

Java图形用户界面设计 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/145/2021\\_2022\\_Java\\_E5\\_9B\\_BE\\_E5\\_BD\\_A2\\_c104\\_145007.htm](https://www.100test.com/kao_ti2020/145/2021_2022_Java_E5_9B_BE_E5_BD_A2_c104_145007.htm) 作为一个 Java 程序员，从论坛上感受到使用 Java 开发程序的人越来越多，心中不免欣慰。但是，同样是从论坛中，看到多数人提到 Java 就以为是网络开发不是这样的，Java 也可以开发应用程序，而且可以开发出漂亮的图形用户界面的应用程序，也就是

Windows/XWindow 应用程序。因此，我写下这篇文章，希望能带你进入Java 图形用户界面设计之门。 一. AWT 和 SWING  
AWT 和 SWING 是 Java 设计 GUI 用户界面的基础。与 AWT 的重量级组件不同，Swing 中大部分是轻量级组件。正是这个原因，Swing 几乎无所不能，不但有各式各样先进的组件，而且更为美观易用。所以一开始使用 AWT 的程序员很快就转向使用 Swing 了。那为什么 AWT 组件没有消亡呢？因为 Swing 是架构在 AWT 之上的，没有 AWT 就没有 Swing。所以程序员可以根据自己的习惯选择使用 AWT 或者是 Swing。但是，最好不要二者混用除开显示风格不同不说，还很可能造成层次 (Z-Order) 错乱，比如下例：/\*\* TestPanels.java\*

```
@author Fancy*/ import javax.swing.*.import java.awt.*.public class TestPanels extends JFrame { public TestPanels() { setDefaultCloseOperation(EXIT_ON_CLOSE). JPanel panel = new JPanel(). for (int i = 0. i panel.add(new JButton("Button 00" i)). } JTextArea textArea = new JTextArea(5, 15). textArea.setLineWrap(true). JScrollPane scrollPane = new JScrollPane(textArea). getContentPane().add(panel,
```

```
BorderLayout.NORTH). getContentPane().add(scrollPane,  
BorderLayout.CENTER). pack(). } public static void main(String[]  
args) { TestPanels tp = new TestPanels(). tp.show(). } }
```

运行这个程序，并用鼠标拖动那个名为“cover”的子窗口，我们会发现一个非常有趣的现象，如图：显然 cover 子窗口是在 controls 子窗口之上的，但是它只遮盖住了 Swing Button，没有遮盖住 AWT Button。再看一会儿，你是不是有这样一种感觉：Swing Button 是“画”上去的，而 AWT Button 则是“贴”上去的。这就是二者混用造成层次错乱的一个例子。Swing 组件有美观、易用、组件量大等特点，也有缺点使用 Swing 组件的程序通常会比使用 AWT 组件的程序运行更慢。但是大家都还是更喜欢用 Swing 组件，原因何在？因为随着计算机硬件的升级，一点点速度已经不是问题。相反的，用户更需要美观的用户界面，开发人员则更需要易用的开发组件。好，我这就来教你使用 Swing 组件开发图形用户界面的 Java 应用程序。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)