



LẬP TRÌNH JAVA 3

BÀI 2: COMPONENTS

PHẦN 1

🎯 Kết thúc bài học này bạn có khả năng

❖ Một số components thường dùng

➤ JTextField

➤ JLabel

➤ JButton

➤ JCheckBox

➤ JRadioButton

➤ JTextArea

➤ JPasswordField

➤ JDialog

❖ Bài tập



- ❑ JButton, JList, JLabel, JTextField, JTextArea
- ❑ JComboBox, JRadioButton, JCheckBox
- ❑ JProgressBar, JSlider, JTable, JToolBar, JTree ...
- ❑ JInternalFrame, JScrollPane,
- ❑ JSplitPane, JTabbedPane ...

MỘT SỐ COMPONENTS THƯỜNG DÙNG

DemoSOF203 - NetBeans IDE 8.1

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

<default config>

Projects **Services** **Files**

- DemoMOB102
- DemoSlide7
- DemoSOF203
 - Source Packages
 - Slide1
 - Slide2
 - StudentDetail.java

Form StudentDetail - Navigator

- Form StudentDetail
 - Other Components
 - [JFrame]
 - label jLabel1 [JLabel]
 - txtName [JTextField]
 - label jLabel2 [JLabel]
 - jScrollPane1 [JScrollPane]
 - txaAddress [JTextArea]
 - label jLabel3 [JLabel]
 - jPanel1 [JPanel]
 - radMale [JRadioButton]
 - radFemale [JRadioButton]
 - label jLabel4 [JLabel]
 - cboQualification [JComboBox]
 - label jLabel5 [JLabel]
 - jPanel2 [JPanel]
 - chkReading [JCheckBox]
 - chkSinging [JCheckBox]
 - chkDancing [JCheckBox]
 - OK btnValidate [JButton]
 - OK btnReset [JButton]
 - label jLabel6 [JLabel]

StudentDetail.java

Source Design History

Student Detail

Name: Qualification: Graduate

Address:

Hobby: ☐ Reading ☐ Singing ☐ Dancing

Sex: ☐ Male ☐ Female

Validate Reset

Palette

- Split Pane
- Tool Bar
- Internal Frame
- Swing Controls**
 - File Menu Bar
 - OK Button
 - Check Box
 - Button Group
 - List
 - Text Area
 - Slider
 - Formatted Field
 - Spinner
 - Text Pane
 - Tree
- Swing Menus

Form StudentDetail - Properties

- Code Generation
 - Variables Modifier: private
 - Local Variables: ☐
 - Generate Full Classnames: ☒
 - Generate Mnemonics Code: ☐
 - Listener Generation Style: Anonymous Inner Cl...
 - Layout Generation Style: Standard Java 6 code
- Resources and Internationalization

Form StudentDetail

Output

Apache Tomcat 8.0.27.0 Log Apache Tomcat 8.0.27.0 SOF203 (run)

```
run:
BUILD SUCCESSFUL (total time: 11 seconds)
```

HTTP Server Monitor

1:1

- ❑ JTextField: Ô văn bản cho phép nhập dữ liệu từ bàn phím or dùng để hiển thị dữ liệu
- ❑ Các hàm tạo (Constructor)
 - ❖ JTextField(): tạo text field trống
 - ❖ JTextField(int): A text field with the specified width
 - ❖ JTextField(String): A text field with text
 - ❖ JTextField(String, int): A text field with the specified text and width
 - ❖ JTextField(int cols): khởi tạo JTextField với số cột quy định.
 - ❖ JTextField(String text, int cols): khởi tạo JTextField với dòng text và số cột quy định.

Ví dụ:

```
JTextField txtName= new JTextField(20);  
Và sau đó setText():  
txtName.setText("Hello!");
```

□ JTextField

➤ Methods:

- ✓ void setText(String text): gán giá trị text cho JTextField
Vd: txtHoten.setText("Nguyen Van Teo")
- ✓ String getText(): lấy giá trị text của JTextField
Vd: String s = txtHoten.getText();
- ✓ void setEditable(boolean b): đặt/tắt chế độ nhập
- ✓ trim(): bỏ đi khoảng trắng đầu và cuối chuỗi
- ✓ setFont: thiết lập font cho JTextField
 - ✓ textField.setFont(new Font("Arial", Font.BOLD,12));
- ✓ requestFocus(): chuyển con trỏ về JTextField

➤ Events:

- ✓ *ActionListener*
- ✓ *TextListener*
 - Cài đặt phương thức *textValueChanged()*;

- ❑ JLabel: JLabel thường được dùng để hiển thị text hoặc hình ảnh để tạo các chỉ dẫn, hướng dẫn trên giao diện người dùng.
- ❑ Các hàm tạo (Constructor)
 - ❖ Label() : An empty label
 - ❖ JLabel(String) : A label with the specified text
 - ❖ JLabel(String, int) : A label with the specified text and alignment LEFT, CENTER, and RIGHT.
 - ❖ JLabel(String, Icon, int) : A label with the specified text, icon, and Alignment

□ JLabel

➤ Methods:

- ✓ void setFont (Font f)
- ✓ void setText(String S) quy định chuỗi văn bản.
- ✓ String getText()
- ✓ void setIcon(Icon) quy định Icon
- ✓ getLength(): đưa ra chiều dài của chuỗi text.

➤ Events:

- ✓ mouseClicked

❑ Example:

- ❖ `JLabel lbl=new JLabel("Họ và tên:");`

- ❖ `JLabel lbl=new JLabel("Ngày sinh:");`

❑ Sử dụng HTML để tạo ra các JLabel nhiều dòng, nhiều định dạng

```
JLabel lbHoten = new JLabel("<html>Dòng 1<p  
style=\"color:red;font-size:20\">Dòng  
2</p></html>");
```



- ❑ JButton: là một đối tượng mà cho phép chúng ta khi click chuột vào sẽ thực hiện một việc gì đó
- ❑ Các hàm tạo (Constructor)
 - ❖ JButton() Creates a button with no set text or icon.
 - ❖ JButton(Icon icon) Creates a button with an icon.
 - ❖ JButton(String text) Creates a button with text.
 - ❖ JButton(String text, Icon icon) Creates a button with initial text and an icon.

❏ JButton

➤ Methods:

- ✓ setText (String text)
- ✓ getText ()
- ✓ setForeground (Color fg)
- ✓ setFocusCycleRoot (boolean b)

➤ Events

- ✓ actionPerformed
- ✓ mousePressed

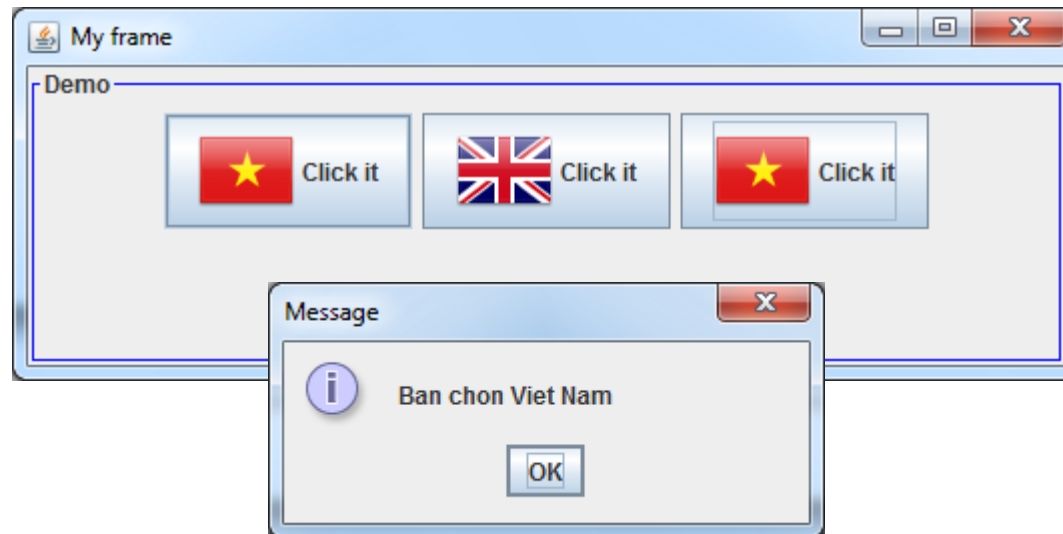
Các cách tạo và bắt sự kiện JButton

```
JButton bt=new JButton("Watch");  
bt.setIcon(new ImageIcon("mywatch.png"));  
bt.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent arg0)  
    {  
        //do something here  
    }  
});
```

❑ Demo 1:

```
public class Demo1 {  
    public static void main(String[] args) {  
        JFrame frame = new JFrame("My frame");  
        frame.setLocation(200, 200);  
        frame.setSize(550, 250);  
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        frame.setVisible(true);  
        ImageIcon usIcon = new ImageIcon("E:\\IconSlide2\\america-usa.png");  
        ImageIcon amIcon = new ImageIcon("E:\\IconSlide2\\united-kingdom.png");  
        ImageIcon vnIcon = new ImageIcon("E:\\IconSlide2\\VietNam.png");  
        JPanel p = new JPanel();  
        JButton b1 = new JButton("Click it", usIcon);  
        JButton b2 = new JButton("Click it", amIcon);  
        JButton b3 = new JButton("Click it", vnIcon);  
        b1.setPressedIcon(amIcon);  
        b1.setRolloverIcon(vnIcon);  
        b3.addActionListener(new ActionListener() {  
            public void actionPerformed(ActionEvent e) {  
                JOptionPane.showMessageDialog(null, "Ban chon Viet Nam");  
            }  
        });  
        p.add(b1); p.add(b2); p.add(b3);  
        p.setBorder(new TitledBorder(new LineBorder(Color.BLUE), "Demo"));  
        frame.add(p);  
    }  
}
```

❑ Demo 1:





DEMO

Chạy và giải thích



❑ Demo 2:

The screenshot displays the NetBeans IDE 8.1 interface. The main window shows a Java Swing GUI titled "JLabel - JTextField - JButton". The GUI contains two text input fields labeled "Nhập tên:" and "Chào bạn:", and a button labeled "Click". A red arrow points from the "Click" button to the Java code below.

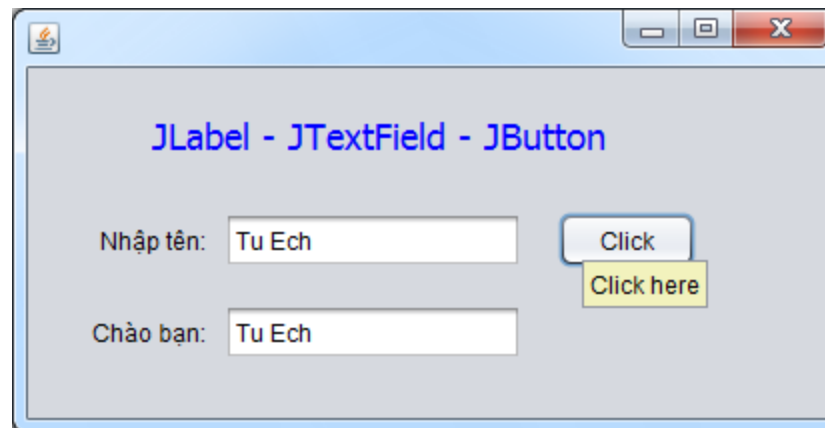
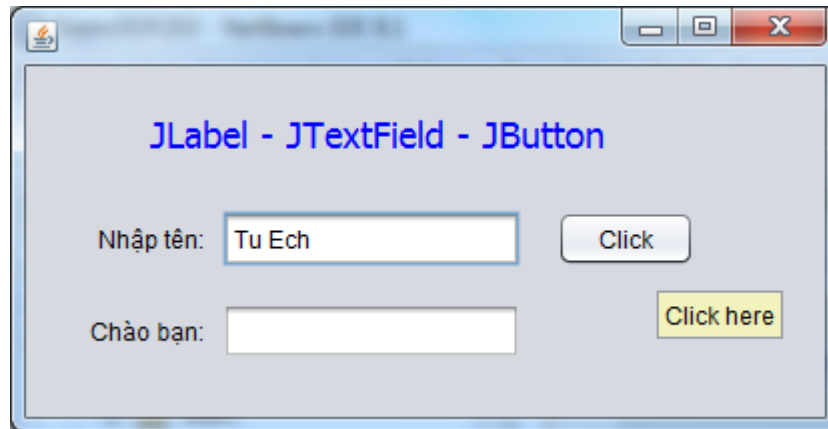
The "Other Components - Navigator" pane on the left shows the project structure, including "Form Demo1" and "Other Components". The "Other Components" list includes:

- label jLabel1 [JLabel]
- txtTen [JTextField]
- label jLabel2 [JLabel]
- txtChao [JTextField]
- OK btnClick [JButton]
- label jLabel3 [JLabel]

The Java code in the main editor window is as follows:

```
private void btnClickActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    String ten = txtTen.getText();  
    txtChao.setText(ten);  
}
```

❑ Demo 2:





DEMO

Chạy và giải thích



- ❑ JCheckBox: là đối tượng cho phép chúng ta chọn nhiều lựa chọn.
- ❑ Các hàm tạo (Constructor):
 - ❖ JCheckBox() Creates an initially unselected check box button with no text, no icon.
 - ❖ JCheckBox(Action a) Creates a check box where properties are taken from the Action supplied.
 - ❖ JCheckBox(Icon icon) Creates an initially unselected check box with an icon.
 - ❖ JCheckBox(Icon icon, boolean selected) Creates a check box with an icon and specifies whether or not it is initially selected.
 - ❖ JCheckBox(String text) Creates an initially unselected check box with text.
 - ❖ JCheckBox(String text, boolean selected)
 - ❖ JCheckBox(String text, Icon icon)
 - ❖ JCheckBox(String text, Icon icon, boolean selected)

JCheckBox

➤ Methods:

- ❖ setSelected (boolean b)
- ❖ isSelected ()
- ❖ setText (String text)
- ❖ getText ()

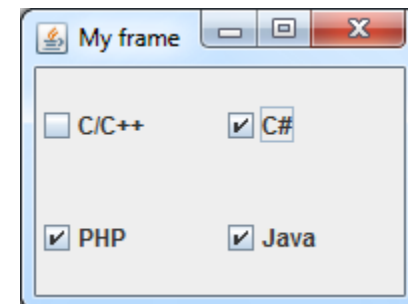
➤ Events:

- ❖ actionPerformed

❑ JCheckBox

❑ Ví dụ:

```
public class Demo3CheckBox {  
    public static void main(String[] args) {  
        JFrame frame = new JFrame("My frame");  
        frame.setLocation(200,200);  
        frame.setSize(150,150);  
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        frame.setVisible(true);  
        JPanel panelCheck=new JPanel();  
        panelCheck.setLayout(new GridLayout(2, 2));  
        JCheckBox chk1=new JCheckBox("C/C++");  
        JCheckBox chk2=new JCheckBox("C#");  
        JCheckBox chk3=new JCheckBox("PHP");  
        JCheckBox chk4=new JCheckBox("Java");  
        panelCheck.add(chk1);  
        panelCheck.add(chk2);  
        panelCheck.add(chk3);  
        panelCheck.add(chk4);  
        frame.add(panelCheck);  
    }  
}
```





DEMO

Chạy và giải thích



- ❑ JRadioButton: là đối tượng cho phép chúng ta chọn một trong nhiều lựa chọn.
- ❑ Các hàm tạo (Constructor):
 - ❖ JRadioButton() Creates an initially unselected radio button with no set text.
 - ❖ JRadioButton(Icon icon) Creates an initially unselected radio button with the specified image but no text.
 - ❖ JRadioButton(Icon icon, boolean selected) Creates a radio button with the specified image and selection state, but no text.
 - ❖ JRadioButton(String text) Creates an unselected radio button with the specified text.
 - ❖ JRadioButton(String text, boolean selected)
 - ❖ JRadioButton(String text, Icon icon) Creates a radio button that has the specified text and image, and that is initially unselected.
 - ❖ JRadioButton(String text, Icon icon, boolean selected)
- ❑ Lưu ý: Phải đưa tất cả các JRadioButton vào trong một ButtonGroup

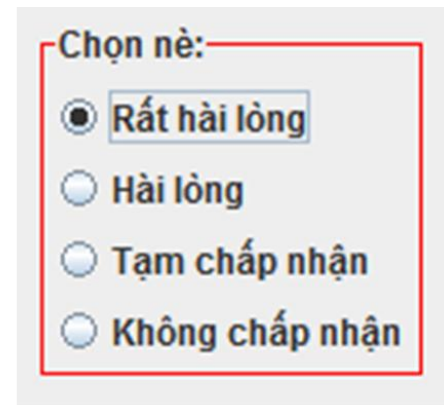
❑ JRadioButton

❑ Methods:

- ❖ setSelected (boolean b)
- ❖ isSelected()
- ❖ setText (String text)
- ❖ getText ()

❑ Events:

- ❖ actionPerformed

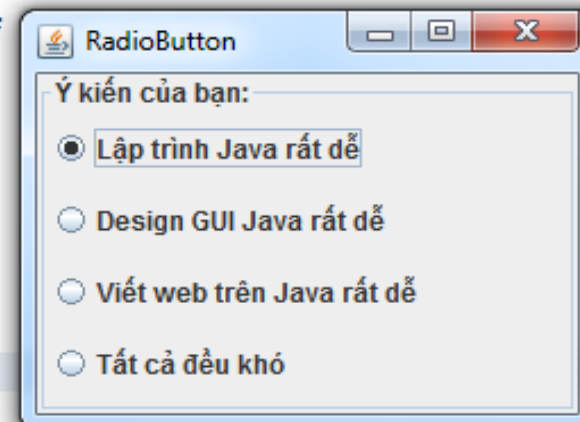


```
if(rad1.isSelected())  
{  
    //...  
}
```

❑ JRadioButton

❑ Ví dụ:

```
public class Demo4RadioButton {  
    public static void main(String[] args) {  
        JFrame frame = new JFrame("RadioButton");  
        frame.setLocation(200,200);  
        frame.setSize(200,170);  
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        frame.setVisible(true);  
        JPanel panelGroup=new JPanel();  
        panelGroup.setBorder(new TitledBorder("Ý kiến của bạn:"));  
        panelGroup.setLayout(new GridLayout(4, 1));  
        JRadioButton rad1=new JRadioButton("Lập trình Java rất dễ");  
        JRadioButton rad2=new JRadioButton("Design GUI Java rất dễ");  
        JRadioButton rad3=new JRadioButton("Viết web trên Java rất dễ");  
        JRadioButton rad4=new JRadioButton("Tất cả đều khó");  
        ButtonGroup group=new ButtonGroup();  
        group.add(rad1);  
        group.add(rad2);  
        group.add(rad3);  
        group.add(rad4);  
        panelGroup.add(rad1);  
        panelGroup.add(rad2);  
        panelGroup.add(rad3);  
        panelGroup.add(rad4);  
        frame.add(panelGroup);  
    }  
}
```



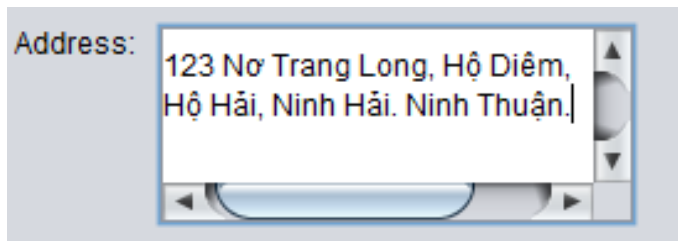


DEMO

Chạy và giải thích



- ❑ JTextArea: là một thành phần cho phép hiển thị nhiều dòng văn bản đồng thời người dùng có thể chỉnh sửa văn bản
- ❑ Các hàm tạo (Constructor):
 - ❖ JTextArea(): Create a default text area.
 - ❖ JTextArea(int rows, int columns): Create a text area with the specified number of rows and columns.
 - ❖ JTextArea(String text)
 - ❖ JTextArea(String text, int rows, int columns)
 - ❖ JTextArea(Document doc): Create a text area that uses the specified Document.
 - ❖ JTextArea(Document doc, String text, int rows, int columns)



□ JTextArea

➤ Methods

- ✓ `setWrapStyleWord(true);`
- ✓ `setLineWrap(true);`
- ✓ `setText("");`
- ✓ `setFont(font);`
- ✓ `setForeground(Color.BLUE);`
- ✓ `void append(String str):` Append the given text to the end of the document.
- ✓ `void insert(String str, int pos):` Insert the specified text at the given position . To insert text at the beginning of the document, use a position of 0.
- ✓ ...

- ❑ JPasswordField: là đối tượng cho phép chúng ta nhập vào một dòng text giống như JTextField nhưng được ẩn bởi các dấu sao (*) hoặc chấm tròn để tạo nên mật khẩu (password)
- ❑ Các hàm tạo (Constructor):
 - ❖ JPasswordField(): Khởi tạo JPasswordField không có text và độ rộng là 0 cột
 - ❖ JPasswordField(int columns): Khởi tạo JPasswordField không có text và độ rộng là columns cột
 - ❖ JPasswordField(String text): Khởi tạo JPasswordField với text ban đầu
 - ❖ JPasswordField(String text, int columns): Khởi tạo JPasswordField với text ban đầu và rộng columns cột

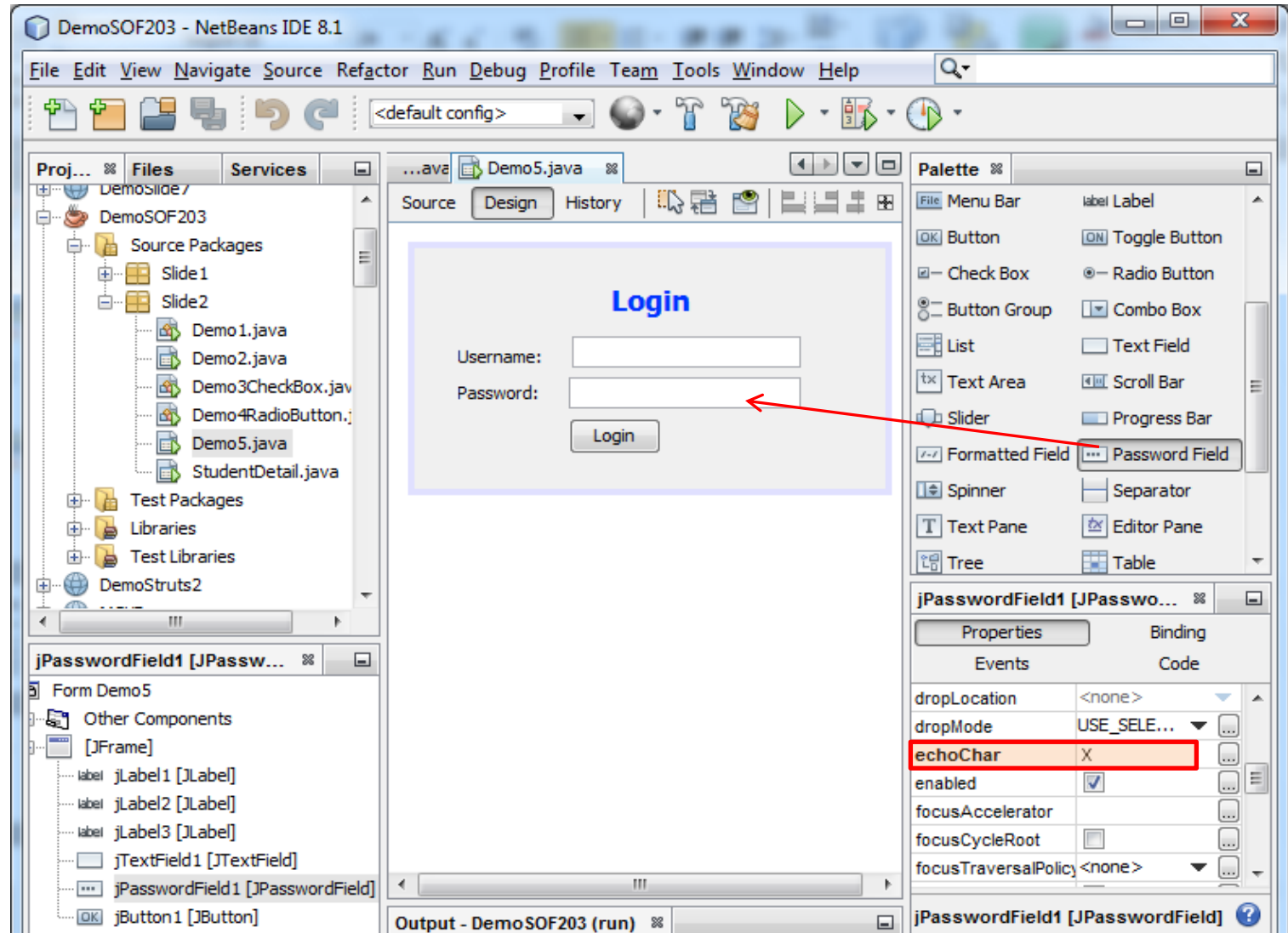
❑ JPasswordField

❑ Methods

- ✓ JPasswordField(String text, int columns)
- ✓ char[] getPassword(): returns the text contained in this password field JPasswordField
- ✓ setEchoChar(char): replacing each input character with the specified character
- ✓ JPasswordField pass = new JPasswordField(20);
- ✓ pass.setEchoChar('#');

□ JPasswordField

□ Ví dụ:





DEMO

Chạy và giải thích





LẬP TRÌNH JAVA 3

BÀI 2: COMPONENTS

PHẦN 2

□ JDialog

- ❖ Hộp thoại thường là cửa sổ phụ trình bày danh sách tùy chọn hoặc hiển thị thông điệp
- ❖ Hộp thoại thường cung cấp nút xác nhận thay đổi hoặc trường nhập vào thành phần, hủy bỏ thay đổi...
- ❖ Có 3 dạng hộp thoại:
 - Hộp thoại thông điệp
 - Hộp thoại xác nhận
 - Hộp thoại tập tin (JFileChooser)

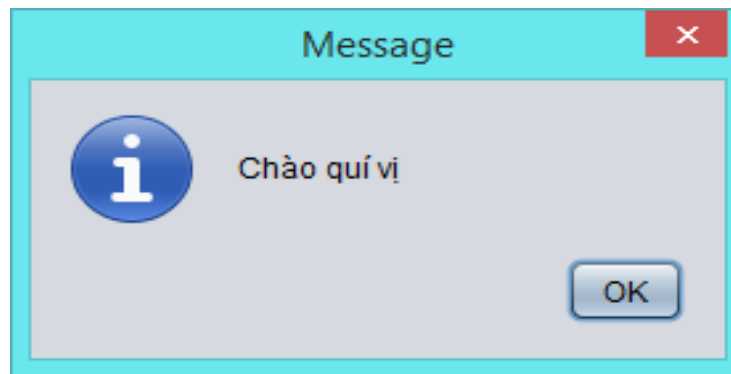
❑ JDialog

- ❖ JDialog có 2 trạng thái
 - Modal: Khi Jdialog thực hiện xong mới được phép thao tác lên form cha .
 - Modeless: Sau khi hiển thị dialog, người dùng có thể thao tác lên form cha
- ❖ JDialog thường được sử dụng với trạng thái Modal
- ❖ Khởi tạo
- ❖ `public JDialog()`
- ❖ `public JDialog(Dialog owner, String title, boolean modal, GraphicsConfiguration gc)`
- ❖ `public JDialog(Frame owner, String title, boolean modal, GraphicsConfiguration gc)`

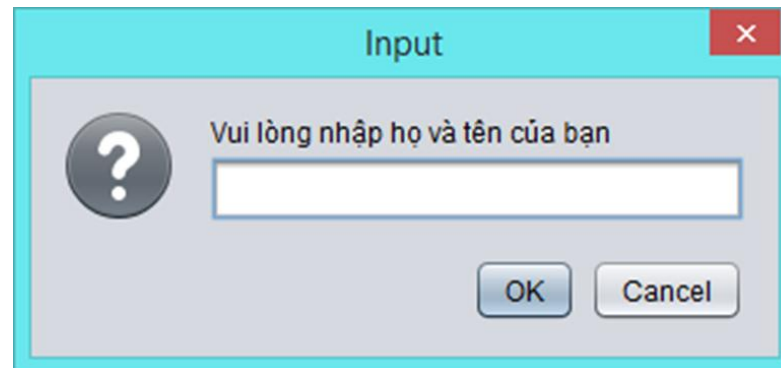
❑ JDialog

❑ Ví dụ:

✓ JOptionPane.showMessageDialog(this,"Chào quý vị");



✓ String input =JOptionPane.showInputDialog(this,"Vui lòng nhập họ và tên của bạn");



❑ JDialog

❑ Ví dụ:

```
int choice=JOptionPane.showConfirmDialog(this,"Bạn có người yêu chưa?");
```

```
If(JOptionPane.YES_OPTION==choice) {
```

```
//do something
```

```
}
```

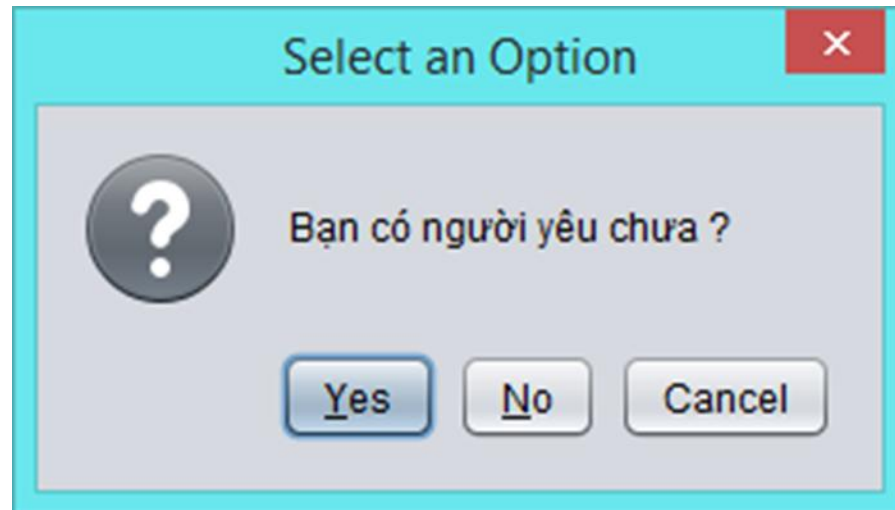
```
else if(JOptionPane.NO_OPTION==choice){
```

```
//do something
```

```
}
```

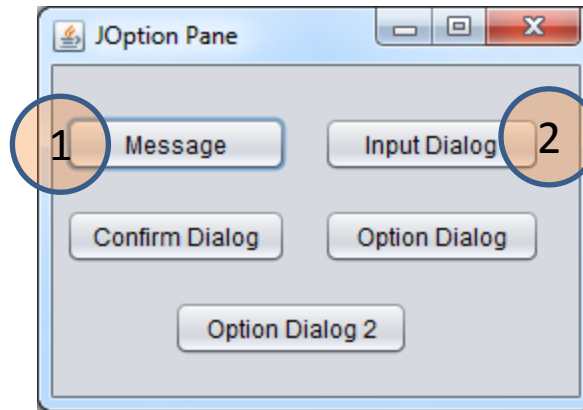
```
else{
```

```
}
```



❑ JDialog

❑ Ví dụ:



1

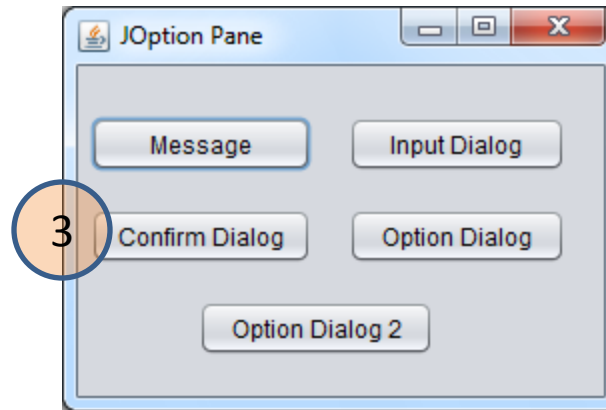
```
private void btnMessageActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    JOptionPane.showMessageDialog(this, "Hi! chao cac ban");  
}
```

2

```
private void btnInputDialogActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    String name=JOptionPane.showInputDialog("Nhap ho ten:");  
    if (name!=null) {  
        JOptionPane.showMessageDialog(this, "Chao ban "+name);  
    }  
}
```

❑ JDialog

❑ Ví dụ:



3

```
private void btnConfirmActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    int ret = JOptionPane.showConfirmDialog(this, "Are you sure?"  
        , "Confirm", JOptionPane.YES_NO_OPTION  
        , JOptionPane.QUESTION_MESSAGE);  
    if(ret == JOptionPane.YES_OPTION) {  
        //Xử lý trường hợp người dùng chọn Yes  
        JOptionPane.showMessageDialog(this, "Ban chon: YES ");  
    } else {  
        //Xử lý trường hợp người dùng chọn No  
        JOptionPane.showMessageDialog(this, "Ban chon: NO");  
    }  
}
```

❑ JDialog

❑ Ví dụ:

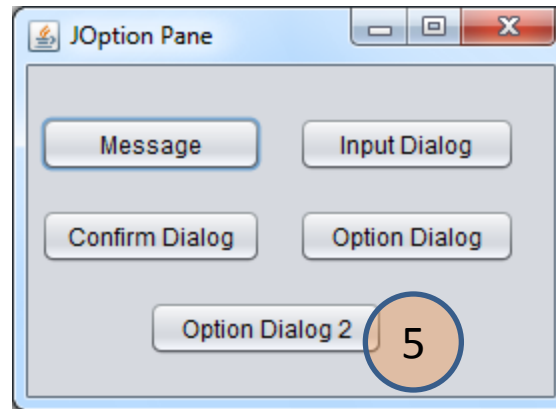


4

```
private void btnOptionDialogActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    Object[] options = {"Yes, please", "No way!"};  
    int n = JOptionPane.showOptionDialog(this, "Would you like green eggs and ham?",  
        "A Silly Question", JOptionPane.YES_NO_OPTION, JOptionPane.QUESTION_MESSAGE,  
        null, //do not use a custom Icon  
        options, //the titles of buttons  
        options[0]); //default button title  
}
```

❑ JDialog

❑ Ví dụ:



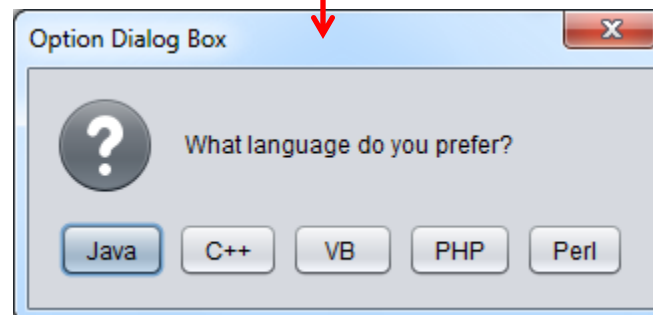
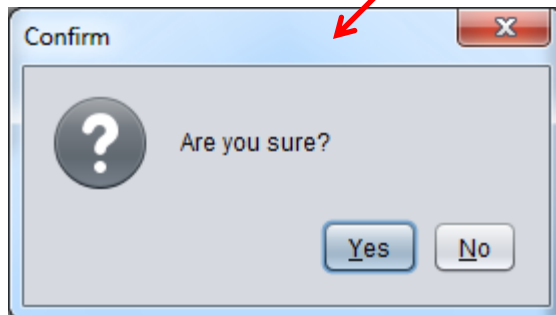
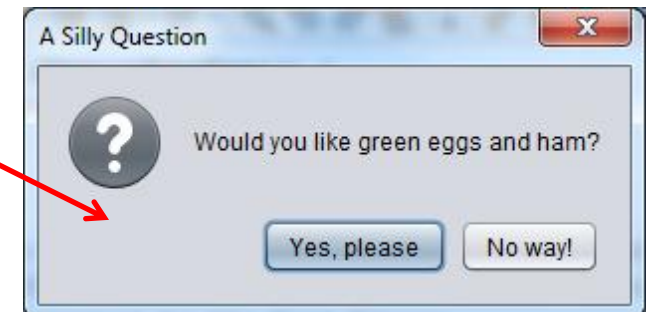
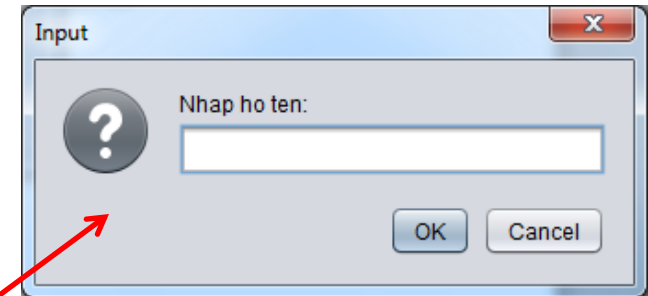
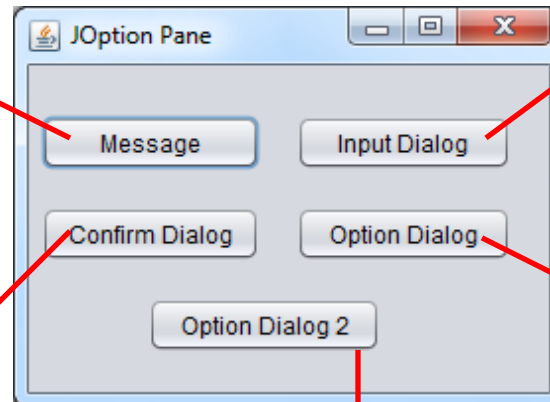
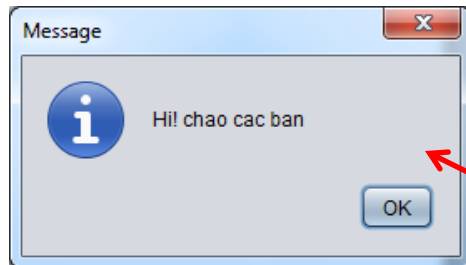
5

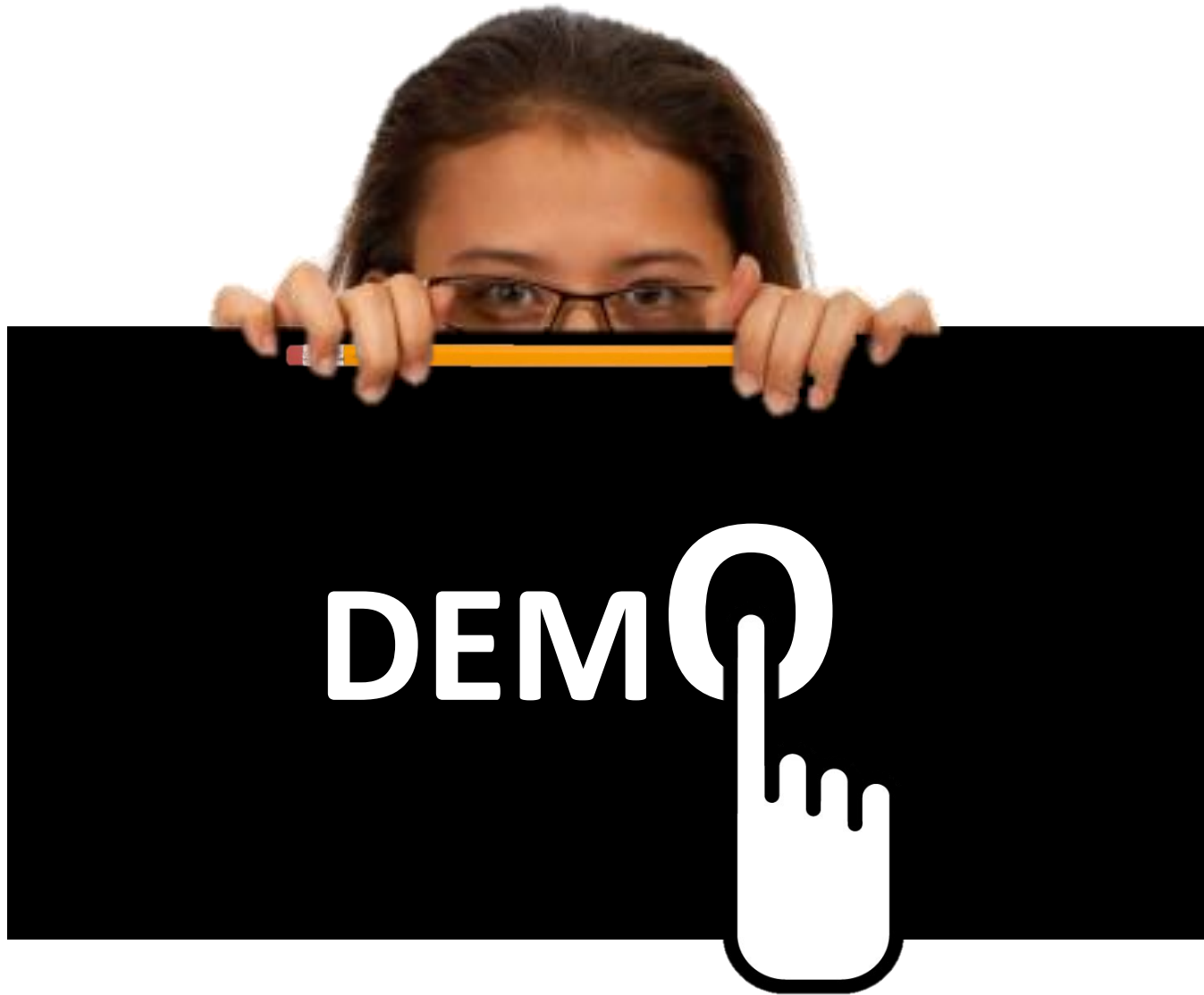
```
private void btnOptionDialog2ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    int messageType = JOptionPane.QUESTION_MESSAGE;  
    String[] options = {"Java", "C++", " VB ", "PHP", "Perl"};  
    int code = JOptionPane.showOptionDialog(this,  
        "What language do you prefer?",  
        "Option Dialog Box", 0, messageType,  
        null, options, "Java");  
    JOptionPane.showMessageDialog(this, "Answer: "+code);  
}
```


MỘT SỐ COMPONENTS THƯỜNG DÙNG

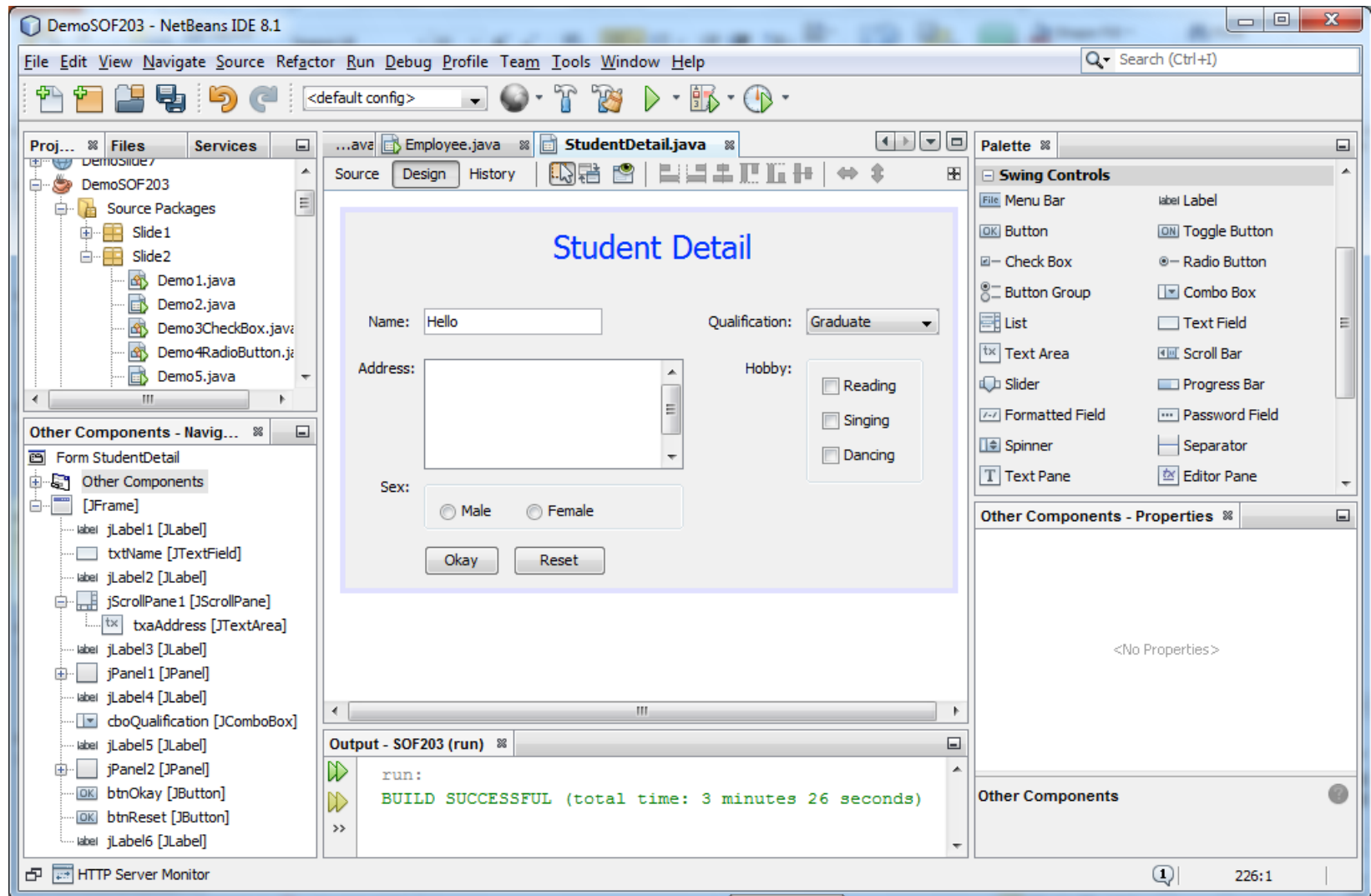
❑ JDialog

❑ Ví dụ:





- Yêu cầu: Thiết kế form như hình, khi nhấn nút Okay hiển thị tất cả các thông tin đã chọn



❖ Một số components thường dùng

- JTextField
- JLabel
- JButton
- JCheckBox
- JRadioButton
- JTextArea
- JPasswordField
- JDialog

❖ Bài tập





Cảm ơn