

Layouts – FlowLayout



Essentially a Flow Layout merely 'flows' the components from left to right rather like a textual document, the flow depends on the number of lines (and their width) available for the Flow Layout to display on, one line and all components will appear on that line only, several lines and the width will start dictating the layout.

The Developer will need to import the following ...

```
import java.awt.FlowLayout;
```

As a default, FlowLayout will 'center' components in the screen and place a gap of 3 pixels between each component.

A panel must set its layout to the Layout method required ...



Layouts – FlowLayout

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

```
public class flowlayout1 {
```

```
    static JLabel msg = new JLabel("This adds a message");  
    static JLabel msg2 = new JLabel("Another message");  
    static JFrame newframe = new JFrame("FlowLayout");  
    static JPanel pane1 = new JPanel();  
    static FlowLayout flow1 = new FlowLayout();
```

```
public static void main(String[] arguments)
```

```
{
```

```
    newframe.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
    pane1.setLayout(flow1);
```

```
    pane1.add(msg);
```

```
    pane1.add(msg2);
```

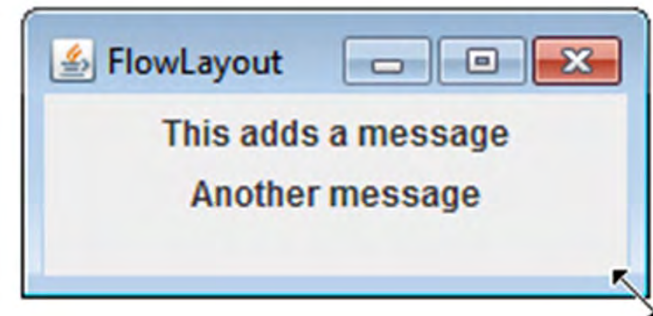
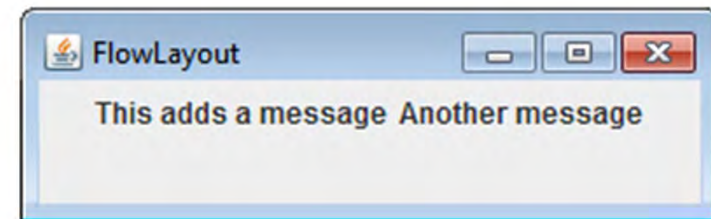
```
    newframe.getContentPane().add(pane1);
```

```
    newframe.pack();
```

```
    newframe.setVisible(true);
```

```
}
```

```
}
```



Layouts – FlowLayout



FlowLayout allows the Developer to control the alignment of the contents as well as the gap both vertically and horizontally between components in the layout.

Here the Developer is specifying that components are right justified ...

```
FlowLayout flow1 = new FlowLayout(FlowLayout.RIGHT);
```

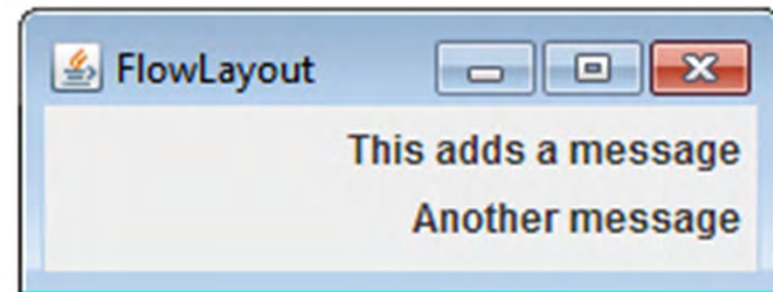
Other options for alignment are ...

FlowLayout.LEFT

FlowLayout.CENTER

FlowLayout.TRAILING

FlowLayout.LEADING

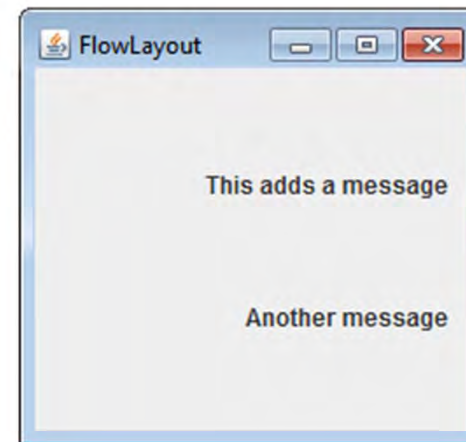
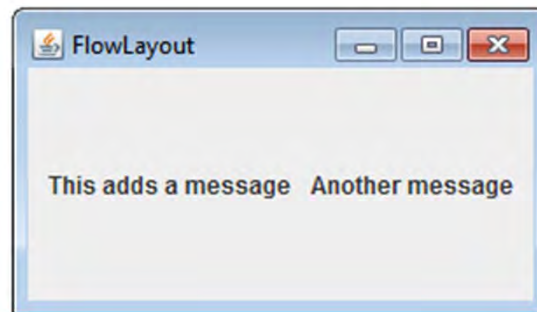


Layouts – FlowLayout



The other two arguments change the pixel distance between the components, the second argument is the horizontal gap, the third the vertical gap.

```
FlowLayout flow1 = new FlowLayout(FlowLayout.RIGHT, 10, 50);
```



Layouts – Some FlowLayout Methods



The following methods are available for the Developer to use during processing ...

`getAlignment()` – Retrieves the Alignment value

```
System.out.println(flow1.getAlignment()); // flow1 is the name of the Layout
```

Returns a numeric value for the setting ...

```
FlowLayout.LEFT = 0  
FlowLayout.CENTER = 1  
FlowLayout.RIGHT = 2  
FlowLayout.LEADING = 3  
FlowLayout.TRAILING = 4
```



Layouts – Some FlowLayout Methods

`getHgap()` – Retrieves the horizontal gap

`getVgap()` – Retrieves the vertical gap

`setAlignment()` – Sets the Alignment style

`setHgap()` – Sets the horizontal gap

`setVgap()` – Sets the vertical gap

`toString()` – Retrieves the Layout's current settings ...

```
System.out.println(flow1.toString());
```

```
java.awt.FlowLayout[hgap=10,vgap=50,align=left]
```