

JC2002 Java 编程 - 实践 7（第 9 天）

本实践的目的是熟悉事件驱动编程和在 Swing 图形用户界面中实现功能。我们将以上次实践中实现的计算器虚拟图形用户界面为起点。

1. 在 **GraphicalCalculator** 类中，首先添加初始化为零的整数变量 **status** 和初始化为空的字符串变量 **storedText**。然后，实现动作监听器，为按钮添加功能，如下所示：

- 对于 "清除"按钮：使用 **JTextField** 类的 **setText()** 方法将文本字段的内容设置为空字符串，将 **status** 设置为零，将 **storedText** 设置为 "0"，然后返回。
- 对于 "添加"按钮：使用 **JTextField** 类的 **getText()** 方法读取文本字段中的字符串，并将其存储到 **storedValue** 中。将文本字段的内容设置为空字符串，并将**状态**设置为 1。
- 对于 "Subtract"按钮：从文本字段中读取字符串并将其转换为浮点数，然后将新解析的值存储到 **storedValue** 中。将文本字段的内容设置为空字符串，并将**状态**设置为 2。
- 对于 "乘法"按钮：从文本字段中读取字符串并将其转换为浮点数，然后将新解析的值存储到 **storedValue** 中。将文本字段的内容设置为空字符串，并将**状态**设置为 3。
- 对于 "除法"按钮：从文本字段中读取字符串并将其转换为浮点数，然后将新解析的值存储到 **storedValue** 中。将文本字段的内容设置为空字符串，并将**状态**设置为 4。
- 对于 "结果"按钮：检查状态。如果**状态**为零，则直接返回。否则，使用方法 **Float.parseFloat()** 将 **storedText** 的值转换为浮点数 **firstValue**，然后从文本字段读取字符串，并将其转换为浮点数 **secondValue**。然后，如果**状态**为 1，计算 **firstValue + secondValue**，并将结果存储在 **float resultValue** 中。如果**状态**为 2，则计算 **firstValue - secondValue** 并将结果存储在 **resultValue** 中。如果**状态**为 3，则计算 **firstValue * secondValue** 并将结果存储在 **resultValue** 中。如果**状态**为 4，则计算 **firstValue / secondValue** 并将结果存储在 **resultValue** 中。最后，使用 **String.valueOf()** 方法将 **resultValue** 转换为字符串，在文本字段中显示结果值，并将**状态**设置为零。

2. 您可能会注意到，如果写入文本字段的输入字符串不能有效地转换为浮点数，**Float.parseFloat()** 方法就会抛出 **NumberFormatException** 异常。实施一个 **try...catch** 异常处理程序来处理非法输入（可以实施一个特定方法，在该方法中解析文本字段值并处理异常），然后使用 **JOptionPane.showMessageDialog()** 方法通知用户非法输入。如果是非法输入，则将 **status** 设置为 0，**storedText** 设置为空，文本字段内容设置为空字符串。
3. 思考改进计算器的各种方法。例如，如何向用户显示状态？