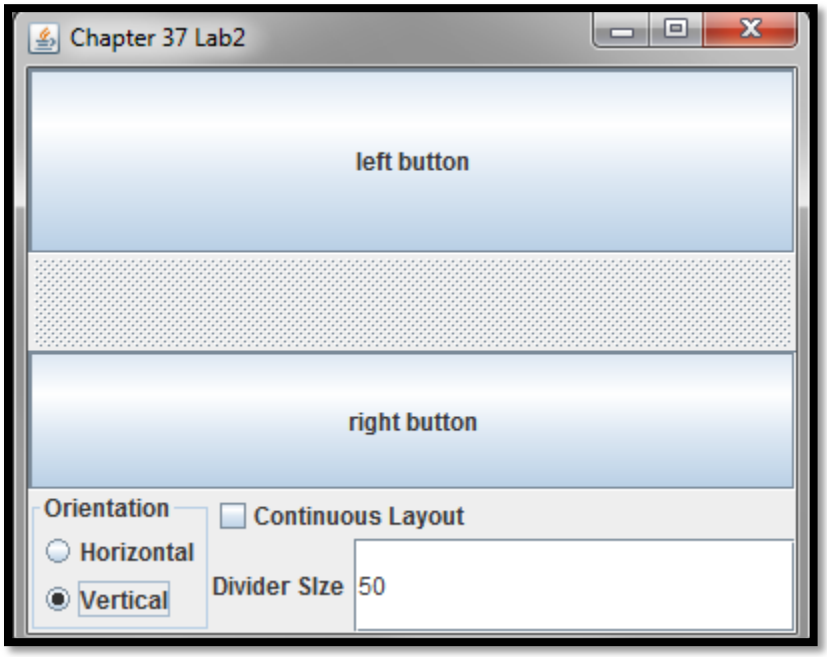
This lab will create a GUI that uses a JSplitPane. We will adjust the space between panes, either horizontally or vertically and set the amount of space. When run it will display as:



If I select Horizontal and enter a divider size of 50 it will adjust to:



If I also choose Vertical it will display as:



Let's get started:

1. Open a new document and save as Ch37Lab2.java. We include documentation as to the purpose of the program:

```
/*  Create a program that will allow the adjustment of spaces between components
    and allow the change in orientation.
*/
```

2. First we will need to import several predefined classes. Type:

```
//import necessary, predefined classes
import javax.swing.*;
import javax.swing.border.*;
import java.awt.*;
import java.awt.event.*;
```

3. Type the class header and opening brace, having the class extend the JApplet class.
4. We need to declare several of our components. Type:

```
//declare components
private JTextField jtfDividerSize;

private JPanel jPanel4;
private JPanel jPanel3;
private JPanel jPanel2;
private JPanel jPanel1;

private JCheckBox jchkContinuousLayout;

private JRadioButton jrbVertical;
private JRadioButton jrbHorizontal;
private ButtonGroup buttonGroup1;

private JSplitPane jSplitPane1;
private JLabel jLabel1;
```

- For our JRadioButtons we need to also create a ButtonGroup that will group them together so that you can select only one.

5. Next we construct the components in our constructor method. Type:

```
public Ch37Lab2()
{
   //construct components
   buttonGroup1 = new ButtonGroup();
   jPanel1 = new JPanel();
   jPanel2 = new JPanel();
   jrbHorizontal = new JRadioButton();
   jrbVertical = new JRadioButton();
   jPanel3 = new JPanel();
   jPanel4 = new JPanel();
   jLabel1 = new JLabel();
   jtfDividerSize = new JTextField();
   jchkContinuousLayout = new JCheckBox();
   jSplitPane1 = new JSplitPane();

   jPanel1.setLayout(new BorderLayout());
```

- Nothing new here. We changed the first panel's layout from the default of FlowLayout to BorderLayout.

6. Let's continue by typing:

```
   jPanel2.setLayout(new GridLayout(2, 0));

   jPanel2.setBorder(new TitledBorder("Orientation"));
   jrbHorizontal.setText("Horizontal");
   buttonGroup1.add(jrbHorizontal);
```

- We set the second panel's layout to GridLayout with 2 rows and no columns and set it have a titled border around it.
- We set the text for the first radio button to Horizontal and add it to the ButtonGroup.

7. Type:

```
//add listeners
 jrbHorizontal.addActionListener(new ActionListener()
{
  public void actionPerformed(ActionEvent evt)
  {
    jrbHorizontalActionPerformed(evt);
  }
});

jPanel2.add(jrbHorizontal);
```

- We add a listener to the horizontal radio button. If this button is selected the jrbHorizontalActionPerformed() method will be invoked.
- We add this radio button to the second panel.

8. We repeat the procedure for the vertical radio button. Type:

```
jrbVertical.setText("Vertical");
buttonGroup1.add(jrbVertical);
jrbVertical.addActionListener(new ActionListener()
{
  public void actionPerformed(ActionEvent evt)
  {
    jrbVerticalActionPerformed(evt);
  }
});

jPanel2.add(jrbVertical);
```

9. Continue by typing:

```
//add panels to the frame
 jPanel1.add(jPanel2, BorderLayout.WEST);
 jPanel3.setLayout(new BorderLayout());
 jPanel4.setLayout(new BorderLayout(5, 0));
 jLabel1.setText("Divider SIze");
 jPanel4.add(jLabel1, BorderLayout.WEST);
```

- We add the second panel to the first panel's WEST section.
- We set the layout for the 3rd and 4th panels. The 4th panel has added 5 horizontal pixels between components.
- We set the text for the label.
- We add the label to the 4th panel's WEST section.

10. Since we will be entering the amount of space between the split panes we need to add a listener to the text field. Type:

```
//add listeners
jtfDividerSize.addActionListener(new ActionListener()
{
  public void actionPerformed(ActionEvent evt)
  {
    jtfDividerSizeActionPerformed(evt);
  }
});
```

11. We need to add the text field to a panel and change the settings for the check box. Type:

```
jPanel4.add(jtfDividerSize, BorderLayout.CENTER);

jchkContinuousLayout.setText("Continuous Layout");
jchkContinuousLayout.addActionListener(new ActionListener()
{
  public void actionPerformed(ActionEvent evt)
  {
    jchkContinuousLayoutActionPerformed(evt);
  }
});
```

- Here we add the text field to the CENTER section of the 4th panel.
- We set the text to display next to the check box.
- We add a listener to the checkbox.

12. Let's finish the constructor method by typing:

```
    jPanel4.add(jchkContinuousLayout, BorderLayout.NORTH);
    jPanel3.add(jPanel4, BorderLayout.CENTER);
    jPanel1.add(jPanel3, BorderLayout.CENTER);
    add(jPanel1, BorderLayout.SOUTH);
    add(jSplitPane1, BorderLayout.CENTER);
  }
```

- We add the check box to the 4th panel's NORTH section.
- We add the 4th panel to the 3rd panel's CENTER.
- We add the 3rd panel to the first panel's CENTER.
- We add the 1st panel to the frame's SOUTH.
- We add the split pane to the frame's CENTER.

13. Now for our methods that listen to the different components when they are selected. Type:

```
  private void jtfDividerSizeActionPerformed(ActionEvent evt)
  {

    jSplitPane1.setDividerSize(new Integer(jtfDividerSize.getText().trim()).intValue());

  }
```

- This is for the text field. We take the integer value of what is typed in the field and trim it of any white space.
- We set the split pane's divider to this value.

```
  private void jchkContinuousLayoutActionPerformed(ActionEvent evt)
  {
      if (jchkContinuousLayout.isSelected())
          jSplitPane1.setContinuousLayout(true);
      else
          jSplitPane1.setContinuousLayout(false);
  }
```

- We set the split panel to have continuous layout if selected.

```
private void jrbVerticalActionPerformed(ActionEvent evt)
{
   jSplitPane1.setOrientation(JSplitPane.VERTICAL_SPLIT);
}
```

- We set the panel to a vertical split if the jrbVertical radio button is selected.

```
private void jrbHorizontalActionPerformed(ActionEvent evt)
{
   jSplitPane1.setOrientation(JSplitPane.HORIZONTAL_SPLIT);
}
```

- To a horizontal split if the jrbHorizontal radio is selected.

14. Our main method is pretty much the same as we have been writing it. Type:

```
public static void main(String[] args)
{
   Ch37Lab2 applet = new Ch37Lab2();
   JFrame frame = new JFrame();
   frame.setTitle("Chapter 37 Lab2");
   frame.add(applet, BorderLayout.CENTER);
   applet.init();
   applet.start();
   frame.setSize(400,320);
   frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
   frame.setLocationRelativeTo(null); // Center the frame
   frame.setVisible(true);
}
```

15. Finally close the class.
16. Compile the program and fix any errors if necessary.
17. Run the program and test it out.
18. Compress all files into a single zip or rar file and submit it.