1. **组件 JComponent**

## 1.1 atomic widgets(or components)

1. **Atomic widgets for user input with the mouse:**
   * **JButton;JCheckBox;JComboBox; JRadioButton;JList;JSlider;JTree**
2. **不可编辑信息的显示：向用户显示不可编辑信息的组件:**

* **JLable;JProgressBar;JProgressMonitor;**
* **为组件设置对用户有帮助的提示信息：每个组件都有方法setToolTipText(“String”)**

1. **可编辑信息的显示：向用户显示能被编辑的格式化信息的组件：**

**如JColorChooser, JFileChoose, JFileChooser, JTable, JTextArea，JTextField。**

## intermediate containers

中级容器用于容纳界面元素，以便在布局管理器的设置下可容纳更多的组件，实现容器的嵌套。

* + **JPanel;JScrollPane; JSplitPane; JTabbedPane; JToolBar; JLayeredPane; JDesktopPane; JInternalFrame; JRootPane**

## 1.3 top-level containers

**A top-level container holds intermediate containers and atomic widgets.**

* + **顶级容器不能被其它的容器包含。**
  + **JFrame for general use**
    - **在构建桌面Java应用程序时，通常创建一个JFrame来容纳所需的组件，以提供期望的GUI外观和功能。**
  + **JDialog for creating pop-up dialog boxes**
  + **JApplet for writing Swing based applets.**
  + **JWindow**

**二、布局管理器**

**布局管理器LayoutManager是一个接口，有三种实现该接口的三种布局类：**

* + **BorderLayout**
  + **FlowLayout**
  + **GridLayout**

**1.2和1.3中的每个容器都有指定一个具体的布局管理器的方法：**

* + - **void** [**setLayout**](../../j2sdk-1_4_1-doc/docs/api/java/awt/LayoutManager.html)**(**[**LayoutManager**](java-source/BorderLayoutDemo.java)**mgr)**

**三、界面响应（Event handling）**

* **We will learn how to recognize events(how a GUI behaves)---i.e. users’ interaction with the GUI/view-----and how to control how the application responds to them.**

**编程人员做两件事：**

* 1. **创建Listener(监听器)类**
     + **对于某种类型的事件XXXEvent, 要想接收并处理这类事件，必须定义实现监听器接口XXXListener监听器类；**
  2. **调用组件的addXXXListener(…)方法注册Listener对象，就可以监听该组件触发的XXXEvent。**