# LAB 6. LẬP TRÌNH SỰ KIỆN (BUỔI 1)

## Mục tiêu

* Biết cách tạo sự kiện cho một components trên Netbeans và bằng code.
* Biết cách sử dụng và tính năng của một số sự kiện cơ bản hay dùng cho lập trình giao diện trên Java.

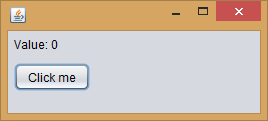
## Yêu cầu

## Bài tập mẫu

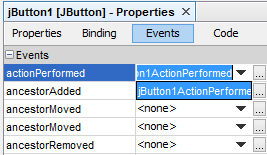
***Bài 1.*** Bắt sự kiện Click cho Button

***Hướng dẫn:***

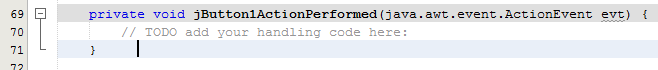
*Bước 1.* Thiết kế giao diện gồm 2 thành phần là JButton, JLabel như hình



*Bước 2.* Bắt sự kiện Click cho Button bằng cách double click vào đối tượng JButton trên khung thiết kế hoặc trên cửa sổ ***Properties***, chuyển sang tab ***Events,*** tại sự kiện ***actionPerformed,*** ta click vào mũi tên trỏ xuống và click double vào sự kiện mà Netbeans gợi ý.



Netbeans sẽ tự động sinh ra phương thức xử lý sự kiện cho chúng tại cửa sổ soạn thảo code. Lúc này, ta có thể viết thêm code xử lý khi có sự kiện click vào JButton này.



*Bước 3*. Viết code xử lý khi xảy ra sự kiện Click vào Button. Trong trường hợp này, ta sẽ có 1 biến đếm kiểu int dùng để lưu trữ số lần Click vào Button, mặc định ban đầu có giá trị bằng 0. Cứ sau mỗi lần Click vào Button, ta sẽ tăng giá trị của biến đếm lên 1 đơn vị và hiển thị giá trị đó lên JLabel.

Ta code như sau:

int iCount = 0; // Khai báo 1 biến đếm, mặc định bằng 0

private void jButton1ActionPerformed(

java.awt.event.ActionEvent evt) {

// Tăng biến đếm lên 1 đơn vị

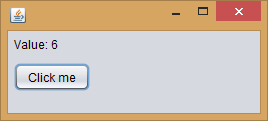
iCount++;

// Hiển thị giá trị của biến đếm lên jLabel

jLabel1.setText("Value: "+ iCount);

}

Kết quả khi chạy chương trình:



* Đối với trường hợp muốn bắt sự kiện cho Button trong code, ta làm như sau:

jButton1.addActionListener(new

java.awt.event.ActionListener() {

public void actionPerformed(

java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

private void jButton1ActionPerformed(

java.awt.event.ActionEvent evt) {

// TODO: Code sự kiện ở đây

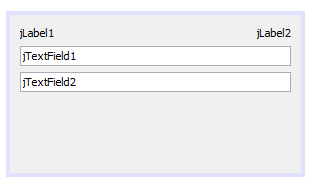
}

***Bài 2.*** Bắt sự kiện của JTextField:

* Sự kiện caretUpdate: Xảy ra khi nội dung của JTextField thay đổi.
* Sự kiện focusLost: Sự kiện khi JTextField mất focus
* Sự kiện focusGained: Sự kiện khi JTextField được focus.

***Hướng dẫn:***

*Bước 1.* Thiết kế giao diện như hình:

****

*Bước 2*. Bắt sự kiện caretUpdate, focusLost và focusGained cho jTextField1. Ta code như sau:

private void jTextField1CaretUpdate(

javax.swing.event.CaretEvent evt){

// Lấy nội dung text của jTextField1

String text = jTextField1.getText();

// Gán nội dung text vừa lấy được vào jLabel1

jLabel1.setText(text);

}

private void jTextField1FocusGained(

java.awt.event.FocusEvent evt) {

jLabel2.setText("GAINED");

}

private void jTextField1FocusLost(

java.awt.event.FocusEvent evt) {

jLabel2.setText("LOST");

}

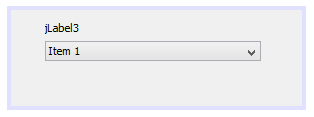
Chạy chương trình và thực hiện nhập nội dung vào jTextField1 và thực hiện nhấn con trỏ chuột vào jTextField1 và jTextField2 và cảm nhận.

***Bài 3.*** Bắt sự kiện của JComboBox:

* Sự kiện ItemStateChanged: Xảy ra khi lựa chọn item của JComboBox thay đổi.

***Hướng dẫn:***

*Bước 1.* Thiết kế giao diện như hình:



*Bước 2.* Vào cửa sổ Properties, tab Events của đối tượng JComboBox ta chọn sự kiện ItemStateChanged. Ta code như sau cho phương thức sự kiện:

private void jComboBox1ItemStateChanged(

java.awt.event.ItemEvent evt) {

// Lấy text của phần tử đang được chọn của JComboBox

String value =

jComboBox1.getSelectedItem().toString();

// Set giá trị vừa lấy được và label

jLabel3.setText(value);

}

Chạy chương trình và kiểm tra kết quả.

## Bài tập

***Bài 1.*** Áp dụng nội dung đã thực hành xây dựng chương trình nhập vào 2 số, tính tổng, hiệu, tích, thương của 2 số đó.

***Bài 2.*** Viết chương trình nhập vào 3 số là 3 cạnh của 1 tam giác. Trên giao diện có 3 JRadioButton: Kiểm tra tam giác cân, kiểm tra tam giác đều, kiểm tra tam giác vuông. Hiển thị kết quả ra JLabel khi một JRadioButton được lựa chọn.

*Hướng dẫn:* Thiết kế giao diện như sau

