



FPT POLYTECHNIC



LẬP TRÌNH JAVA 2

BÀI 2: SWING INTRODUCTION

PHẦN 1: THIẾT KẾ GIAO DIỆN

www.poly.edu.vn

- ❑ Kết thúc bài học này bạn có khả năng
 - ❖ Thiết kế được giao diện đồ họa đơn gian
 - ❖ Xử lý được sự kiện click chuột
 - ❖ Đọc dữ liệu từ giao diện người dung
 - ❖ Hiển thị được dữ liệu lên giao diện người dùng

*Chú ý: Bài học này chỉ có mục đích giúp sinh viên sử dụng giao diện đồ họa một cách cơ bản để thay thế Console bằng Graphics. Swing sẽ được học một cách đầy đủ ở môn **Java 3**. Không nên sa đà vào vấn đề này làm sai lệch mục tiêu của môn học.*

- ❑ Ứng dụng giao diện đồ họa là ứng dụng sử dụng giao diện đồ họa để giao tiếp với người dùng
 - ❖ Nhập và hiển thị dữ liệu từ các thành phần đồ họa

QUẢN LÝ NHÂN VIÊN

10:22 AM

MÃ NHÂN VIÊN

HỌ VÀ TÊN

TUỔI

EMAIL

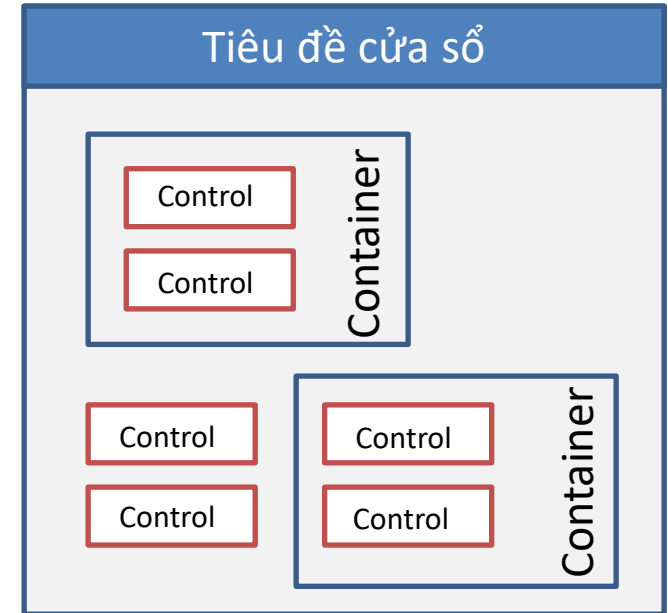
LƯƠNG

< << >> >| Record: 1 of 10

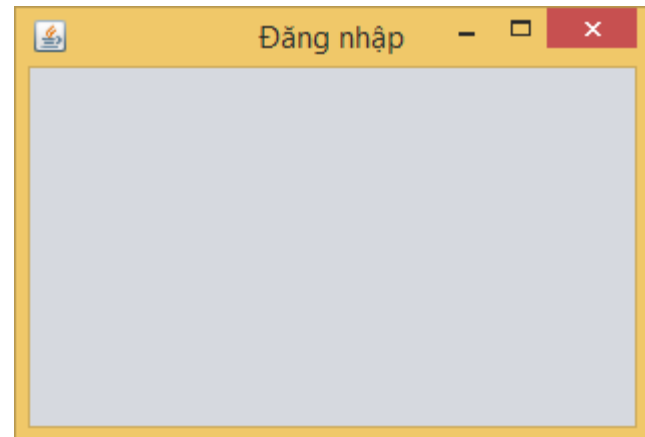
New
Save
Delete
Find
Open
Exit

MÃ	HỌ VÀ TÊN	TUỔI	EMAIL	LƯƠNG
----	-----------	------	-------	-------

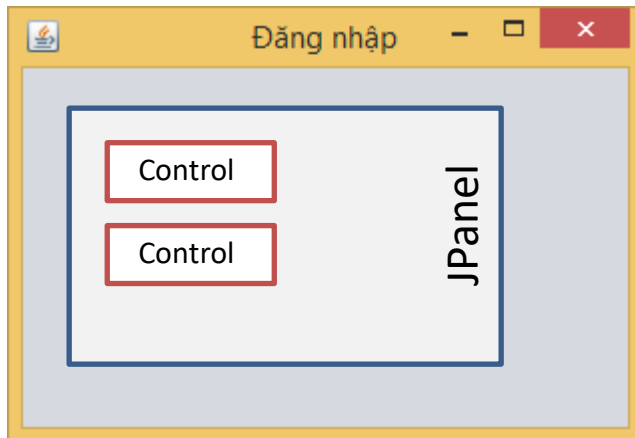
- ❑ Cửa sổ - Window
 - ❖ Cửa sổ ứng dụng.
- ❑ Khung chứa – Container
 - ❖ Chứa một hoặc nhiều điều khiển
- ❑ Điều khiển – Control
 - ❖ TextField, Radio, CheckBox, Button, Table...
- ❑ Chú ý:
 - ❖ Một ứng dụng có thể có nhiều cửa sổ.
 - ❖ Một khung chứa có thể chứa nhiều điều khiển hoặc khung chứa khác



- ❑ Trong Swing, JFrame được sử dụng để tạo cửa sổ
- ❑ Các thành phần liên quan
 - ❖ Biểu tượng - Icon
 - ❖ Tiêu đề - Title
 - ❖ Thu, phóng, tắt cửa sổ



- ❑ Trong Swing có nhiều loại Container. Trong bài này chúng ta chỉ tìm hiểu JPanel
- ❑ JPanel là container được sử dụng để nhóm các control hoặc container khác.



- ❑ Control là các phần tử giao diện được sử dụng để tiếp nhận dữ liệu từ người dung hoặc trình bày dữ liệu đến người dung
- ❑ Các Control thường dung
 - ❖ JLabel - Nhãn
 - ❖ JTextField – Ô nhập
 - ❖ JButton – Nút nhấn
 - ❖ JCheckBox – Hộp kiểm
 - ❖ JRadioButton – Lựa chọn
 - ❖ JComboBox – Hộp chọn
 - ❖ JTable – Bảng
 - ❖ JOptionPane – Hộp thoại
 - ❖ ...

❑ JLabel là điều khiển được sử dụng để tạo nhãn.

❖ Thuộc tính quan trọng nhất của JLabel là Text

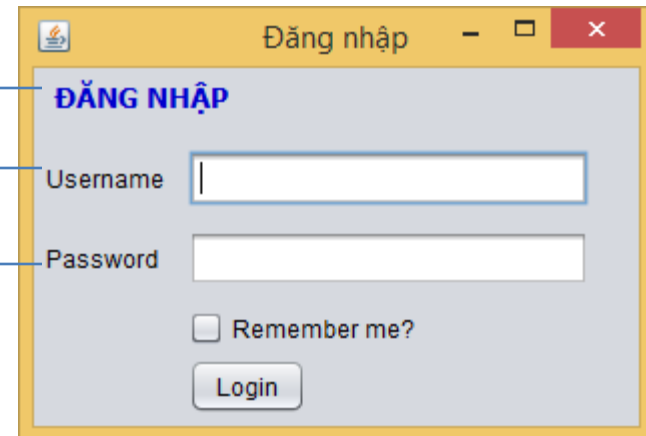
❖ Khi đặt tên bạn cần bắt đầu bởi lbl.

❖ Ví dụ:

➤ lblTitle

➤ lblUsername

➤ lblPassword

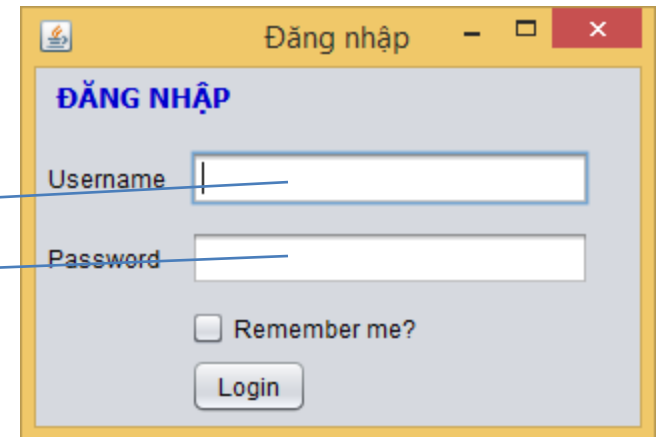


❑ JTextField được sử dụng để tạo ô nhập. 2 thuộc tính thường dùng là Text và Name

- ❖ Text: giá trị của ô nhập
- ❖ Name: tên của ô nhập, nên bắt đầu bởi txt
- ❖ Ví dụ:
 - txtUsername
 - txtPassword

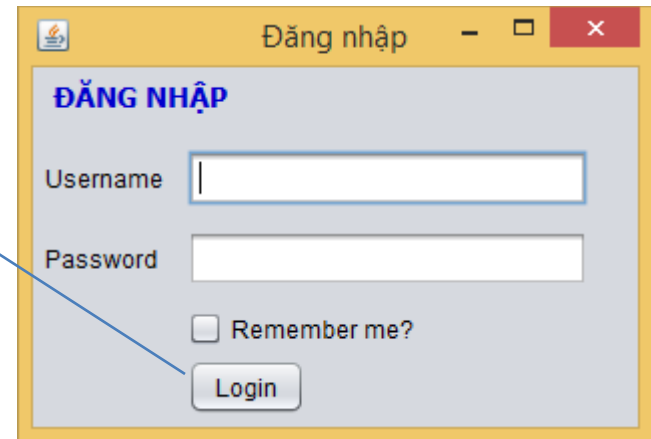
❑ JTextArea chỉ khác JTextField là cho phép chứa Text nhiều dòng

❑ JPasswordField thì không cho nhìn thấy nội dung của Text



❑ JButton được sử dụng để tạo nút nhấn. Các thuộc tính thường sử dụng

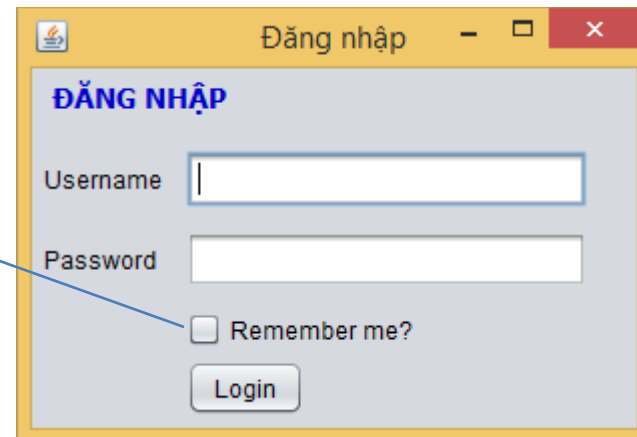
- ❖ Text: nhãn của nút
- ❖ Name: tên của nút, nên bắt đầu bởi btn
- ❖ Ví dụ:
 - btnLogin



❑ JCheckBox được sử dụng để tạo hộp kiểm. Các thuộc tính thường dùng

- ❖ Text: Nhãn đính kèm
- ❖ Selected: Trạng thái
- ❖ Name: Tên, nên bắt đầu bởi chk
- ❖ Ví dụ:

➤ chkRemember

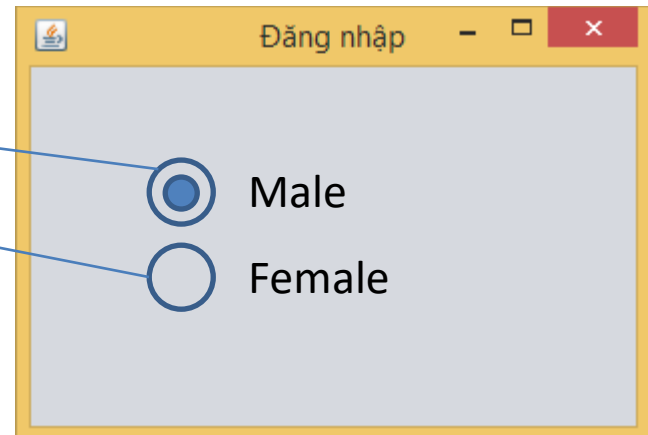


The image shows a screenshot of a web application window titled "Đăng nhập" (Login). Inside the window, there is a section titled "ĐĂNG NHẬP" in blue. Below the title, there are two input fields: "Username" and "Password". Below the "Password" field, there is a checkbox labeled "Remember me?". Below the checkbox, there is a "Login" button. A blue arrow points from the text "chkRemember" in the previous block to the "Remember me?" checkbox.

❑ JRadioButton được sử dụng để tạo các mục chọn. Các thuộc tính thường dùng

- ❖ Text: Nhãn đính kèm
- ❖ Selected: Trạng thái
- ❖ Name: Tên, nên bắt đầu bởi rdo
- ❖ ButtonGroup: nhóm mà radio thuộc vào. Trong mỗi nhóm bạn chỉ có thể chọn một radio.
- ❖ Ví dụ:

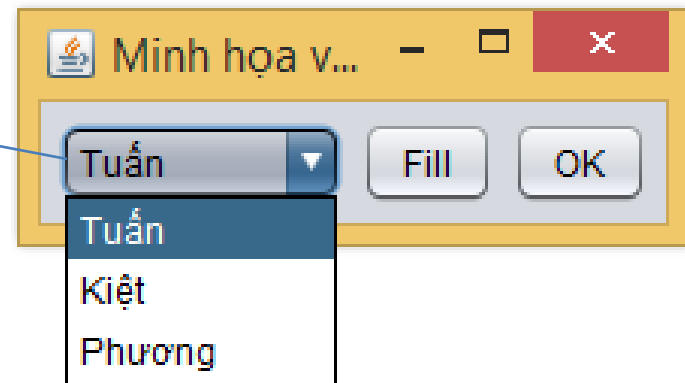
- rdoMale
- rdoFemale



❑ JComboBox được sử dụng để tạo danh sách xổ xuống và chỉ cho phép chọn 1. Các thuộc tính thường sử dụng.

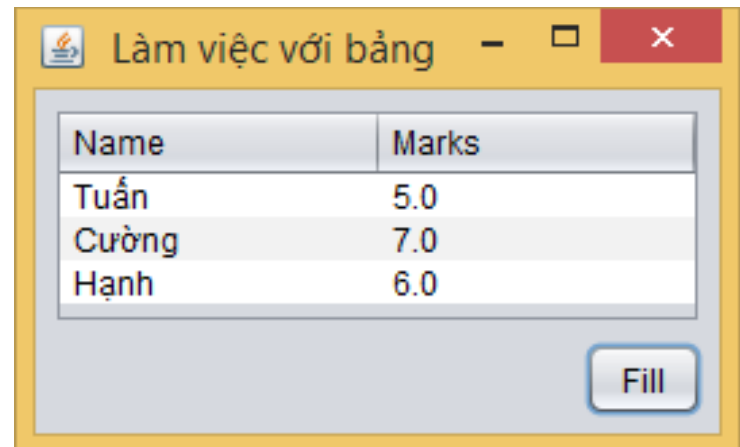
- ❖ Model: chứa danh sách dữ liệu
- ❖ SelectedIndex: vị trí mục được chọn
- ❖ SelectedItem: dữ liệu mục chọn
- ❖ Name: Tên, nên bắt đầu bởi cbo
- ❖ Ví dụ:

➤ cboNames



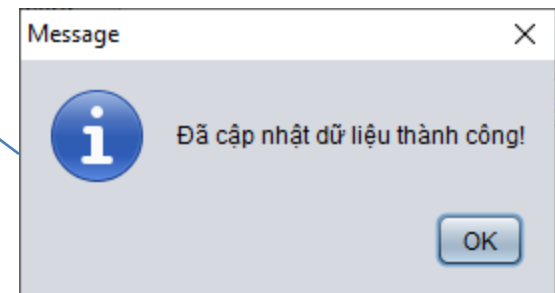
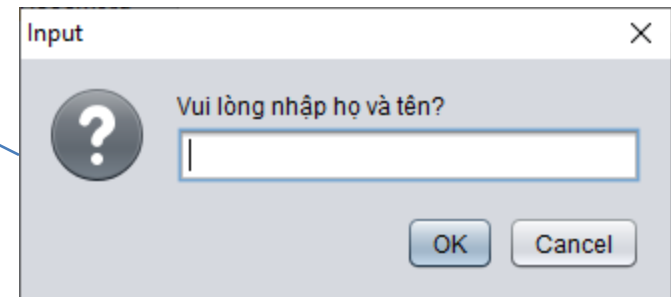
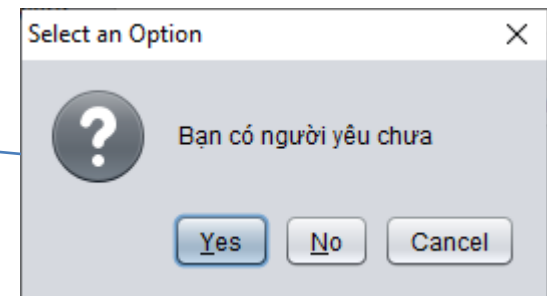
❑ JTable được sử dụng để biểu diễn dữ liệu dạng bảng. Các thuộc tính thường dùng

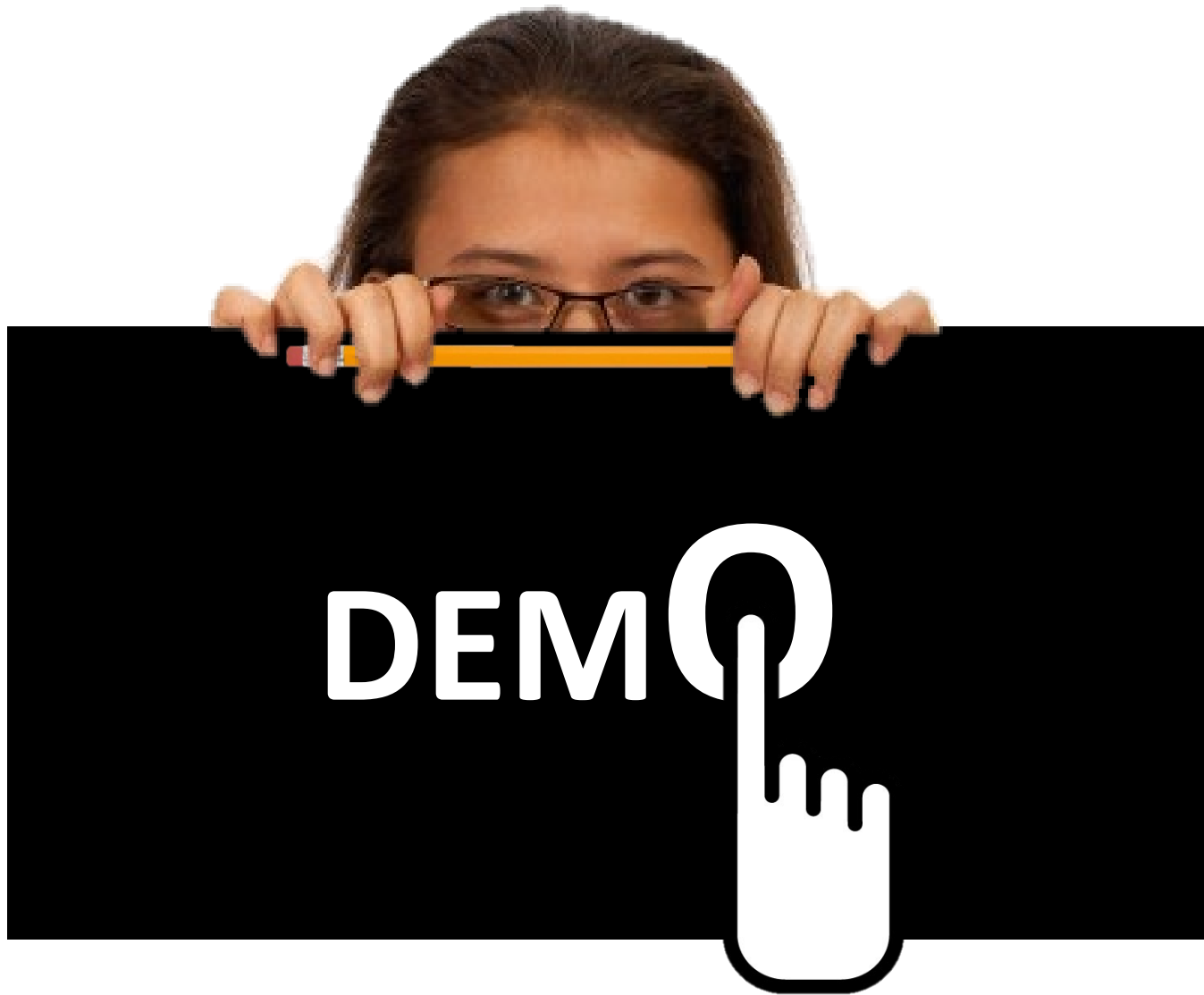
- ❖ Model: dữ liệu bảng
- ❖ SelectionMode: chế độ chọn các hàng
- ❖ RowHeight: chiều cao mỗi hàng
- ❖ Name: Tên, nên bắt đầu bởi tbl
- ❖ Ví dụ:
 - tblStudents



❑ JOptionPane là một dạng cửa sổ được sử dụng để tạo các hộp thoại thông báo. Có 3 dạng thông báo thường dùng

- ❖ Xác nhận – Confirm
- ❖ Nhập – Input
- ❖ Thông báo - Message







FPT POLYTECHNIC



LẬP TRÌNH JAVA 2

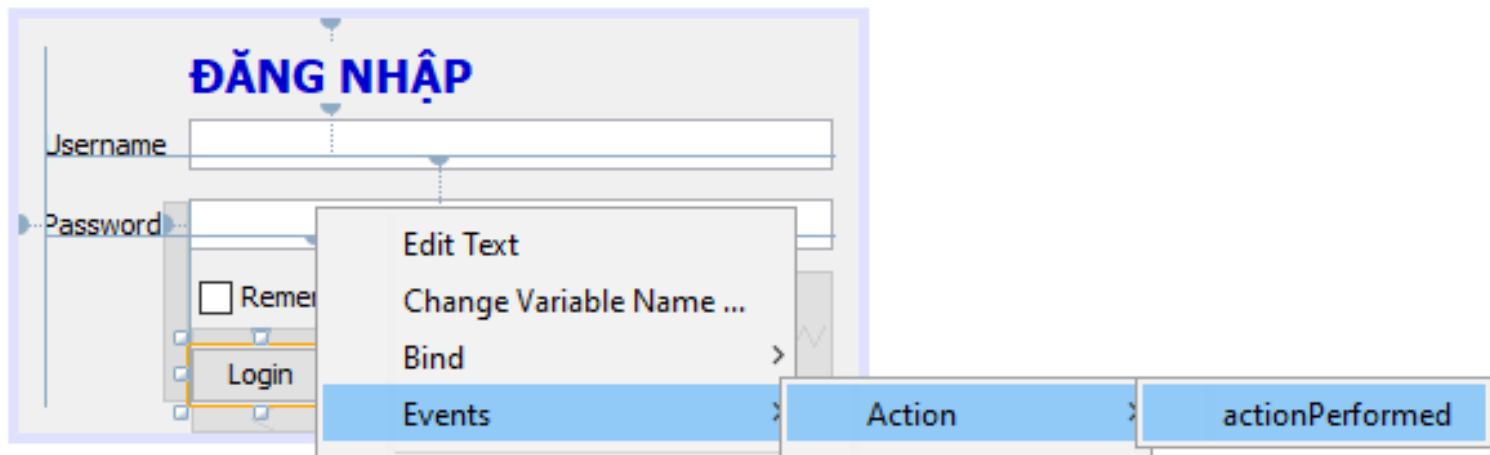
BÀI 2: SWING INTRODUCTION

PHẦN 2: LẬP TRÌNH GIAO DIỆN

www.poly.edu.vn

- ❑ Lập trình giao diện là công việc lập trình để
 - ❖ Thu nhận dữ liệu từ người dùng
 - ❖ Hiển thị dữ liệu để người dùng xem
 - ❖ Thay đổi giao diện tùy thuộc vào dữ liệu
- ❑ Lập trình giao diện thường gắn kết với các sự kiện xảy ra trong quá trình sử dụng
- ❑ Sự kiện phát sinh từ giao diện rất phong phú (sẽ được học trong Java 3). Trong bài học này chúng ta chỉ nghiên cứu sự kiện Action.

- ❑ Khi bạn click chuột vào nút thì sự kiện Action xảy ra.
- ❑ Bạn cần viết mã để điều khiển sự kiện này
 - ❖ Đọc dữ liệu từ các điều khiển
 - ❖ Xử lý theo yêu cầu nghiệp vụ
 - ❖ Hiển thị, thay đổi dữ liệu trên giao diện
- ❑ Nhấp đúp vào nút hoặc thực hiện theo hình sau

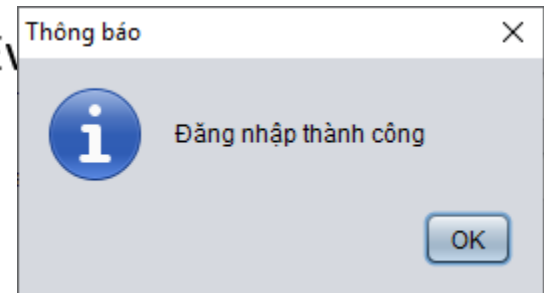


- ❑ Đặt code xử lý của bạn ngay sau dòng comment TODO để thực hiện công việc của bạn.

```
private void btnLoginActionPerformed(ActionEvent evt) {  
    // TODO add your handling code here:  
}
```

- ❑ Ví dụ: Khi click vào sẽ hiển thị hộp thoại như sau

```
private void btnLoginActionPerformed(ActionEvent evt) {  
    // TODO add your handling code here:  
    JOptionPane.showMessageDialog(this,  
        "Đăng nhập thành công", "Thông báo",  
        JOptionPane.INFORMATION_MESSAGE);  
}
```



❑ Để đọc/ghi giá trị thuộc tính **XYZ** của Control bạn sử dụng:

❖ Đọc (lấy dữ liệu từ control)

➤ **Type** value = control.get**XYZ**()

❖ Ghi (thay đổi dữ liệu trên control)

➤ control.set**XYZ**(**Type** value)

❑ Ví dụ:

❖ Đọc

➤ String username = txtUsername.getText();

❖ Ghi

➤ txtUsername.setText("TeoNV")

❑ Để đọc giá trị của Username, Password và Remember

❖ String username = txtUsername.getText();

❖ **Boolean** remember = chkRemember.**is**Selected();

❑ Để ghi dữ liệu lên các điều khiển

❖ txtUsername.setText("");

❖ chkRemember.**set**Selected(**false**);

❑ Chú ý:

❖ Khi kiểu thuộc tính là Boolean, thì theo quy ước của JavaBean chúng ta sử dụng **is**Xyz() thay vì **get**Xyz()

- ❑ Với JComboBox, bạn cần lập trình thực hiện các công việc sau:
 - ❖ Đổ dữ liệu vào JComboBox
 - ❖ Làm việc với các mục
 - ❖ Làm việc với vị trí và giá trị của mục chọn được chọn

- ❑ Đổ dữ liệu vào JComboBox bạn sử dụng đoạn mã sau đây

```
String[] data = {  
    "Việt Nam", "United States", "Singapore", "Malaysia"  
};  
cboCountry.setModel(new DefaultComboBoxModel(data));
```

- ❑ Làm việc với mục chọn

```
cboCountry.setSelectedIndex(1)  
int index = cboCountry.getSelectedIndex()  
String country = (String)cboCountry.getSelectedItem()
```


- ❑ Các phương thức sau đây giúp bạn làm việc với các mục trong JComboBox
 - ❖ `cboCountry.getItemCount()`
 - ❖ `cboCountry.getItemAt(index)`
 - ❖ `cboCountry.addItem(item)`
 - ❖ `cboCountry.removeItem(item)`
 - ❖ `cboCountry.removeItemAt(index)`
 - ❖ `cboCountry.removeAllItems()`

- ❑ Với JTable, chúng ta cần lập trình thực hiện được các công việc sau:
 - ❖ Đổ dữ liệu vào bảng
 - ❖ Làm việc với các hàng được chọn
 - ❖ Làm việc với các hàng

❑ Bạn có thể đổ dữ liệu vào bảng bằng đoạn mã sau:

```
String [] columns = {  
    "NO", "FULLNAME", "EMAIL", "COUNTRY", "GENDER", "STATUS"  
}  
  
Object[][] rows = {  
    {1, "Nguyễn Chí Phèo", "PheoNC@gmail.com", true, false},  
    {2, "Phạm Thị Nở", "NoPT@gmail.com", false, true}  
}  
  
tblStaffList.setModel(new DefaultTableModel(rows, columns))
```

❑ Lập trình lấy vị trí các hàng được chọn

- ❖ `int index = tblStaffList.getSelectedRow();`
 - Vị trí hàng được chọn đầu tiên
- ❖ `int count = tblStaffList.getSelectedRowCount();`
 - Số hàng được chọn
- ❖ `int[] indices = tblStaffList.getSelectedRows();`
 - Vị trí các hàng được chọn

❑ Lấy Model

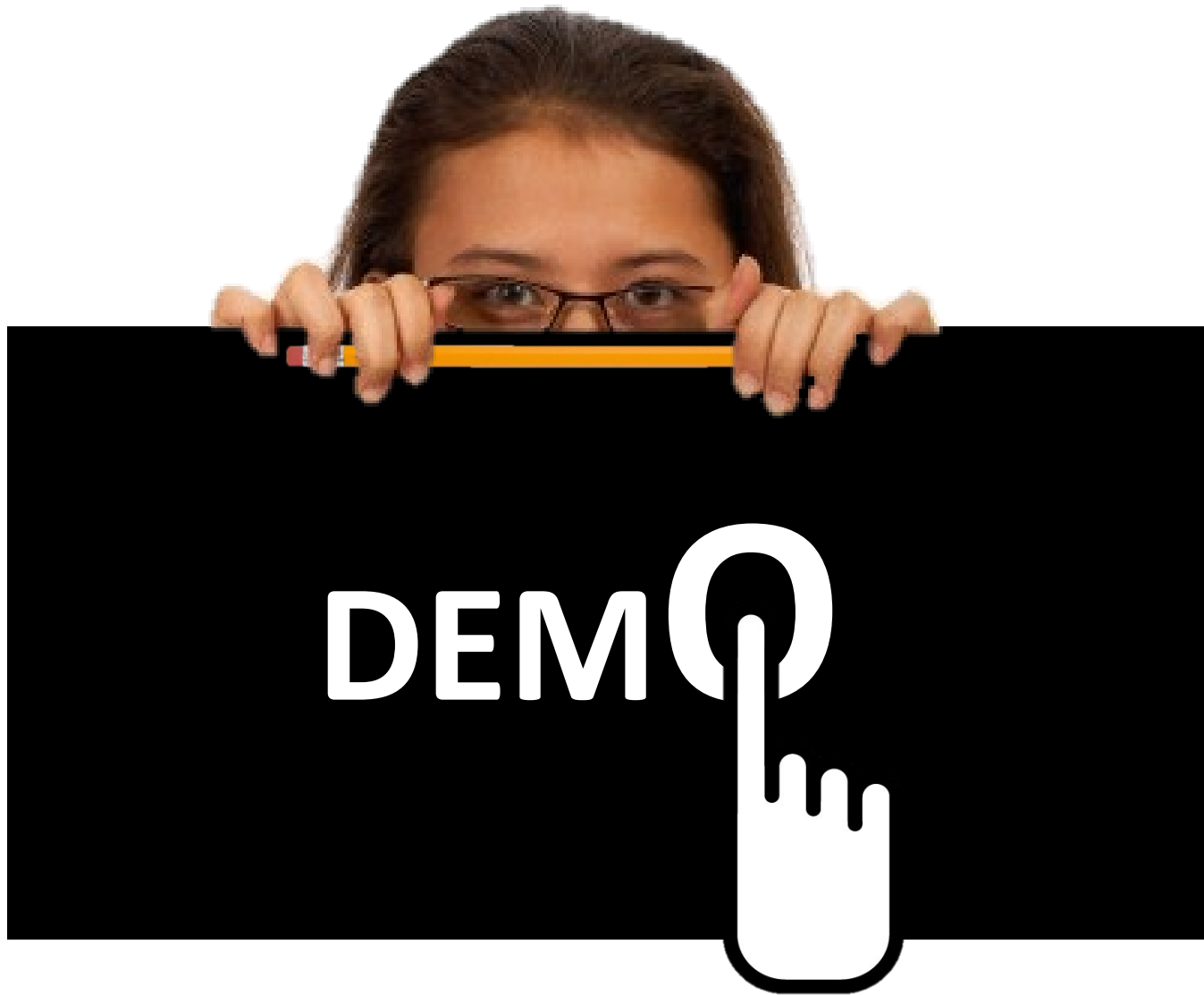
- ❖ `DefaultTableModel model = (DefaultTableModel) tblStaffList.getModel();`

❑ Làm việc với các hàng

- ❖ `model.addRow(rowData);`
- ❖ `model.insertRow(rowIndex, rowData);`
- ❖ `model.removeRow(rowIndex);`
- ❖ `model.setRowCount(rowCount);`

❑ Làm việc với dữ liệu trong ô

- ❖ `model.getValueAt(rowIndex, columnIndex);`
- ❖ `model.setValueAt(value, rowIndex, columnIndex);`



❑ Các thành phần giao diện

- ❖ Container: JFrame, JPanel

- ❖ Control:

- JLabel - Nhãn
- JTextField – Ô nhập
- JButton – Nút nhấn
- JCheckBox – Hộp kiểm
- JRadioButton – Lựa chọn
- JComboBox – Hộp chọn
- JTable – Bảng

❑ Lập trình giao diện

- ❖ Làm việc với thuộc tính của Control
- ❖ Làm việc với JComboBox
- ❖ Làm việc với JTable

