# Swing常用组件（二）

## 实验目的

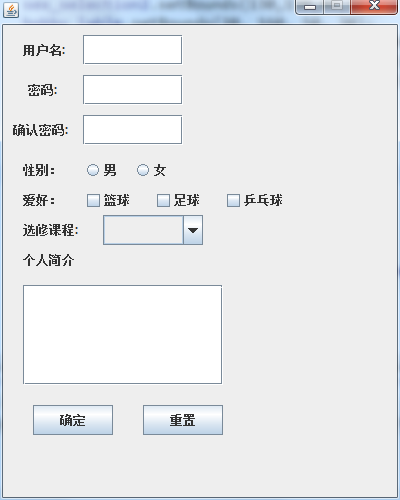
1. 了解Swing的常用组件，了解Swing中常用事件。
2. 可以使用Swing完成简单的界面设计，可以实现简单的人机交互。

## 前提条件

1. 配置完成Eclipse开发环境；
2. 对java中Swing中的常用组件和事件等有一定的了解。
3. 了解Swing中的常用组件JButton、JPanel、JLabel、JTextField、JPasswordField、JCheckedBox、JRadioButton等。

## 实验任务及完成标准

任务一：完成一个如图所示的界面：点击注册按钮时，检测密码和确认密码是否一致，检测用户名是否为空，如果有错误弹出错误提示，如果没有错误，弹出已填写的信息。



步骤：

1. 打开eclipse。
2. 新建一个类文件。
3. 因为是要在窗体中显示所以创建一个JFrame的子类，这样就可以直接从JFrame中继承窗体所有的属性。
4. 在新建的类文件中定义一个类并命名为LogIn。在类中声明所需的组件。

//创建所需的组件

JPanelpanel = **new**JPanel();

ButtonGroupradio\_group = **new**ButtonGroup();

//用户名密码确认密码等

JLabelname\_label;

JTextFieldname\_text;

JLabelpass\_label;

JPasswordFieldpass\_text;

JLabelrepass\_label;

JPasswordFieldrepass\_text;

//性别

JLabelsex\_lable;

JRadioButtonsex\_selection1;

JRadioButtonsex\_selection2;

//爱好

JLabelhobby\_lable;

JCheckBoxhobby\_basketball;

JCheckBoxhobby\_football;

JCheckBoxhobby\_pingpang;

//选修课程

JLabelcource\_lable;

JComboBox<String>course\_box;

//个人简介

JLabelinfo\_lable;

JTextAreainfo\_area;

//确定、重置按钮和提示框

JButtonok\_button;

JButtonreset\_button;

1. 声明好所需的组件以后就可以在构造函数中实例化该组件并且设置该组件的属性等。

在构造函数中可以通过以下步骤来使用这么声明的组件：

将盛放组件的容器的布局设置为null----实例化组件----定义组件的属性和位置-----为组件添加事件----将组件添加到容器中-----显示组件

1. 将盛放组件的容器的布局设置为null：

panel.setLayout(**null**);

1. 实例化各各组件：

name\_label = **new**JLabel("用户名:");

name\_text = **new**JTextField();//用于填写姓名

pass\_label = **new**JLabel("密码:");

pass\_text = **new**JPasswordField();//用于填写密码

repass\_label = **new**JLabel("确认密码:");

repass\_text = **new**JPasswordField();//用于填写确认密码

sex\_lable = **new**JLabel("性别：");

sex\_selection1 = **new**JRadioButton("男");

sex\_selection2 = **new**JRadioButton("女");

hobby\_lable = **new**JLabel("爱好：");

hobby\_basketball = **new**JCheckBox("篮球");

hobby\_football = **new**JCheckBox("足球");

hobby\_pingpang = **new**JCheckBox("乒乓球");

cource\_lable = **new**JLabel("选修课程:");

course\_box = **new**JComboBox<String>();//用于显示选课信息

info\_lable = **new**JLabel("个人简介");

info\_area = **new**JTextArea(5,20);

ok\_button = **new**JButton("确定");

reset\_button = **new**JButton("重置");

1. 定义组件的属性和位置：

name\_label.setBounds(20, 10, 50, 30);

name\_text.setBounds(80, 10, 100, 30);

pass\_label.setBounds(25,50,50,30);

pass\_text.setBounds(80,50,100,30);

pass\_text.setEchoChar('\*');

repass\_label.setBounds(10,90,60,30);

repass\_text.setBounds(80,90,100,30);

sex\_lable.setBounds(20, 130, 50, 30);

sex\_selection1.setBounds(80,130,40,30);

sex\_selection2.setBounds(130,130,40,30);

hobby\_lable.setBounds(20, 160, 50, 30);

hobby\_basketball.setBounds(80, 160, 60, 30);

hobby\_football.setBounds(150, 160, 60, 30);

hobby\_pingpang.setBounds(220, 160, 80, 30);

cource\_lable.setBounds(20, 190, 70, 30);

course\_box.setBounds(100, 190, 100, 30);

info\_lable.setBounds(20, 220, 100, 30);

ok\_button.setBounds(30, 380, 80, 30);

reset\_button.setBounds(140, 380, 80, 30);

info\_area.setLineWrap(**true**); JScrollPanejsp = **new**JScrollPane(info\_area);

jsp.setBounds(20, 260, 200, 100);

panel.add(jsp);

course\_box.insertItemAt("JAVA", 0);

course\_box.insertItemAt("c#", 1);

course\_box.insertItemAt("OS", 2);

1. 为组件添加事件：

ok\_button.addActionListener(**new**ActionListener(){

**publicvoid**actionPerformed(ActionEvent e) {

String str = ""; **if**(!name\_text.getText().equals("")&& !pass\_text.getPassword().toString().equals("") &&!repass\_text.getPassword().toString().equals(""){

**if**(Arrays.*equals*(pass\_text.getPassword(), repass\_text.getPassword()))

{

str += "用户名：" +name\_text.getText() + "\n" +

"性别：" + (sex\_selection1.isSelected()?"男":"") + (sex\_selection2.isSelected()?"女":"")+"\n" +

"爱好："+(hobby\_basketball.isSelected()?"篮球":"")+(hobby\_football.isSelected()?"足球":"")+(hobby\_pingpang.isSelected()?"乒乓球":"")+"\n" +

"选修课程：" +course\_box.getItemAt(course\_box.getSelectedIndex()) + "\n"+

"个人简介:" +info\_area.getText();

}

**else**

{

str = "两次输入的密码不一样。";

}

}

**else**

{

str = "请输入必要的信息。";

}

JOptionPane.*showConfirmDialog*(**null**, str, "您输入的信息", JOptionPane.*CLOSED\_OPTION*);

}

});

reset\_button.addActionListener(**new**ActionListener(){

@Override

**publicvoid**actionPerformed(ActionEvent e) {

name\_text.setText("");

pass\_text.setText("");

repass\_text.setText("");

radio\_group.clearSelection();

course\_box.setSelectedIndex(-1);

info\_area.setText("");

hobby\_basketball.setSelected(**false**);

hobby\_football.setSelected(**false**);

hobby\_pingpang.setSelected(**false**);

}

});

1. 将组件添加到指定的容器：

radio\_group.add(sex\_selection1);

radio\_group.add(sex\_selection2);

panel.add(name\_label);panel.add(name\_text);

panel.add(pass\_label);panel.add(pass\_text);

panel.add(repass\_label);panel.add(repass\_text);panel.add(sex\_lable);panel.add(sex\_selection1);

panel.add(sex\_selection2);panel.add(hobby\_lable);

panel.add(hobby\_basketball);panel.add(hobby\_football);

panel.add(hobby\_pingpang);panel.add(cource\_lable);

panel.add(course\_box);panel.add(info\_lable);

panel.add(ok\_button);panel.add(reset\_button);

1. 将容器添加到窗体上并显示组件和窗体。

**this**.add(panel);

**this**.setResizable(**false**);

**this**.setBounds(300, 300, 400, 500);

**this**.setVisible(**true**);

1. 在主函数中生成类的实例。

**publicstaticvoid** main(String[] args){

**new**LogIn();

}