121 Discussion #13 12/6/07
Announcements:
Project 7 up – demo required for hand-in!
Last OWL assignment now available.
TA Office hours: (LGRT 213): M 12:30 - 2:30; Tu 1-5; W 3-5; Th 10-11, 2:45-3:45; F 12:30 - 4:30

Topics for this week – mouse interactions; layout managers.
Example below: code does this: you click, you get 5 concentric circles centered at click location.

```java
import java.awt.*; import javax.swing.*;
import java.awt.event.*;
public class CircleClicker extends JPanel implementsMouseListener{
    private int newX = -200; // keeps initial, accidental pattern off screen
    private int newY = -200;
    private int gap = 5; int size = 20;
    public CircleClicker(){
        addMouseListener(this);
    }
    public void paintComponent(Graphics g){
        int x, y, width, height;
        for(int j = 0; j < 5; j++){
            x = newX - (size + j*gap)/2;
            y = newY - (size + j*gap)/2;
            width = size + j*gap;
            height = size + j*gap;
            g.drawOval(x,y,width, height);
        }
    }
    public void mouseClicked(MouseEvent e){
        newX = e.getX(); newY = e.getY();
        repaint();
    }
    public void mouseEntered(MouseEvent e){}
    public void mouseExited(MouseEvent e){}
    public void mousePressed(MouseEvent e){}
    public void mouseReleased(MouseEvent e){}
}
public static void main(String[] args){  // driver
    DisplayWindow d = new DisplayWindow();
    CircleClicker p = new CircleClicker();
    d.addPanel(p);
    d.showFrame();
}
```

(a) How would you change CircleClicker so that you get a random number of rings from 1 – 5?
(b) How would you change the code so that the number of rings at a click is provided via a JTextField (so that if you enter an 11 in the text field and then click, you get 11 rings, and so forth)? (c) Could you do this with a slider? How? (d) How would you add a quit button?

Last programming assignment (#7) A sample slider:

```java
import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
import javax.swing.event.*;
public class BabySliderPanel extends JPanel implements ChangeListener{
    private JSlider sizer = new JSlider(); // make slider object: sizer
    private int val = 10;
```
final int MAX = 200;

public BabySliderPanel()
{
    setLayout(new BorderLayout());
    add(sizer, BorderLayout.WEST);
    sizer.setOrientation(JSlider.VERTICAL); // default: horizontal
    sizer.addChangeListener(this);
    sizer.setMinimum(10);
    sizer.setMaximum(MAX);
    sizer.setValue(10);
    sizer.setMajorTickSpacing(30);
    sizer.setMinorTickSpacing(5);
    sizer.setPaintTicks(true);
    sizer.setPaintLabels(true);
    setBackground(Color.white);
}

public void paintComponent(Graphics g)
{
    super.paintComponent(g);
    g.setColor(Color.red);
    g.fillOval(150, 100, val, val);
    g.setColor(Color.blue);
    g.fillOval(300, 100, MAX-val, MAX-val);
}

public void stateChanged(ChangeEvent e)
{
    // notice that e is not used in body of method – but parameter required by interface
    val = sizer.getValue();
    repaint();
}

Compare this with CircleClicker in the following dimensions: interfaces; layout; components added; listener
mechanism; driver class. How would you place the slider horizontally, at the top? How would you keep the two circles
exactly the same size? How would you set this up so that red and blue are on different sliders, red vertically at left,
blue vertically at right. Can you design this so that each slider controls its circle’s size independently?