

本文链接: <http://blog.csdn.net/xietansheng/article/details/72814548>

[Java Swing 图形界面开发 \(目录\)](#)

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。

GridLayout构造方法:

// 默认构造, 每个组件占据一行一列 `GridLayout()` // 指定 行数 和 列数 的网格布局

`GridLayout(int rows,int cols)` // 指定 行数 和 列数 的网格布局, 并指定 水平 和 竖直 网格

间隙 `GridLayout(int rows,int cols,int hgap,int vgap)`

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8

2. 代码演示

```
package com.xiets.swing;import javax.swing.*;import
java.awt.*;publicclassMain{publicstaticvoidmain(String[] args){    JFrame jf
=newJFrame("测试窗口");    jf.setSize(200,250);
jf.setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
jf.setLocationRelativeTo(null);// 创建 3 行 3 列 的网格布局    GridLayout layout
=newGridLayout(3,3);// 设置 水平 和 竖直 间隙// layout.setHgap(10);//
```

```

layout.setVgap(10);      JPanel panel =newJPanel(layout);      JButton btn01
=newJButton("按钮01");    JButton btn02 =newJButton("按钮02");    JButton
btn03 =newJButton("按钮03");    JButton btn04 =newJButton("按钮04");
JButton btn05 =newJButton("按钮05");    JButton btn06 =newJButton("按钮06");
JButton btn07 =newJButton("按钮07");    JButton btn08 =newJButton("按钮08");
panel.add(btn01);        panel.add(btn02);        panel.add(btn03);
panel.add(btn04);        panel.add(btn05);        panel.add(btn06);
panel.add(btn07);        panel.add(btn08);        jf.setContentPane(panel);
jf.setVisible(true);}}

```

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24
- 25
- 26

- 27
- 28
- 29
- 30
- 31
- 32
- 33
- 34
- 35
- 36
- 37
- 38
- 39
- 40
- 41
- 42
- 43
- 44
- 45

结果展示:



result.png