# LAB5. LẬP TRÌNH GIAO DIỆN (BUỔI 1)

## Mục tiêu

Tìm hiểu về các thuộc tính và cách sử dụng cơ bản của các components JLabel, JTextField, JButton, JRadioButton, JTable, JComboBox. Từ đó biết cách xây dựng giao diện bằng code trực tiếp không qua kéo thả.

## Yêu cầu

* Sinh viên làm theo các phần theo nội dung hướng dẫn
* Làm bài tập về nhà phần bài tập

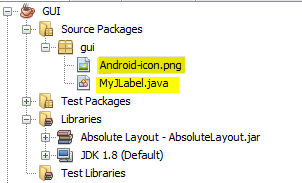
## Hướng dẫn

***Bài 1.*** Dùng code để thêm các Component sau vào giao diện: JLabel, JTextField, JButton, JRadioButton, JTable, JComboBox.

***Hướng dẫn:***

* 1. ***JLabel***

Tạo project mới, tiếp theo ta tạo 1 class mới có tên là MyJLabel. Tiếp theo đó, kéo thả 1 hình ảnh vào package chứa class đó. Kết quả ta có hình như sau:



Trong file MyJLabel.java, ta viết code như sau:

package gui;

import java.awt.GridLayout;

import javax.swing.Icon;

import javax.swing.ImageIcon;

import javax.swing.JFrame;

import javax.swing.JLabel;

public class MyJLabel extends JFrame {

public MyJLabel() {

// Tạo layout cho Frame: thuộc loại GridLayout gồm 1 hàng, 3 cột, cách nhau theo chiều ngang 5px, dọc 5px

setLayout(new GridLayout(1, 3, 5, 5));

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

// Tạo hình ảnh icon

Icon icon = new ImageIcon(getClass().getResource("Android-icon.png"));

// Tạo label chỉ gồm Text

JLabel lb1 = new JLabel("Label 1");

// Tạo label chỉ có hình ảnh icon

JLabel lb2 = new JLabel(icon);

// Tạo label có cả text và icon

JLabel lb3 = new JLabel("Icon and text", icon, JLabel.CENTER);

lb3.setVerticalTextPosition(JLabel.BOTTOM);

lb3.setHorizontalTextPosition(JLabel.CENTER);

// Thêm 3 label vào frame

add(lb1);

add(lb2);

add(lb3);

// Hiển thị JLabel lên

pack();

setLocationRelativeTo(null);

setVisible(true);

}

public static void main(String[] args) {

new MyJLabel();

}

}

* 1. ***JButton***

Tương tự với phần Label, ta tạo class tên là ***MyJButton*** và viết code như sau:

package gui;

import java.awt.GridLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

public class MyJButton extends JFrame {

private JLabel lb;

public MyJButton() {

// Tạo frame

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setLayout(new GridLayout(2, 1, 5, 5));

setSize(500, 200);

// Tạo 1 label

lb = new JLabel("My JLabel");

lb.setHorizontalAlignment(JLabel.CENTER);

add(lb);

// create JButton with text "Click Button"

JButton btn = new JButton("Click Button");

// Thêm sự kiện cho button. Ta sẽ tìm hiểu phần này ở chương tiếp

btn.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent arg0) {

changeTextJLabel();

}

});

// Thêm button vào Frame

add(btn);

// Hiển thị Frame

pack();

setLocationRelativeTo(null);

setVisible(true);

}

// change text of lb

private void changeTextJLabel() {

lb.setText("You are clicked JButton");

}

public static void main(String[] args) {

new MyJButton();

}

}

* 1. ***JTextField***

package gui;

import java.awt.GridLayout;

import javax.swing.JLabel;

import javax.swing.JFrame;

import javax.swing.JTextField;

public class MyJTextField extends JFrame {

JLabel lb;

public MyJTextField() {

// create JFrame

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setLayout(new GridLayout(2, 1, 5, 5));

// Tạo và thêm JTextField vào frame

JTextField tf = new JTextField(20);

add(tf);

// Thiết lập text cho TextField

tf.setText("This is text of TextField");

// Khởi tạo 1 JLabel và thêm vào Frame

lb = new JLabel();

add(lb);

// Lấy nội dung text của TextField

String text = tf.getText();

// Gán nội dung text vừa lấy được

// từ TextField vào JLabel

lb.setText(text);

// Hiển thị Frame

pack();

setLocationRelativeTo(null);

setVisible(true);

}

public static void main(String[] args) {

new MyJTextField();

}

}

* 1. ***JRadioButton***

package gui;

import java.awt.GridLayout;

import javax.swing.ButtonGroup;

import javax.swing.JRadioButton;

import javax.swing.JFrame;

public class MyJRadioButton extends JFrame {

private javax.swing.ButtonGroup buttonGroup;

public MyJRadioButton() {

// create JFrame

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setLayout(new GridLayout(2, 1, 5, 5));

// Khởi tạo 1 JRadioButton có text

JRadioButton rdNam = new JRadioButton("Nam");

// Khởi tạo 1 JRadioButton không có text

JRadioButton rdNu = new JRadioButton();

// Set text qua phương thức setText()

rdNu.setText("Nữ");

// Thiết lập JRadioButton được chọn

rdNu.setSelected(true);

add(rdNam);

add(rdNu);

// Khởi tạo ButtonGroup và thêm

// các thành phần thuộc nhóm đó

buttonGroup = new ButtonGroup();

buttonGroup.add(rdNu);

buttonGroup.add(rdNam);

// Hiển thị Frame

pack();

setLocationRelativeTo(null);

setVisible(true);

}

public static void main(String[] args) {

new MyJRadioButton();

}

}

* 1. ***JComboBox***

package gui;

import java.awt.GridLayout;

import javax.swing.ButtonGroup;

import javax.swing.DefaultComboBoxModel;

import javax.swing.JComboBox;

import javax.swing.JRadioButton;

import javax.swing.JFrame;

public class MyJComboBox extends JFrame {

public MyJComboBox() {

// create JFrame

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setLayout(new GridLayout(3, 1, 15, 15));

JComboBox jcbo1 = new JComboBox();

JComboBox jcbo2 = new JComboBox();

JComboBox jcbo3 = new JComboBox();

add(jcbo1);

add(jcbo2);

add(jcbo3);

// Cách 1. Sử dụng thông qua mảng

// khởi tạo Model cho JComboBox

String[] items = {"C#", ".NET", "Java", "PHP"};

DefaultComboBoxModel model1 =

new DefaultComboBoxModel(items);

jcbo1.setModel(model1);

// Cách 2. Sử dụng thông qua Model,

// thêm các phần tử đơn lẻ

DefaultComboBoxModel model2 =

new DefaultComboBoxModel();

model2.addElement("C#");

model2.addElement(".NET");

model2.addElement("Java");

model2.addElement("PHP");

model2.addElement("Python");

jcbo2.setModel(model2);

// Cách 3. Sử dụng trực tiếp trên JComboBox

jcbo3.addItem("C#");

jcbo3.addItem(".NET");

jcbo3.addItem("Java");

jcbo3.addItem("PHP");

// Hiển thị Frame

pack();

setLocationRelativeTo(null);

setVisible(true);

}

public static void main(String[] args) {

new MyJComboBox();

}

}

* 1. ***JTable***

package gui;

import java.awt.GridLayout;

import javax.swing.ButtonGroup;

import javax.swing.DefaultComboBoxModel;

import javax.swing.JComboBox;

import javax.swing.JRadioButton;

import javax.swing.JFrame;

import javax.swing.JScrollPane;

import javax.swing.JTable;

import javax.swing.table.DefaultTableModel;

import javax.swing.table.TableColumn;

public class MyJTable extends JFrame {

public MyJTable() {

// create JFrame

setLayout(new GridLayout(1, 1, 15, 15));

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setSize(500, 500);

// Khởi tạo 1 JTable

JTable tbl = new JTable();

// Khởi tạo 1 mảng String chứa các tiêu đề của cột

String[] headers = new String[]

{"Title 1",

"Title 2",

"Title 3",

"Title 4"};

// Tạo 1 model cho table có tiêu đề cột và 0 hàng

DefaultTableModel model =

new DefaultTableModel(headers, 0);

//Thêm dữ liệu cho Table.

model.addRow(new String[]{"11", "12", "13", "14"});

model.addRow(new String[]{"21", "22", "23", "24"});

model.addColumn("Title 5",

new String[]{"15", "25"});

tbl.setModel(model);

// Tạo thanh cuộn cho JTable khi kích thước

// của table vượt quá kích thước của khung chứa

JScrollPane sp = new JScrollPane(tbl);

add(sp);

// Hiển thị Frame

pack();

setLocationRelativeTo(null);

setVisible(true);

}

public static void main(String[] args) {

new MyJTable();

}

}

## Bài tập

***Bài 1.*** Tương tự với các Control đã làm qua phần hướng dẫn, tìm hiểu và thực hành với JList, JTree, JProgressBar.

***Bài 2***. Kết hợp những phần đã thực hành ở trên để xây dựng giao diện form nhập thông tin đã thực hiện ở LAB4.