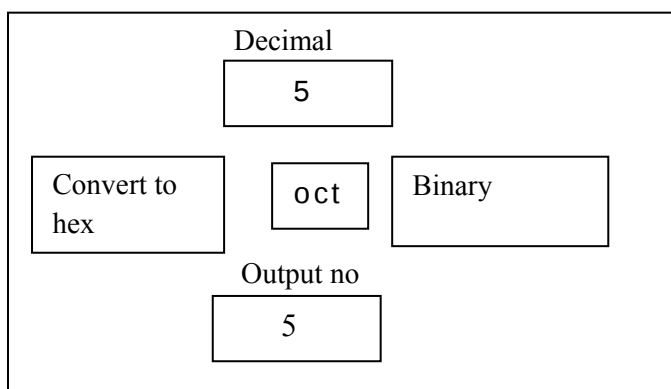
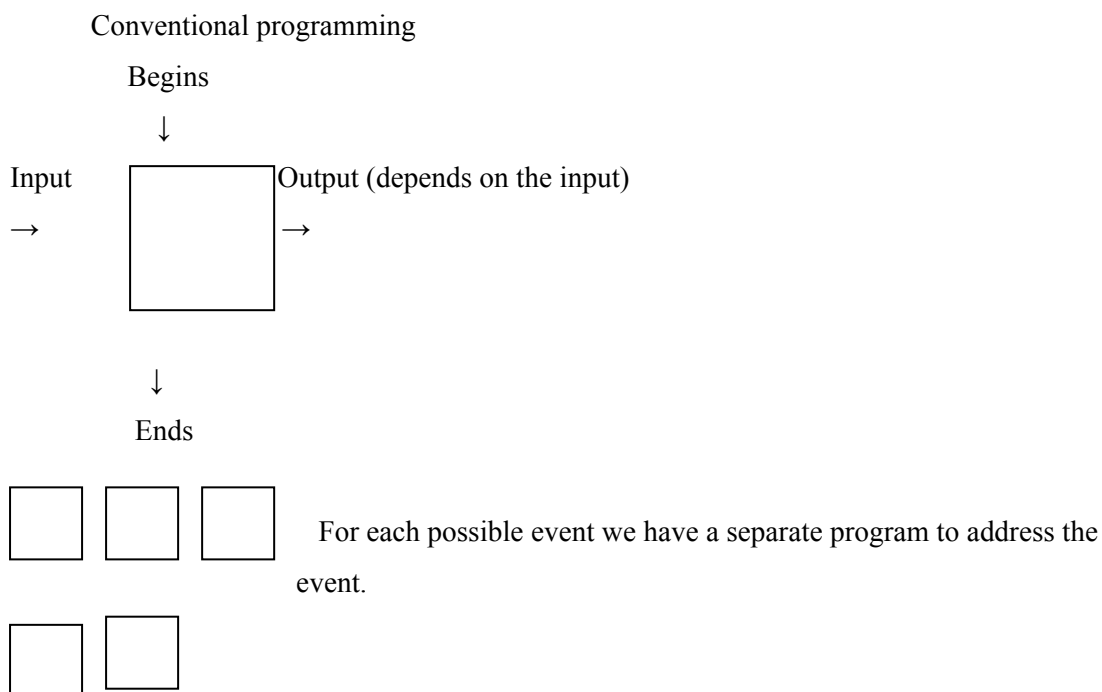


LECTURE NOTES AND EXAMPLES FOR WEEK 12

April 29 & May 1, 2008

**GUI: GRAPHICAL USER INTERFACE**

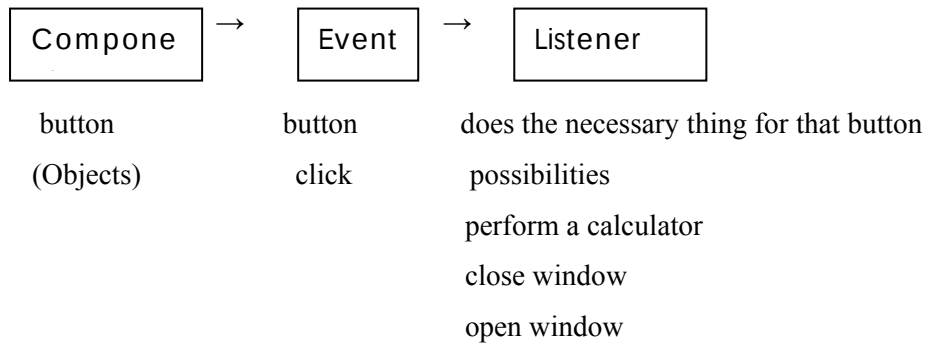
**Event-Driven Programming**



Events

- Pushing a button
- Moving mouse
- Clicking mouse
- Entering a value into a field

↑  
 Event Listeners  
 Awt → Abstract Window Toolkit  
 Swing



In the event based programming the swing system automatically invokes certain methods when an event signals that the method needs to be invoked (called).

### Swing Example

```

import javax.swing.*.*;

public class FirstSwingDemo {

    public static final int width=300;
    public static final int height=200;

    public static void main (String []args)
    {
        JFrame mywindow = new JFrame();
        mywindow.setSize(width, height);

        JLabel myLabel= new JLabel(" Hello World ");
        getContentPane().add(myLabel);
        mywindow.setVisible(true);
        mywindow.addWindowListener(new WindowDestroyer());
    } // end of main
} //end of FirstSwingDemo
  
```

```

import java.awt.*;

public class WindowDestroyer extends WindowAdapter {

    public void windowClosing(Window event e)
    {
        System.exit(0); →exit the program and go back to operating system
    }
}

```

### **A window with colors**

```

Color.BLACK
Color.BLUE
Color.GRAY → import java.awt.*;
Color.GREEN
.
.

import javax.swing.*;
import java.awt.*; //(for colors)

public class SecondWindow extends JFrame
{
    width, height
    public SecondWindow()
    {
        setSize(width, height);
        Container contentPane = getContentPane();
        contentPane.add(label);
        setTitle("SecondWindow");
        contentPane.setBackground(Color.BLUE);
        add.windowListener(new WindowDestroyer);
    }
}

```

```

import java.awt.*; //(for colors)

public class SecondWindowDemo
{
    public static void main (String []args)
    {
        SecondWindow window1 = new SecondWindow();
        SecondWindow window2 = new SecondWindow(Color.ORANGE);

        Window1.setVisible(true);
        Window2.setVisible(true);
    }
}

public class SecondWindow (Color customColor)
{
    width, height
    public SecondWindow()
    {
        setSize(width, height);
        Container contentPane = getContentPane();
        contentPane.add(label);
        setTitle("SecondWindow");
        contentPane.setBackground(Color.BLUE);
        add.windowListener(new WindowDestroyer);
        contentPane.setBackground(Custom Color)
    }
}

```

## USER INTERFACE

**Container classes :** JFrame , JPanel , JApplet,...

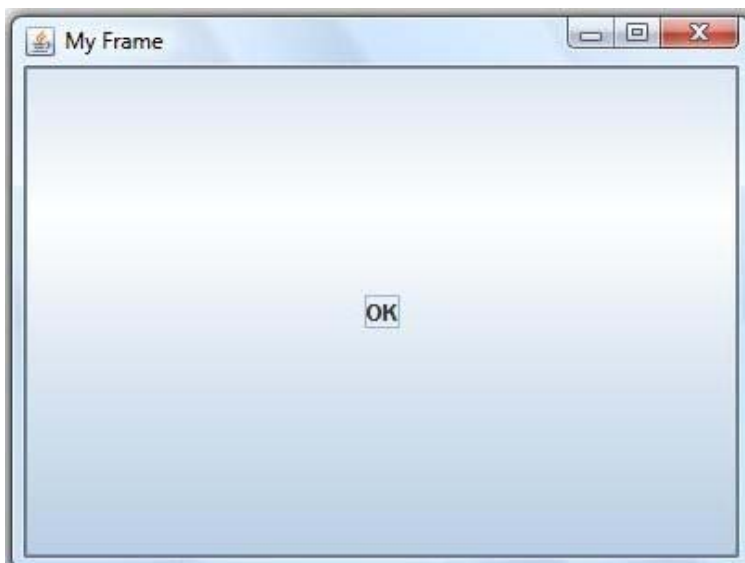
**Component classes :** JButton , JLabel , JTextArea ,...

**Helper classes :** Color , Font , LayoutManager,...

Creating a Frame a simple example:

```
import javax.swing.*;

public class myFrame
{
    public static void main (String[] args)
    {
        JFrame frame = new JFrame ("My Frame");
        frame.setSize(400,300);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JButton myButton = new JButton("OK");
        frame.add(myButton);
        frame.setVisible(true);
    }
}
```



## LAYOUT TYPES:

**Flow Layout** : (default layout type)



### Flow Layout Example:

```
import java.awt.*;
import javax.swing.*;

public class FlowPanel extends JPanel
{
    public FlowPanel()
    {
        setLayout(new FlowLayout());
        setBackground(Color.green);
        JButton b1 = new JButton("Button 1");
        JButton b2 = new JButton("Button 2");
        JButton b3 = new JButton("Button 3");
        JButton b4 = new JButton("Button 4");
        JButton b5 = new JButton("Button 5");
        add(b1);
        add(b2);
        add(b3);
        add(b4);
        add(b5);
    }
}
```

### Grid Layout:

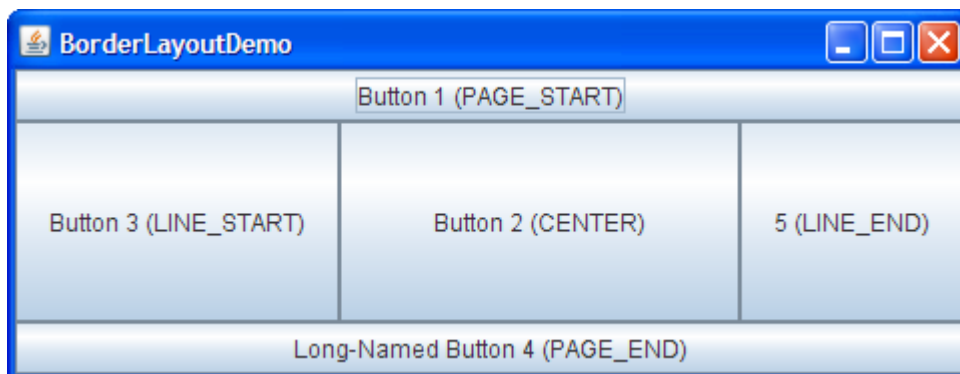


### Grid Layout Example:

```
import java.awt.*;
import javax.swing.*;

public class GridPanel extends JPanel
{
    public GridPanel()
    {
        setLayout(new GridLayout(2,3));
        setBackground(Color.green);
        JButton b1 = new JButton("Button 1");
        JButton b2 = new JButton("Button 2");
        JButton b3 = new JButton("Button 3");
        JButton b4 = new JButton("Button 4");
        JButton b5 = new JButton("Button 5");
        add(b1);
        add(b2);
        add(b3);
        add(b4);
        add(b5);
    }
}
```

### Border Layout:

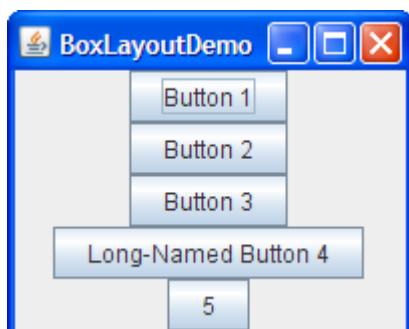


### Border Layout Example:

```
import java.awt.*;
import javax.swing.*;

public class BorderPanel extends JPanel
{
    public BorderPanel()
    {
        setLayout(new BorderLayout());
        setBackground(Color.green);
        JButton b1 = new JButton("Button 1");
        JButton b2 = new JButton("Button 2");
        JButton b3 = new JButton("Button 3");
        JButton b4 = new JButton("Button 4");
        JButton b5 = new JButton("Button 5");
        add(b1, BorderLayout.CENTER);
        add(b2, BorderLayout.NORTH);
        add(b3, BorderLayout.SOUTH);
        add(b4, BorderLayout.EAST);
        add(b5, BorderLayout.WEST);
    }
}
```

**Box Layout:** (opposite of the flow layout)



### Box Layout Example:

```
import java.awt.*;
import javax.swing.*;

public class BoxPanel extends JPanel
{
    public BoxPanel()
    {
        setLayout(new BorderLayout(this, BorderLayout.Y_AXIS));
        setBackground(Color.green);
        JButton b1 = new JButton("Button 1");
        JButton b2 = new JButton("Button 2");
        JButton b3 = new JButton("Button 3");
        JButton b4 = new JButton("Button 4");
        JButton b5 = new JButton("Button 5");
        add(b1);
        add(b2);
        add(b3);
        add(b4);
        add(b5);
    }
}
```

### Layout Demo:

```
import javax.swing.*;

public class LayoutDemo
{
    public static void main(String args[])
    {
        JFrame frame = new JFrame("Layout Manager Demo");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        JTabbedPane tp = new JTabbedPane();
        tp.addTab("Flow" , new FlowPanel());
    }
}
```

```

        tp.addTab("Border" , new BorderPanel());
        tp.addTab("Grid" , new GridPanel());
        tp.addTab("Box" , new BoxPanel());
        frame.getContentPane().add(tp);
        frame.pack();
        frame.setVisible(true);
    }
}

```





**EXAMPLE:** (calculating the circumference and area of a circle)

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.text.DecimalFormat;

public class Circle extends JFrame implements ActionListener {
    // Declare the GUI components and other variables as needed
    private JLabel radiusLabel, resultLabel;
    private JTextField radiusField, resultField;
    private JButton circumferenceButton, areaButton;

    public Circle()
    {
        Container c = getContentPane();

        c.setLayout( new FlowLayout() ); // Set the layout type for
                                         // placement of GUI components

        radiusLabel = new JLabel( "Radius" ); // Create the JLabel object
        c.add( radiusLabel );                // Add to container

        radiusField = new JTextField( 10 ); // Create the JTextField object
        c.add( radiusField );                // Add to container

        resultLabel = new JLabel( "Result" ); // Create the JLabel object
        c.add( resultLabel );                // Add to container
```

```

resultField = new JTextField( 10 ); // Create the JTextField object
resultField.setEditable( false );
c.add( resultField );           // Add to container

    circumferenceButton = new JButton("Circum");// Create the JButton
    circumferenceButton.addActionListener(this);// Add to ActionListener
    c.add(circumferenceButton);           // Add to container

    areaButton = new JButton("Area"); // Create the JButton
    areaButton.addActionListener( this ); // Add to ActionListener
    c.add(areaButton);           // Add to container
}

```

```

public void actionPerformed( ActionEvent e )
{
    // local parameters
    double circumference, area, radius;

    // if event source is "circumferenceButton" object
    if (e.getSource() == circumferenceButton) {
        // Get the radius entered by user
        radius = Double.parseDouble( radiusField.getText() );

        // Calculate the circumference
        circumference = 2 * Math.PI * radius;

        // Round the output to three decimal places
        DecimalFormat fmt = new DecimalFormat("0.###");

        // Show the result within the JTextField object
        resultField.setText( fmt.format(circumference) );
    }

    // if event source is "areaButton" object
    if (e.getSource() == areaButton ) {
        // Get the radius entered by user

```

```

radius = Double.parseDouble( radiusField.getText() );

    // Calculate the area
    area = Math.PI * Math.pow(radius, 2);

    // Round the output to three decimal places
    DecimalFormat fmt = new DecimalFormat("0.###");

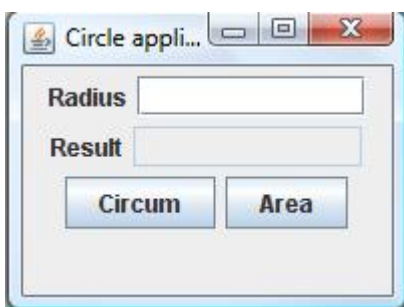
    // Show the result within the JTextField object
    resultField.setText( fmt.format(area) );
}

}

public static void main( String [] args ){
    Circle myApi = new Circle();
    // Close the application when the form's X icon is clicked
    myApi.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    myApi.setLocation(500, 250);
    myApi.setTitle("Circle application"); // Set the form's title
    myApi.setSize(200, 150); // Set the frame's size
    myApi.setVisible(true); // Make the frame visible
}
}

```



**EXAMPLE:** (calculating the hypotenuse of a triangle)

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.text.DecimalFormat;

```

```

public class Triangle extends JFrame implements ActionListener {
    private JLabel side1Label, side2Label, hypotenuseLabel;
    private JTextField side1Field, side2Field, hypotenuseField;
    private JButton calculateButton;

    public Triangle(){

        Container c = getContentPane();
        c.setLayout( new FlowLayout() );

        side1Label = new JLabel( "Side1" );
        c.add( side1Label );

        side1Field = new JTextField( 10 );
        c.add( side1Field );

        side2Label = new JLabel( "Side2" );
        c.add( side2Label );

        side2Field = new JTextField( 10 );
        c.add( side2Field );

        hypotenuseLabel = new JLabel( "Hypotenuse" );
        c.add( hypotenuseLabel );

        hypotenuseField = new JTextField( 10 );
        hypotenuseField.setEditable( false );
        c.add( hypotenuseField );

        calculateButton = new JButton("Calculate");
        calculateButton.addActionListener(this);
        c.add(calculateButton);

    }

    public void actionPerformed( ActionEvent e ){

```

```

    double side1, side2, hypotenuse;

    if (e.getSource() == calculateButton){
        side1 = Double.parseDouble( side1Field.getText() );

        side2 = Double.parseDouble( side2Field.getText() );

        hypotenuse=Math.sqrt(side1*side1+side2*side2);

        DecimalFormat fmt = new DecimalFormat("0.###");

        hypotenuseField.setText( fmt.format(hypotenuse) );

    }
}

```

```

public static void main( String [] args ){
    Triangle myApi = new Triangle();
    myApi.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    myApi.setLocation(500, 250);
    myApi.setTitle("Triangle application");
    myApi.setSize(200, 120);
    myApi.setVisible(true);
}
}

```

