



cep.edu.vn

TRƯỜNG CAO ĐẲNG KINH TẾ - KẾ HOẠCH ĐÀ NẴNG

KHOA TIN HỌC & NGOẠI NGỮ

IFF

SLIDE BÀI GIẢNG

LẬP TRÌNH JAVA 2

THS. NGUYỄN ĐÌNH THÀ

Đà Nẵng – 2022

CHƯƠNG 2. JAVA SWING VÀ JDBC

MỤC TIÊU


Kết thúc bài học này, sinh viên có khả năng:


- Biết cách tạo những ứng dụng có sự kết hợp giữa thư viện Swing và JDBC cho phép người dùng thao tác với ứng dụng thông qua giao diện đồ hoạ (GUI) và quản lý dữ liệu thông qua JDBC
- Sử dụng JTextField và JComboBox để hiển thị dữ liệu từ Database, truyền dữ liệu giữa 2 form
- Sử dụng JTable để hiển thị dữ liệu từ Database
- Có thể viết được những chương trình nhỏ mà phần giao diện được thiết kế bằng cách sử dụng Swing và phần cơ sở dữ liệu sẽ sử dụng JDBC và hệ quản trị cơ sở dữ liệu SQL Server

Giới thiệu

Sản phẩm lập trình Java Swing và Jdbc là những ứng dụng có sự kết hợp giữa thư viện Swing và JDBC cho phép người dùng thao tác với ứng dụng thông qua giao diện đồ họa (GUI) và quản lý dữ liệu thông qua JDBC.

Chuẩn bị CSDL

SCD050718.EMPDB - dbo.emp			
	Column Name	Data Type	Allow Nulls
	uname	varchar(20)	<input type="checkbox"/>
	uemail	varchar(30)	<input checked="" type="checkbox"/>
	upass	varchar(20)	<input checked="" type="checkbox"/>
	ucountry	varchar(20)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

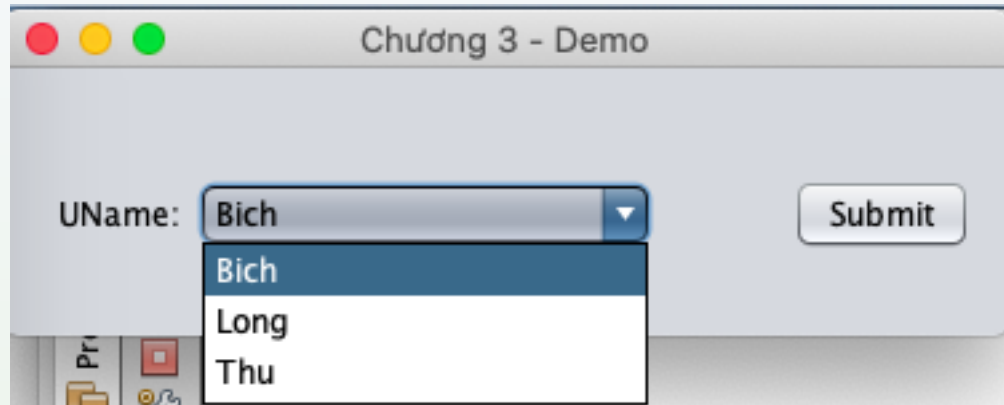
SCD050718.EMPDB - dbo.emp		SCD050718.EMPDB - dbo.emp		
	uname	uemail	upass	ucountry
	Bich	bich@gmail.com	123	HCM
	Hieu	hieu@gmail.com	456	Da Nang
	Ngan	ngan@gmail.com	123	Ha Noi
	Teo	teo@gmail.com	welcome	Tay Nguyen
	Trung	trung@gmail.com	321	Nha Trang
*	NULL	NULL	NULL	NULL

Sử dụng JTextField và JComboBox để hiển thị dữ liệu từ Database

```
//Ghi dữ liệu ra một JTextField
tf1 = new JTextField();
tf1.setEditable(false);
while (rs.next()) {
    ..... tf1.setText(rs.getString(1));
}
```

```
//Ghi dữ liệu ra một JComboBox
JComboBox bx=new JComboBox;
while (rs.next()) {
    ..... String s = rs.getString(1);
    ..... bx.addItem(s);
}
```

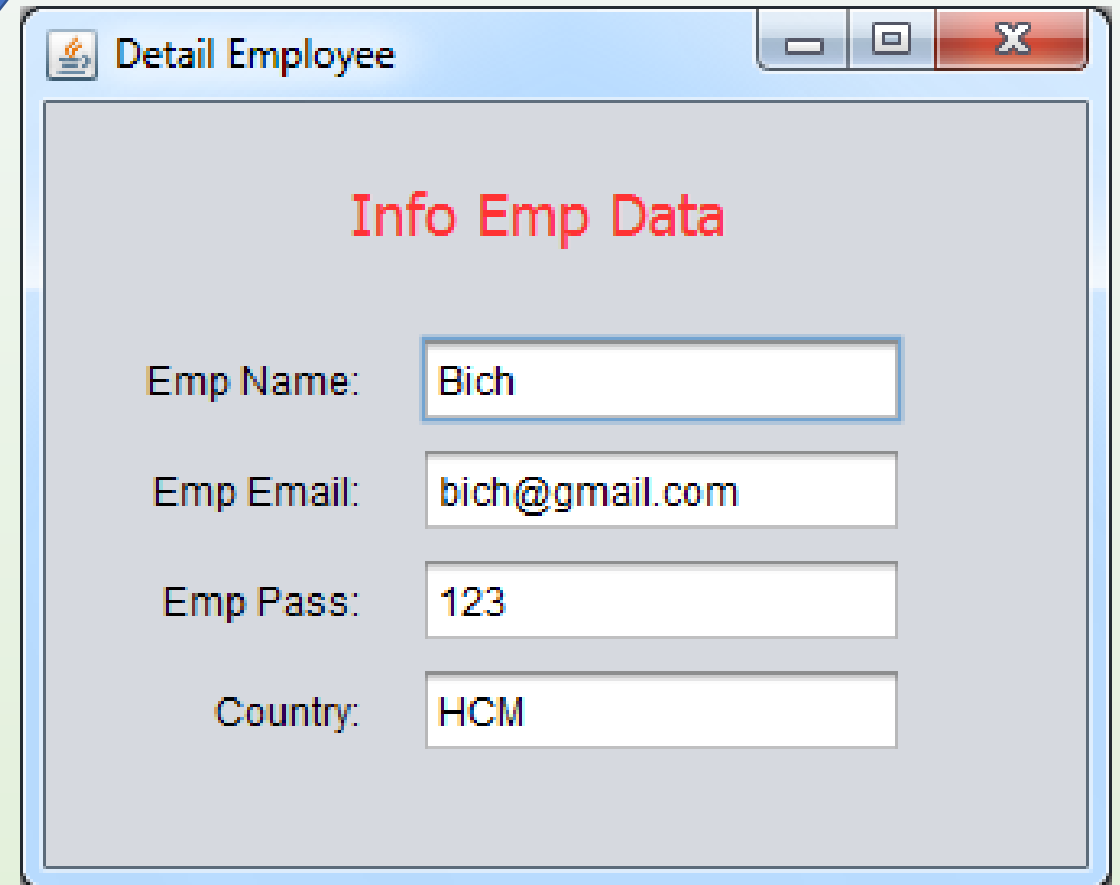
Xây dựng chương trình thực hiện việc đọc CSDL như sau



Yêu cầu:

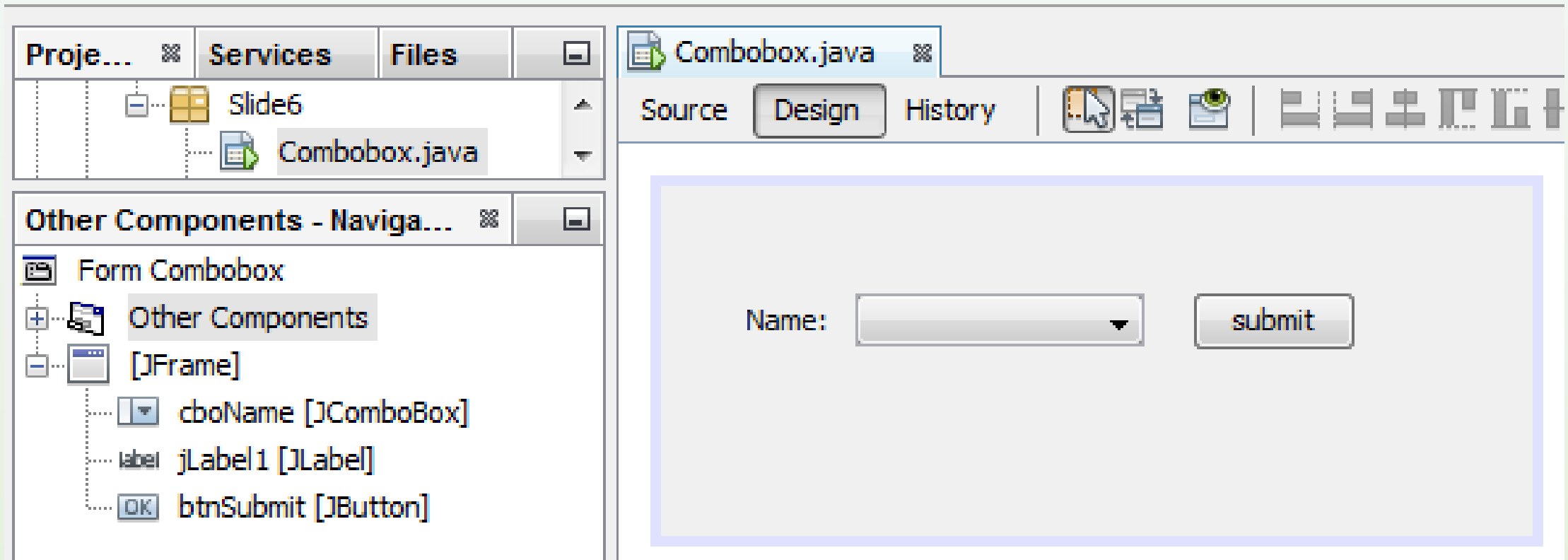
Dữ liệu được load từ Database rồi đưa vào JComboBox

Khi chọn một tên trong danh sách và Click nút Submit, JFrame mới mở lên và hiển thị thông tin chi tiết của UName được chọn bên JComboBox



Xây dựng chương trình thực hiện việc đọc CSDL như sau

Tạo package có 2 class cho chương trình.



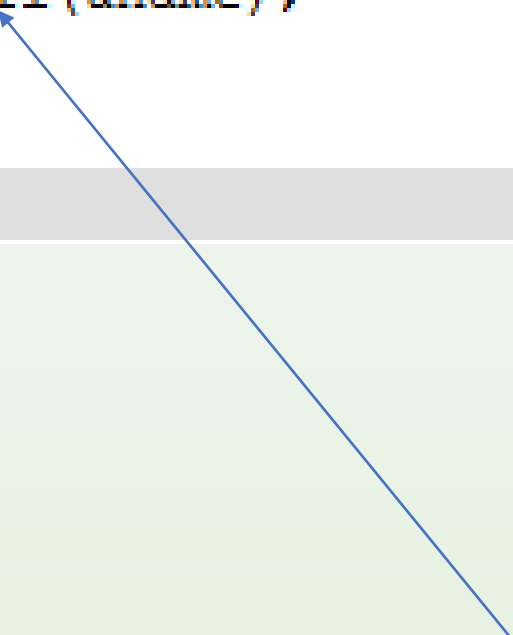
Phương thức **LoadDataToComboBox()**

```
public class Combobox extends javax.swing.JFrame {  
    public Combobox() {  
        initComponents();  
        LoadDataToCombobox();  
    }  
    public void LoadDataToCombobox() { //Load uname vào JComboBox  
        try{  
            String userName="sa";  
            String password="";  
            String url= "jdbc:sqlserver://localhost:1433;databaseName=EMPDB";  
            Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");  
            Connection con = DriverManager.getConnection(url, userName, password);  
            Statement st = con.createStatement();  
            String sql="select uname from emp";  
            ResultSet rs = st.executeQuery(sql);  
            while(rs.next()) {  
                String s = rs.getString(1);  
                cboName.addItem(s);  
            }  
            con.close();  
        } catch (Exception ex) {  
            System.out.println(ex);  
        }  
    }  
    /** .....  
}
```

Phương thức này dùng để đọc *uname* từ bảng **emp** vào **JComboBox**

Sự kiện khi người dùng Click nút Submit

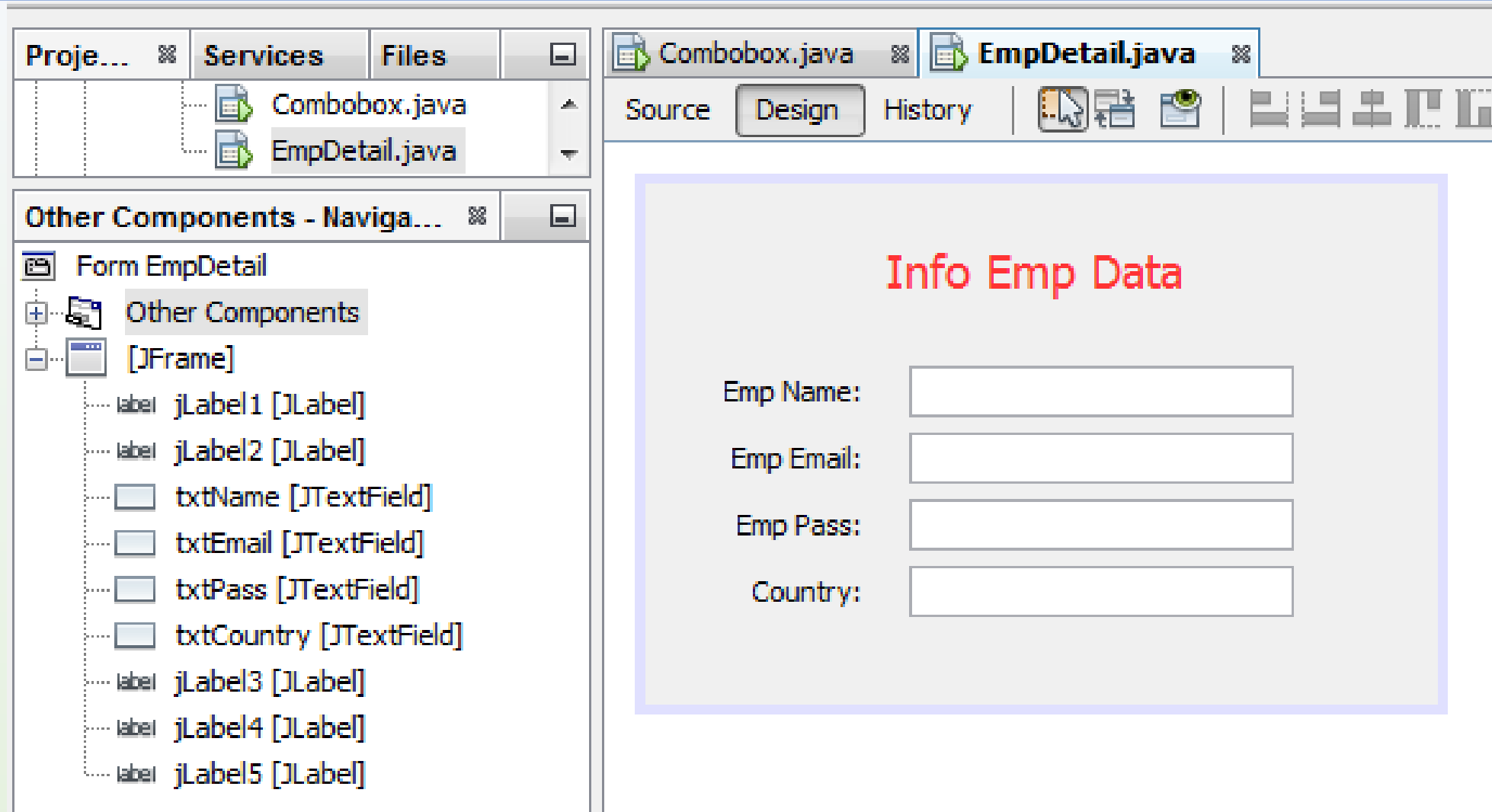
```
private void btnSubmitActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    String uname = (String)cboName.getSelectedItem();  
    EmpDetail emp = new EmpDetail(uname);  
    emp.setVisible(true);  
    //dispose();  
}
```



- 1, Lấy giá trị từ JComboBox
- 2, Truyền giá trị cho JFrame
EmpDetail (chúng ta thực hiện
việc truyền dữ liệu từ form này
qua form kia)

Khởi tạo đối tượng cho class form detail, truyền tham số uname cho hàm constructor, rồi gọi phương thức setVisible để hiển thị form detail

Thiết kế form detail



Form hiển thị thông tin chi tiết của một emp được chọn

Phương thức Display(String name)

```
public class EmpDetail extends javax.swing.JFrame {  
    public EmpDetail(String uname) {  
        initComponents();  
        Display(uname);  
    }  
    public void Display(String uname){  
        try{  
            String userName="sa";  
            String password="";  
            String url= "jdbc:sqlserver://localhost:1433;databaseName=EMPDB";  
            Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");  
            Connection con = DriverManager.getConnection(url, userName, password);  
            PreparedStatement st = con.prepareStatement("select * from emp where uname=?");  
            st.setString(1, uname);  
            ResultSet rs = st.executeQuery();  
            while(rs.next()){  
                txtName.setText(rs.getString(1));  
                txtEmail.setText(rs.getString(2));  
                txtPass.setText(rs.getString(3));  
                txtCountry.setText(rs.getString(4));  
            }  
        }catch(Exception ex){  
            System.out.println(ex);  
        }  
    }  
    /** .....  
}
```

Phương thức display thực hiện lấy thông tin chi tiết từ database hiển thị lên các JTextField dựa trên tham số uname nhận được ở constructor

JTABLE

- JTable là component thuộc swing cho phép hiển thị thông tin ở dạng bảng. Với những thông tin phức tạp, khối lượng thông tin nhiều thì việc sử dụng JTable để thể hiện thông tin là cách thể hiện trực quan và thân thiện nhất.
- Tuy nhiên JTable không chứa dữ liệu trực tiếp, nó chỉ chứa đối tượng TableModel và đối tượng TableModel chứa dữ liệu hiển thị.

JTABLE

- Ứng dụng: Viết chương trình thực hiện việc đọc dữ liệu đổ lên JTable và có các chức năng: New (form nhập mới), Save (lưu mới), Update (cập nhật theo Name), Delete (xoá theo Name). Khi người dùng click vào 1 dòng trên JTable thì hiển thị chi tiết dòng đang chọn lên các Text phía trên (dữ liệu được lấy từ CSDL đã thiết kế ở SQL Server)



Quản Lý Nhân Viên

Quản Lý Nhân Viên

Name: Ngan

Pass: 123

Email: ngan@gmail.com

Country: Ha Noi

New

Save

Update

Delete

Name	Email	Pass	Country
Bich	bich@gmail.com	123	HCM
Hieu	hieu@gmail.com	456	Da Nang
Ngan	ngan@gmail.com	123	Ha Noi
Teo	teo@gmail.com	welcome	Tay Nguyen
Trung	trung@gmail.com	321	Nha Trang

JTABLE - Viết code cho các chức năng

```
public class QLNhanVien extends javax.swing.JFrame {  
    String head[]={"Name","Email","Pass","Country"};  
    DefaultTableModel model = new DefaultTableModel(head, 0);  
    public QLNhanVien() {  
        initComponents();  
        LoadDataToJTable();  
    }  
}
```

Gọi thực hiện hàm LoadDataToJTable để đổ dữ liệu từ Database lên Jtable khi mở form

JTABLE - Viết code cho các chức năng

```
public void LoadDataToJTable() {  
    try{  
        model.setRowCount(0);  
        String user="sa";  
        String pass="";  
        Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");  
        String url="jdbc:sqlserver://localhost:1433;databaseName=EMPDB";  
        Connection con= DriverManager.getConnection(url, user, pass);  
        Statement st = con.createStatement();  
        String sql="select * from emp";  
        ResultSet rs= st.executeQuery(sql);  
        while(rs.next()){  
            Vector row = new Vector();  
            row.add(rs.getString(1));  
            row.add(rs.getString(2));  
            row.add(rs.getString(3));  
            row.add(rs.getString(4));  
            model.addRow(row);  
        }  
        jTable1.setModel(model);con.close();  
    }catch(Exception e){  
        System.out.println(e);  
    }  
}
```

/**

Xây dựng phương thức LoadDataToJTable của lớp QLNhanVien để đổ dữ liệu từ bảng emp của Database lên các dòng của Jtable khi mở form

Có thể khởi tạo mảng String khi addRow cho model

JTABLE - Viết code cho các chức năng

```
private void btnDeleteActionPerformed(java.awt.event.ActionEvent evt) {  
    try{  
        String user="sa";  
        String pass="";  
        Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");  
        String url="jdbc:sqlserver://localhost:1433;databaseName=EMPDB";  
        Connection con = DriverManager.getConnection(url, user, pass);  
        String sql="delete from emp where uname=?";  
        PreparedStatement st = con.prepareStatement(sql);  
        st.setString(1, txtName.getText());  
        st.executeUpdate();  
        con.close();  
        LoadDataToJTable();  
    }catch (Exception e){  
        System.out.println(e);  
    }  
}
```

Xoá dòng theo sql
theo txtName và tải
lại dữ liệu mới từ
databse về Jtable

JTABLE - Viết code cho các chức năng

```
private void btnUpdateActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try{  
        String user="sa";  
        String pass="";  
        Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");  
        String url="jdbc:sqlserver://localhost:1433;databaseName=EMPDB";  
        Connection con = DriverManager.getConnection(url, user, pass);  
        String sql="update emp set umail=?, upass=?, ucountry=? where uname=?";  
        PreparedStatement st = con.prepareStatement(sql);  
        st.setString(1, txtEmail.getText());  
        st.setString(2, txtPass.getText());  
        st.setString(3, txtCountry.getText());  
        st.setString(4, txtName.getText());  
        st.executeUpdate();  
        con.close();  
        LoadDataToJTable();  
    }catch(Exception e){  
        System.out.println(e);  
    }  
}
```

Cập nhật lại dữ liệu theo txtName và tải lại dữ liệu mới từ database về Jtable

JTABLE - Viết code cho các chức năng

```
private void btnSaveActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try{  
        String user="sa";  
        String pass="";  
        Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");  
        String url="jdbc:sqlserver://localhost:1433;databaseName=EMPDB";  
        Connection con = DriverManager.getConnection(url, user, pass);  
        String sql="insert into emp values(?,?,?,?)";  
        PreparedStatement st = con.prepareStatement(sql);  
        st.setString(1, txtName.getText());  
        st.setString(2, txtEmail.getText());  
        st.setString(3, txtPass.getText());  
        st.setString(4, txtCountry.getText());  
        st.executeUpdate();  
        con.close();  
        LoadDataToJTable();  
    }catch(Exception e){  
        System.out.println(e);  
    }  
}
```

Thêm mới 1 dòng dữ liệu vào bảng và tải lại dữ liệu mới từ database về Jtable

JTABLE - Viết code cho các chức năng

```
private void btnNewActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    txtCountry.setText("");  
    txtEmail.setText("");  
    txtName.setText("");  
    txtPass.setText("");  
    txtName.requestFocus();  
}
```

Xoá trắng các mục
nhập để nhập dữ
liệu mới

JTABLE - Viết code cho các chức năng

```
private void jTable1MouseClicked(java.awt.event.MouseEvent evt) {  
    // TODO add your handling code here:  
    int row = jTable1.getSelectedRow();  
    txtName.setText(jTable1.getValueAt(row, 0).toString());  
    txtEmail.setText(jTable1.getValueAt(row, 1).toString());  
    txtPass.setText(jTable1.getValueAt(row, 2).toString());  
    txtCountry.setText(jTable1.getValueAt(row, 3).toString());  
}
```

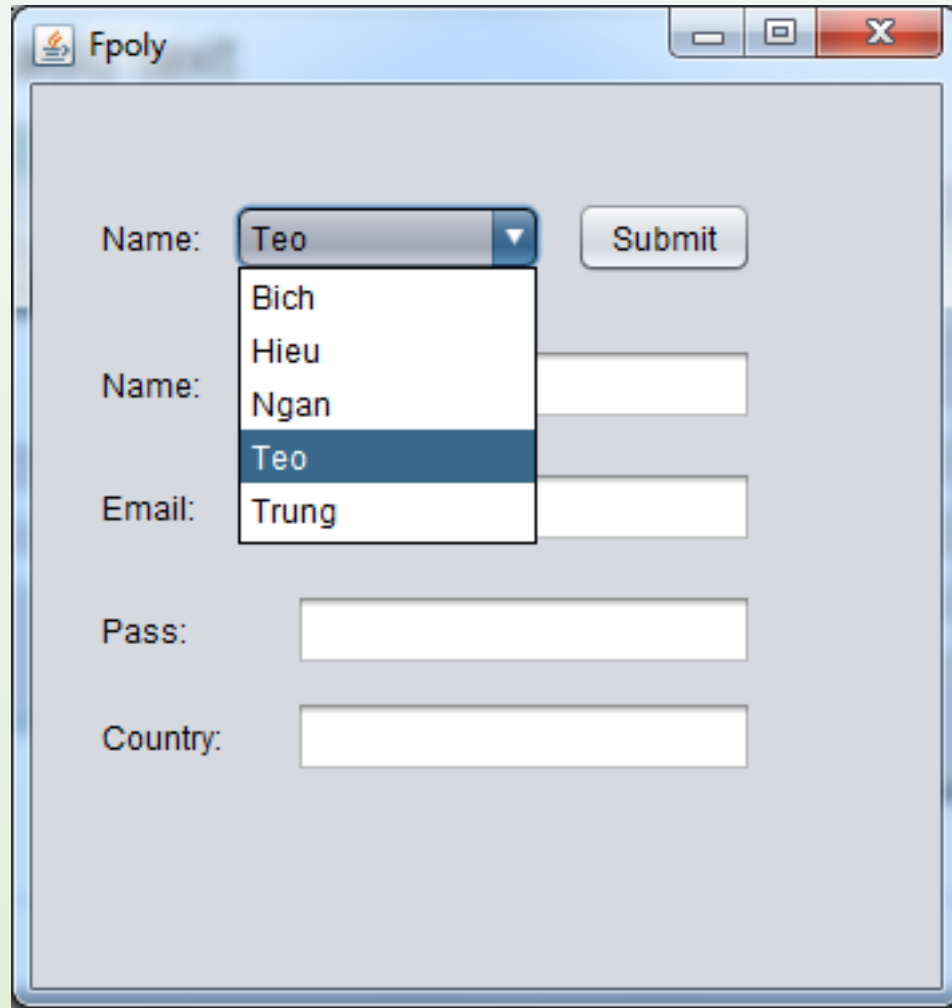
Sự kiện nhấn chuột vào dòng ở Jtable thì dữ liệu dòng đó sẽ được đưa lên các textField ở form.

Sử dụng Netbean để code ứng dụng trên!

Java Swing & JDBC

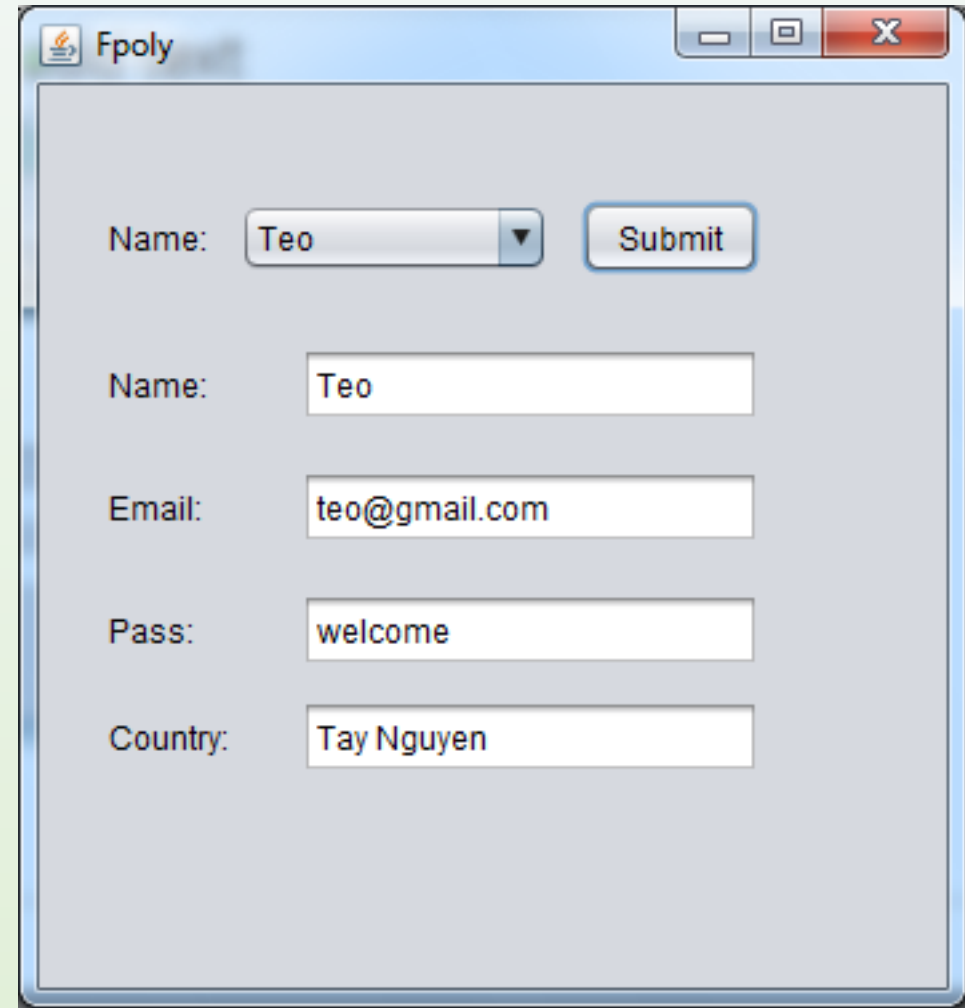
Xây dựng ứng dụng với chức năng:

1. Load dữ liệu từ database rồi đưa vào JComboBox
2. Click nút submit, hiển thị các thông tin chi tiết lên JTextField như form



The screenshot shows a Java Swing window titled "Fpoly". Inside, there is a form with the following elements:

- A label "Name:" followed by a JComboBox. The dropdown menu is open, showing a list of names: Bich, Hieu, Ngan, Teo (highlighted), and Trung.
- A "Submit" button to the right of the first Name field.
- Below the first Name field, there is another "Name:" label followed by an empty JTextField.
- An "Email:" label followed by an empty JTextField.
- A "Pass:" label followed by an empty password field.
- A "Country:" label followed by an empty JTextField.



The screenshot shows the same Java Swing window "Fpoly" after clicking the "Submit" button. The form now displays the following information:

- The first Name field contains "Teo".
- The Email field contains "teo@gmail.com".
- The Pass field contains "welcome".
- The Country field contains "Tay Nguyen".

Code load dữ liệu từ database vào ComboBox

```
// Load du lieu tu database vao ComboBox
public void LoadDataToCombo() {
    try{
        String user="sa";
        String pass="";
        Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
        String url="jdbc:sqlserver://localhost:1433;databaseName=EMPDB";
        Connection con= DriverManager.getConnection(url, user, pass);
        Statement st = con.createStatement();
        String sql="select * from emp";
        ResultSet rs= st.executeQuery(sql);
        while(rs.next()){
            cboName.addItem(rs.getString("uname"));
        }
        con.close();

    }catch(Exception e){
        System.out.println(e);
    }
}
```


Code cho button Submit

```
private void btnSubmitActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try{  
        String user="sa";  
        String pass="";  
        Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");  
        String url="jdbc:sqlserver://localhost:1433;databaseName=EMPDB";  
        Connection con= DriverManager.getConnection(url, user, pass);  
        String sql="select * from emp where uname=?";  
        PreparedStatement st = con.prepareStatement(sql);  
        //Statement st = con.createStatement();  
        //String sql="select * from emp where uname='"+cboName.getSelectedItem()+"'";  
        st.setString(1, cboName.getSelectedItem().toString());  
        ResultSet rs= st.executeQuery();  
        while(rs.next()){  
            txtName.setText(rs.getString(1));  
            txtEmail.setText(rs.getString(2));  
            txtPass.setText(rs.getString(3));  
            txtCountry.setText(rs.getString(4));  
        }  
        con.close();  
  
    }catch(Exception e){  
        System.out.println(e);  
    }  
}
```

Sử dụng Netbean để code ứng dụng trên!

Code lại ứng dụng ở chương 2 Với CSDL tạo ở SQL Server

Bài tập
