**GUI02**

**Directions**

When creating a GUI program you must first create the frame window. The next step is to add graphical components to the frame. Graphical components consist of things like labels, buttons, text fields, text areas, combo boxes, check boxes, radio buttons, sliders, and panels. In this lesson we learn how to add labels to our frame.

A **JLabel** is a swing component that is a display area for short text or an image. Adding a JLabel to a frame is a four step processes. This same four step processes will be used later to add other graphical components to the frame.

1. create a JLabel variable
2. create a JLabel object and store its reference in the JLabel variable
3. set the attributes for the label (location, size, color, etc.)
4. add the JLabel to the content pane of the frame

import java.util.\*;

import java.awt.\*;

import javax.swing.\*;

import java.awt.event.\*;

public class GUI02 extends JFrame

{

// Step 1 : create JLabel variable

private JLabel label;

// constructor

public GUI02()

{

// Step 2 : create JLabel object and store its reference in label

label = new JLabel("This is a test");

// Step 3 : set label attributes

label.setLocation(10, 10);

label.setSize(200, 20);

label.setForeground(Color.BLUE);

label.setFont(new Font("Arial", Font.PLAIN, 24));

// Step 4: add label to content pane of frame

getContentPane().add(label);

// set frame attributes

setLayout(null);

setSize(500, 500);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

getContentPane().setBackground(Color.white);

setVisible(true);

}

// main method

public static void main(String[] args)

{

GUI02 app = new GUI02(); // run program

}

}

Before we examine the JLabel code it must be pointed out that frame attributes have not changed. You will notice, however, that the code is located at the bottom of the constructor. In our GUI program ordering you will always put the code for the graphical components above the code that sets the attributes for the frame.

Step 1 : create a JLabel variable

private JLabel label;

The JLabel variable label is used to store a reference to the JLabel object so that the object can be accessed in our GUI program. All graphical component variables should be declared as **instance variables**.

Step 2 : create a JLabel object and store its reference in the JLabel variable

label = new JLabel("This is a test");

To create a JLabel object you use the keyword new followed by JLabel. The parentheses contain the text that will be displayed on the screen.

Step 3 : set the attributes for the label (location, size, color, etc.)

label.setLocation(10, 10);

label.setSize(200, 20);

label.setForeground(Color.BLUE);

label.setFont(new Font("Arial", Font.PLAIN, 24));

**setLocation** - determines the location with the frame where the label will be displayed.  
**setSize** - sets the size of the label. The size is dependent upon the size of the text in the label and the font size being used.  
**setForeground** - sets the color of the label's text.  
**setFont** - sets the font that will be used to display the text. When setting the font you must create a new font object and provide the font name, font style, and font size.

There are three font style attributes.

|  |  |
| --- | --- |
| **Font Style** | **Description** |
| **PLAIN** | The plain style constant (Font.PLAIN) |
| **BOLD** | The bold style constant. (Font.BOLD) |
| **ITALIC** | The italicized style constant (Font.ITALIC) |

Here are the established baseline fonts: SansSerif, Serif, Dialog, DialogInput, Monospaced. These are not system dependent.

**Example**

Font myFont = new Font("Serif", Font.PLAIN, 18);

You can also use the system specific font names like Times New Roman and Arial which are defined in the Windows operating system.

**Example**

Font myFont = new Font("Times New Roman", Font.BOLD, 24);

Step 4 : add the JLabel to the content pane of the frame

getContentPane().add(label);

To add a graphical component to the content pane of the frame you use the add method.

**Exercise**

1. Right-click on the GUI project name and select Add -> New File.  
2. Type GUI02 and click Finish.  
3. Copy the code above into the GUI02 file.

**Modifications**

4. Change the text of the label to "Have a nice day!".  
5. Center the label in the middle of the frame window.  
6. Change the font attributes for the label. Adjust the size of the label as needed.  
7. Change the color of the label.

Here is the list of the preset colors.

|  |  |
| --- | --- |
| **Color** | **Command** |
| black | Color.black |
| blue | Color.blue |
| cyan | Color.cyan |
| gray | Color.gray |
| green | Color.green |
| magenta | Color.magenta |
| orange | Color.orange |
| white | Color.white |
| yellow | Color.yellow |

**Source File**

GUI02.java

**Sample Run**

