

1. 概述

官方JavaDocsApi: [java.awt.GridLayout](#)

GridLayout，网格布局管理器。它以矩形网格形式对容器的组件进行布置，把容器按行列分成**大小相等**的矩形网格，一个网格中放置一个组件，**组件宽高自动撑满网格**。

以行数和总数优先: 通过构造方法或 `setRows` 和 `setColumns` 方法将行数和列数都设置为非零值时，指定的列数将被忽略。列数通过指定的行数和布局中的组件总数来确定。因此，例如，如果指定了三行和两列，在布局中添加了九个组件，则它们将显示为三行三列。仅当将行数设置为零时，指定列数才对布局有效。

构造方法:

```
1 // 默认构造，每个组件占据一行一列
2 GridLayout()
3 // 指定 行数 和 列数 的网格布局
4 GridLayout(int rows,int cols)
5 // 指定 行数 和 列数 的网格布局，并指定 水平 和 竖直 网格间隙
6 GridLayout(int rows,int cols,int hgap,int vgap)
```

•

2. 代码演示

```
1 package com.xiets.swing;
2 import javax.swing.*;
3 import java.awt.*;
4 public class Main{
5     public static void main(String[] args){
6         JFrame jf =newJFrame("测试窗口");
7         jf.setSize(200,250);
8         jf.setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
9         jf.setLocationRelativeTo(null);
10        // 创建 3 行 3 列 的网格布局
11        GridLayout layout =newGridLayout(3,3);
12        // 设置 水平 和 竖直 间隙
13        // layout.setHgap(10);
14        // layout.setVgap(10);
15        JPanel panel =newJPanel(layout);
```

```
16 JButton btn01 =newJButton("按钮01");
17 JButton btn02 =newJButton("按钮02");
18 JButton btn03 =newJButton("按钮03");
19 JButton btn04 =newJButton("按钮04");
20 JButton btn05 =newJButton("按钮05");
21 JButton btn06 =newJButton("按钮06");
22 JButton btn07 =newJButton("按钮07");
23 JButton btn08 =newJButton("按钮08");
24 panel.add(btn01);
25 panel.add(btn02);
26 panel.add(btn03);
27 panel.add(btn04);
28 panel.add(btn05);
29 panel.add(btn06);
30 panel.add(btn07);
31 panel.add(btn08);
32 jf.setContentPane(panel);
33 jf.setVisible(true);
34 }
35 }
```

结果展示:



result.png