

14.4.3.1 Die Klasse `FlowLayout`

Um die Komponenten in einem Container fließend anzuordnen, verwendet man ein Objekt der Klasse `FlowLayout` als Layout-Manager. „Fließend“ bedeutet hier, dass die Komponenten zeilenweise von links nach rechts in den Container eingefügt werden. Das heißt, die Komponenten werden so lange in der Reihenfolge ihres Einfügens von links nach rechts nebeneinander platziert, bis kein Platz mehr für die nächste Komponente verfügbar ist und mit einer neuen Zeile begonnen werden muss, die dann genauso gefüllt wird. Die Ausrichtung der Komponenten innerhalb der Zeile erfolgt dabei standardmäßig zentriert. Zwischen den Komponenten findet sich horizontal und vertikal jeweils ein Abstand von 5 Pixel. Die Größe der Komponenten wird nicht verändert.

In der Klasse `FlowLayout` finden wir die folgenden Konstruktoren:

- **public** `FlowLayout()`
erzeugt ein `FlowLayout`-Objekt mit den Standardeinstellungen (zentrierte Ausrichtung der Zeilen, 5-Pixel-Abstände).
- **public** `FlowLayout(int align)`
erzeugt ein `FlowLayout`-Objekt mit einer Ausrichtung gemäß `align` und der Standardeinstellung für die Abstände.
- **public** `FlowLayout(int align, int h, int v)`
erzeugt ein `FlowLayout`-Objekt mit einer Ausrichtung gemäß `align` und horizontalen bzw. vertikalen Abständen von `h` bzw. `v` Pixel.

Für die Wahl der Ausrichtung stehen die vordefinierten Konstanten `LEFT` (für linksbündige Ausrichtung), `RIGHT` (für rechtsbündige Ausrichtung) und `CENTER` (für zentrierte Ausrichtung) als finale Klassenvariablen der Klasse `FlowLayout` zur Verfügung.

In unserem Beispielprogramm

```

1  import java.awt.*;
2  import javax.swing.*;
3  /** Erzeuge ein Swing-Fenster mit Flowlayout */
4  public class FrameMitFlowLayout extends JFrame {
5      Container c;           // Container dieses Frames
6      // Feld fuer Labels, die im Frame erscheinen sollen
7      FarbigesLabel fl[] = new FarbigesLabel[4];
8
9      public FrameMitFlowLayout() { // Konstruktor
10         c = getContentPane();    // Container bestimmen
11         c.setLayout(new FlowLayout()); // Layout setzen
12
13         // Erzeuge die Labelobjekte mit Uebergabe der Labeltexte
14         for (int i = 0; i < 4; i++) {
15             int rgbFg = 255 - i*80; // Farbwert fuer Vordergrund
16             int rgbBg = i*80;      // Farbwert fuer Hintergrund
17             fl[i] = new FarbigesLabel("Nummer " + (i+1),
18                                     new Color(rgbFg, rgbFg, rgbFg),
19                                     new Color(rgbBg, rgbBg, rgbBg));
20             fl[i].setFont(new Font("Serif", Font.ITALIC, 28));

```