



LẬP TRÌNH JAVA 2

BÀI 5: LOAD DỮ LIỆU TỪ DATABASE LÊN JTEXTFIELD, JCOMBOBOX VÀ JTABLE

PHẦN 1

- ◎ Kết thúc bài học này bạn có khả năng
 - ❖ Sử dụng JTextField và JComboBox để hiển thị dữ liệu từ Database, truyền dữ liệu giữa 2 form
 - Ví dụ Demo
 - ❖ Sử dụng JTable để hiển thị dữ liệu từ Database
 - Ví dụ Demo
 - ❖ Bài tập làm thêm



taint.EMPDB - dbo.Employees

	Column Name	Data Type	Allow Nulls
id		int	<input type="checkbox"/>
first		nvarchar(255)	<input checked="" type="checkbox"/>
last		nvarchar(255)	<input checked="" type="checkbox"/>
age		int	<input type="checkbox"/>
			<input type="checkbox"/>

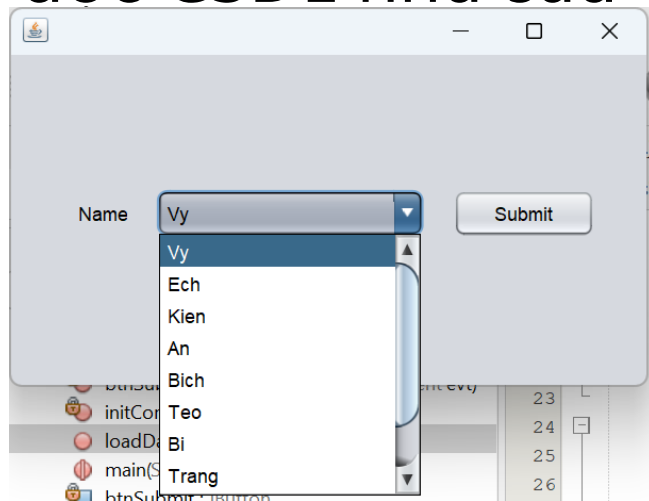
Results		Messages		
	id	first	last	age
1	1	Vy	Nguyen Thi Tuong	35
2	2	Ech	Tu	36
3	3	Kien	Le Phung Hieu	6
4	4	An	Le Phung Hieu	3
5	5	Bich	Le Thi Ngoc	20
6	6	Teo	Le Van	20
7	7	Bi	Nguyen Van	20
8	8	Trang	Le Thi	25
9	9	Toan	Nguyen Van	20
10	10	Gai	Nguyen Thi	21

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

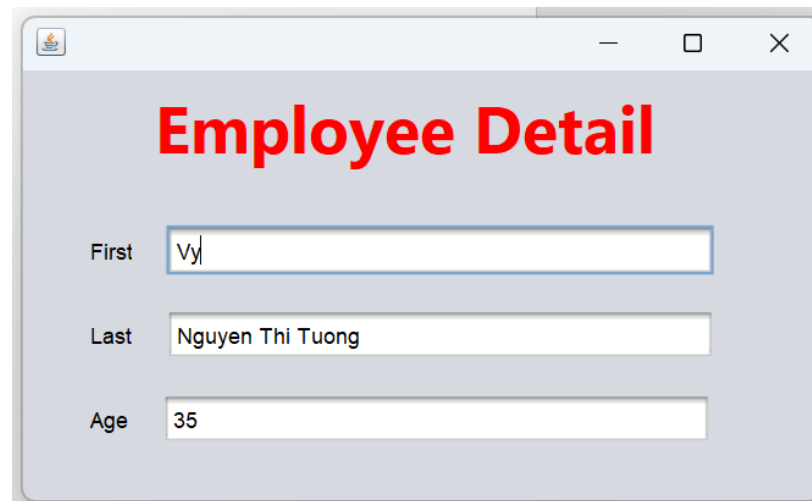
```
//Ghi dữ liệu ra một JTextField  
tf1 = new JTextField();  
tf1.setEditable(false);  
while (rs.next()) {  
    tf1.setText(rs.getString(1));  
}
```

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

❑ Ứng dụng: xây dựng chương trình thực hiện việc đọc CSDL như sau



- Dữ liệu được load từ Database vào JComboBox
- Chọn một tên trong danh sách và Click Submit, JFrame mới mở lên và hiển thị thông tin chi tiết của Name được chọn bên JComboBox



❑ Thiết kế form như hình



The image displays a form design within a light gray rectangular container. On the left, the text "Name" is positioned. To its right is a white rectangular input field with a thin gray border. Inside the input field, on the right side, is a small black downward-pointing chevron icon. Further to the right of the input field is a white rectangular button with a thin gray border, containing the text "Submit" in a black sans-serif font.

- ❑ Phương thức **LoadDataToComboBox()** dùng để đọc first từ bảng employees vào JComboBox

```
public Combobox() {  
    initComponents();  
    loadDataToCombobox();  
}  
  
public void loadDataToCombobox() {  
    try{  
        String userName = "sa";  
        String password = "123";  
        String url = "jdbc:sqlserver://localhost:1433;database=empdb;encrypt=false;";  
        Class.forName(className: "com.microsoft.sqlserver.jdbc.SQLServerDriver");  
        Connection conn = DriverManager.getConnection(url, user: userName, password);  
        Statement st = conn.createStatement();  
        String sql = "select first from employees";  
        ResultSet rs = st.executeQuery(string: sql);  
        while(rs.next()){  
            String s = rs.getString(i: 1);  
            cboName.addItem(item: s);  
        }  
    } catch (Exception ex) {  
        System.out.println(x: ex);  
    }  
}
```

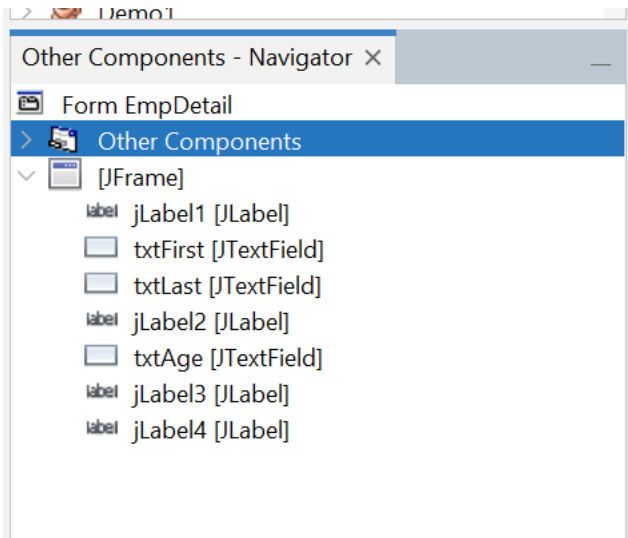
❑ Sự kiện khi người dùng Click nút Submit

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

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)

Thiết kế form



Employee Detail

First

Last

Age

- ❑ Phương thức Display(String name) thực hiện lấy thông tin chi tiết từ database hiển thị lên JTextField dựa trên tham số name.

```
public EmpDetail(String first) {
    initComponents();
    Display(first);
}

public void Display(String first){
    try{
        String userName = "sa";
        String password = "123";
        String url = "jdbc:sqlserver://localhost:1433;database=empdb;encrypt=false;";
        Class.forName(className: "com.microsoft.sqlserver.jdbc.SQLServerDriver");
        Connection conn = DriverManager.getConnection(url, user:userName, password);

        String sql = "select * from employees where first = ?";
        PreparedStatement st = conn.prepareStatement(string: sql);
        st.setString(i: 1, string: first);
        ResultSet rs = st.executeQuery();
        while(rs.next()){
            txtFirst.setText(t: rs.getString(string: "first"));
            txtLast.setText(t: rs.getString(string: "last"));
            txtAge.setText(t: rs.getString(string: "age"));
        }
    } catch (Exception ex) {
        System.out.println(x: ex);
    }
}
```



DEMO

Chạy và giải thích





LẬP TRÌNH JAVA 2

BÀI 5: LOAD DỮ LIỆU TỪ DATABASE LÊN JTEXTFIELD, JCOMBOBOX VÀ JTABLE

PHẦN 2

- ❑ 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.

- ❑ Ứng dụng: Viết chương trình thực hiện việc đọc dữ liệu đổ lên JTable và có các chức năng thêm, Xóa, Sửa, Update. Khi người dùng click vào 1 dòng trên JTable thì hiển thị chi tiết dòng đang chọn lên các Text phía trên

Quản lý nhân viên

QUẢN LÝ NHÂN VIÊN

ID:

Name:

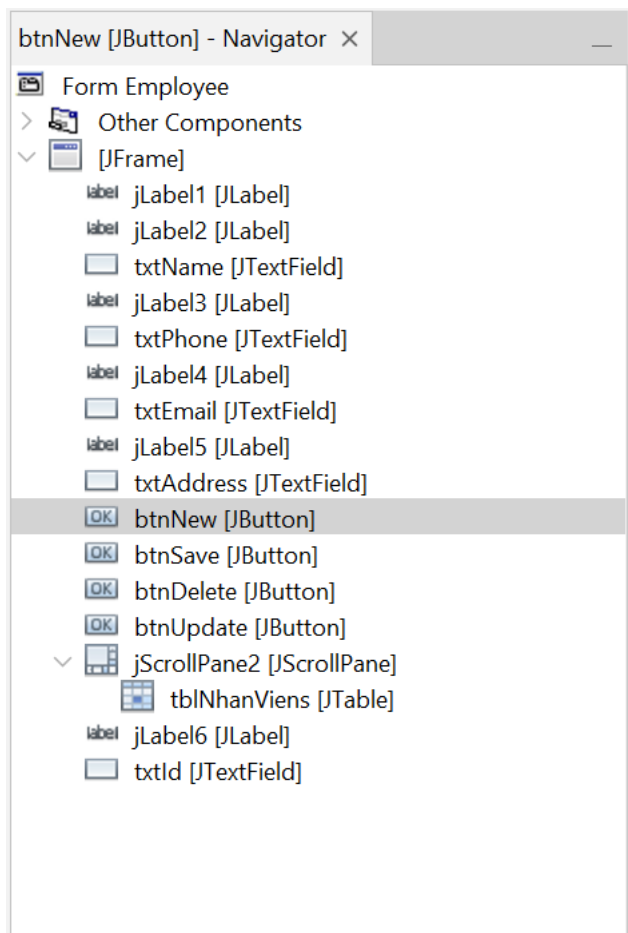
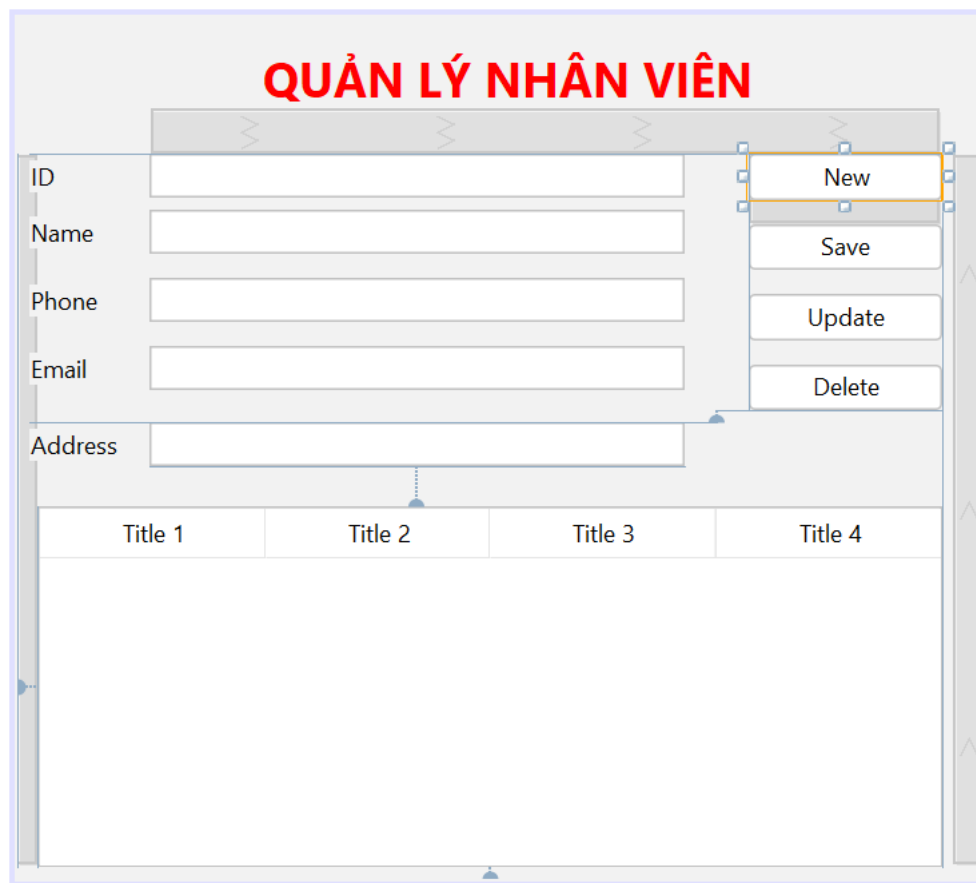
Pho...:

Email:

Address:

ID	NAME	PHONE	EMAIL	ADDRESS
1	Tam	0919222333	tam@fpt.edu...	Cần Thơ
2	Hieu	0919123456	hieu@fpt.ed...	Cần Thơ
3	Ngan	0919123457	ngan@fpt.ed...	Vinh Long
4	Trung	0919123458	Trung@fpt.e...	Hậu Giang

Thiết kế form như hình

QUẢN LÝ NHÂN VIÊN

ID:

Name:

Phone:

Email:

Address:

New
 Save
 Update
 Delete

Title 1	Title 2	Title 3	Title 4

Viết code cho các chức năng

```

public void loadDataToTable() {
    String[] header = {"ID", "NAME", "PHONE", "EMAIL", "ADDRESS"};
    DefaultTableModel model = new DefaultTableModel(columnNames:header, rowCount:0);
    try{
        String userName = "sa";
        String password = "123";
        String url = "jdbc:sqlserver://localhost:1433;database=empdb;encrypt=false;";
        Class.forName(className: "com.microsoft.sqlserver.jdbc.SQLServerDriver");
        Connection conn = DriverManager.getConnection(url, user: userName, password);

        String sql = "select * from emp";
        Statement st = conn.createStatement();
        ResultSet rs = st.executeQuery(string: sql);
        while(rs.next()){
            model.addRow(new Object[]{rs.getInt(string: "id"), rs.getString(string: "name"),
                rs.getString(string: "phone"), rs.getString(string: "email"), rs.getString(string: "address")});
        }
        tblNhanViens.setModel(dataModel: model);
    }catch(Exception ex){
        System.out.println(x: ex);
    }
}

```

Đọc tất cả dữ liệu từ bảng
 “emp” đổ lên JTable khi
 form mở

❑ Viết code cho các chức năng

```
private void btnDeleteActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try{  
        String userName = "sa";  
        String password = "123";  
        String url = "jdbc:sqlserver://localhost:1433;database=empdb;encrypt=false;";  
        Class.forName(className: "com.microsoft.sqlserver.jdbc.SQLServerDriver");  
        Connection conn = DriverManager.getConnection(url, user: userName, password);  
        String sql = "delete from emp where id = ?";  
        PreparedStatement st = conn.prepareStatement(string: sql);  
        st.setInt(i: 1, i1: Integer.parseInt(s: txtId.getText()));  
        st.executeUpdate();  
        conn.close();  
        loadDataToTable();  
    } catch (Exception ex) {  
        System.out.println(x: ex);  
    }  
}
```

Thực hiện xóa dòng dữ liệu dưới bảng
“emp” dựa trên txtId

❑ Viết code cho các chức năng

```
private void btnUpdateActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try{  
        String userName = "sa";  
        String password = "123";  
        String url = "jdbc:sqlserver://localhost:1433;database=empdb;encrypt=false;";  
        Class.forName(className: "com.microsoft.sqlserver.jdbc.SQLServerDriver");  
        Connection conn = DriverManager.getConnection(url, user:userName, password);  
        String sql = "update emp set name=?,phone=?,email=?,address=? where id = ?";  
        PreparedStatement st = conn.prepareStatement(string: sql);  
        st.setString(i: 1, string: txtName.getText());  
        st.setString(i: 2, string: txtPhone.getText());  
        st.setString(i: 3, string: txtEmail.getText());  
        st.setString(i: 4, string: txtAddress.getText());  
        st.setInt(i: 5, i1: Integer.parseInt(s: txtId.getText()));  
        st.executeUpdate();  
        conn.close();  
        loadDataToTable();  
    }catch(Exception ex){  
        System.out.println(x: ex);  
    }  
}
```

Update dữ liệu vào bảng “emp” theo các tham số

Viết code cho các chức năng

```

private void btnSaveActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try{
        String userName = "sa";
        String password = "123";
        String url = "jdbc:sqlserver://localhost:1433;database=empdb;encrypt=false;";
        Class.forName(className: "com.microsoft.sqlserver.jdbc.SQLServerDriver");
        Connection conn = DriverManager.getConnection(url, user:userName, password);

        String sql = "insert into emp values(?,?,?,?,?)";
        PreparedStatement st = conn.prepareStatement(string: sql);
        st.setInt(i: 1, i1: Integer.parseInt(txtName.getText()));
        st.setString(i: 2, string: txtPhone.getText());
        st.setString(i: 3, string: txtAddress.getText());
        st.setString(i: 4, string: txtEmail.getText());
        st.setString(i: 5, string: txtAddress.getText());
        st.executeUpdate();
        conn.close();
        loadDataToTable();
    }catch(Exception ex){
        System.out.println(x: ex);
    }
}
  
```

Lưu dữ liệu vào bảng “emp” lấy từ các JTextField

❑ Viết code cho các chức năng

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

Clear tất cả dữ liệu trên các JTextField cho người dùng nhập mới

❑ Viết code cho các chức năng

```
private void tblNhanViensMouseClicked(java.awt.event.MouseEvent evt) {  
    // TODO add your handling code here:  
    int row = tblNhanViens.getSelectedRow();  
    txtId.setText(t: tblNhanViens.getValueAt(row, column: 0).toString());  
    txtName.setText(t: tblNhanViens.getValueAt(row, column: 1).toString());  
    txtPhone.setText(t: tblNhanViens.getValueAt(row, column: 2).toString());  
    txtEmail.setText(t: tblNhanViens.getValueAt(row, column: 3).toString());  
    txtAddress.setText(t: tblNhanViens.getValueAt(row, column: 4).toString());  
}
```

Sự kiện khi người dùng Click mouse lên các dòng trên Jtable, đọc dữ liệu chi tiết trên dòng đang chọn đổ lên các JTextField



DEMO

Chạy và giải thích



- ❖ Sử dụng JTextField và JComboBox để hiển thị dữ liệu từ Database, truyền dữ liệu giữa hai form
 - Ví dụ Demo
- ❖ Sử dụng JTable để hiển thị dữ liệu từ Database
 - Ví dụ Demo
- ❖ Bài tập làm thêm





Cảm ơn