

Bài 6:

JDBC



Mục tiêu bài học

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

Chuẩn bị CSDL

■ emp.sql

- create table emp

- (
 uname varchar(20), uemail varchar(30), upass varchar(20), ucountry
 varchar(20)

-);

■ Đưa dữ liệu mẫu vào

- INSERT INTO emp VALUES ('sandeep',
 'sandy05.1991@gmail.com', 'welcome', 'India');

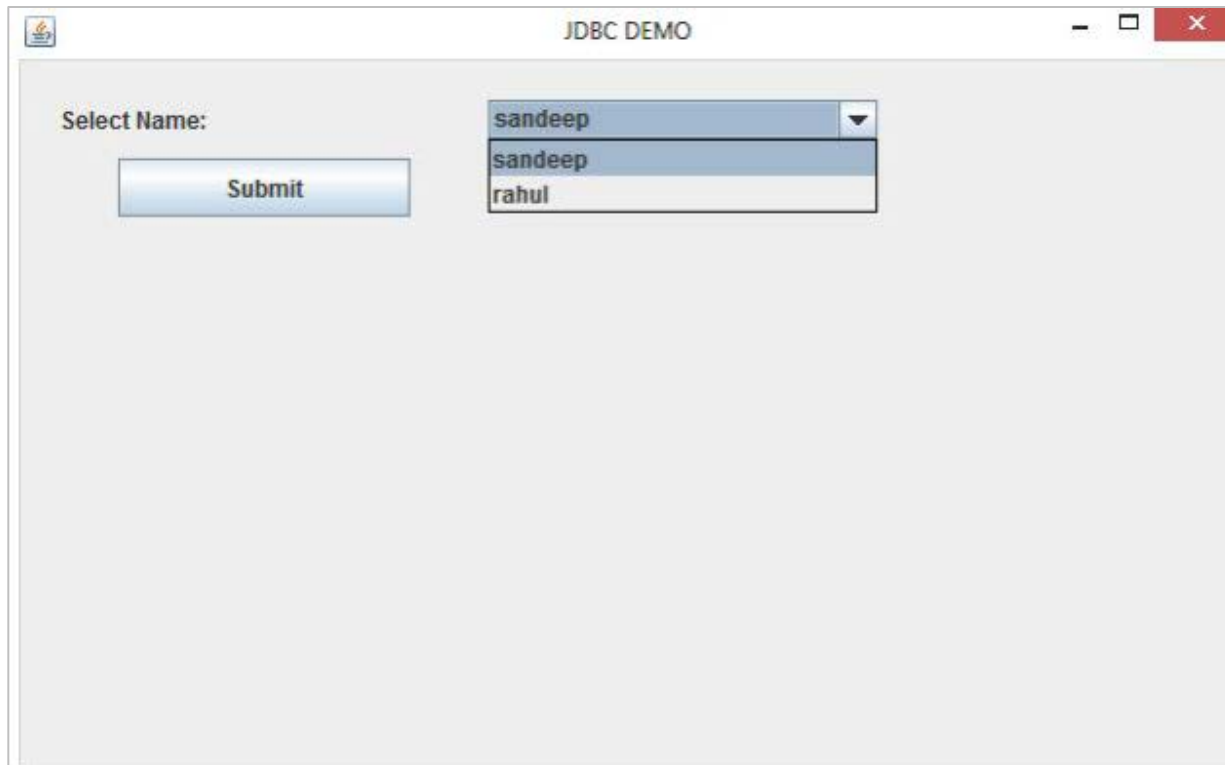
- INSERT INTO emp VALUES ('rahul', 'rahul@gmail.com' ,
 '123', 'India');

```
//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);
}
```

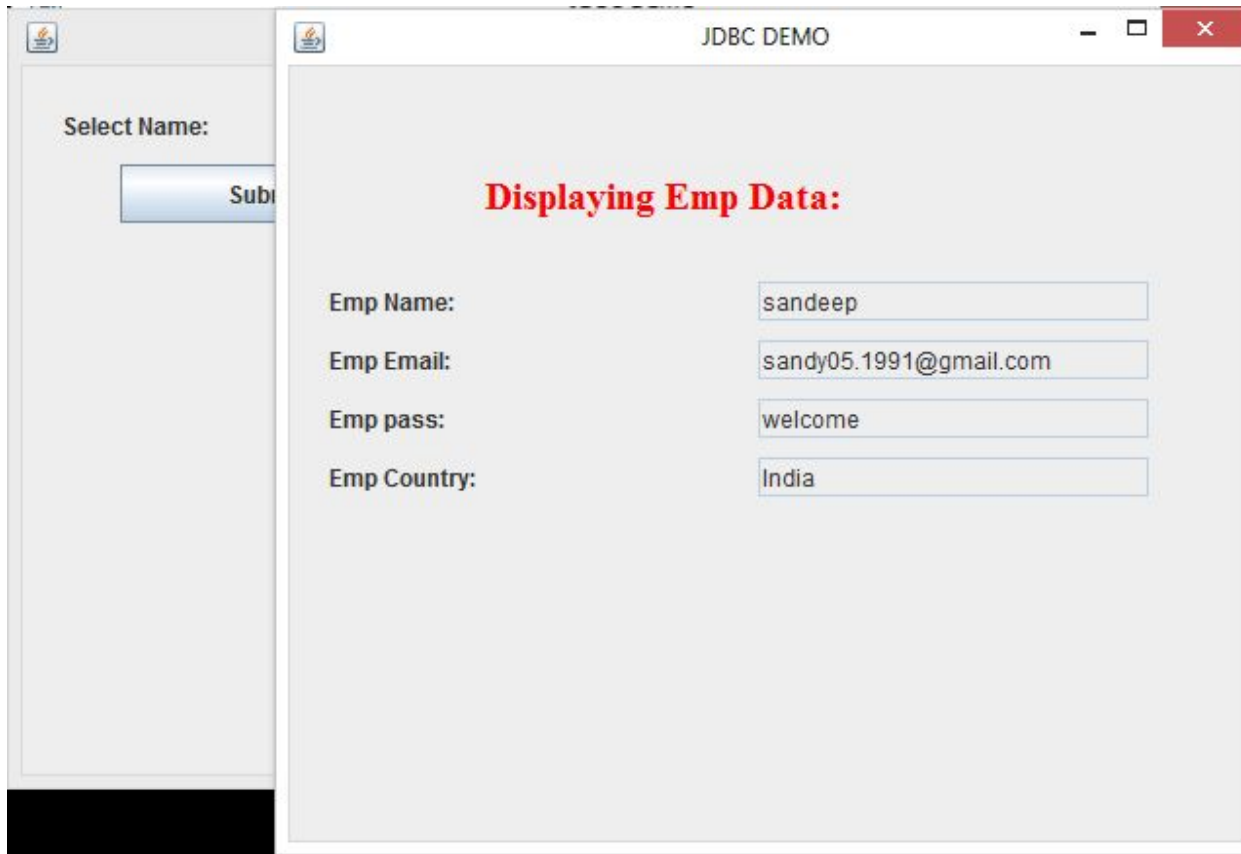
JTextField và JComboBox

- Ứng dụng: xây dựng 1 chương trình Java thực hiện việc đọc CSDL như sau



JTextField và JComboBox

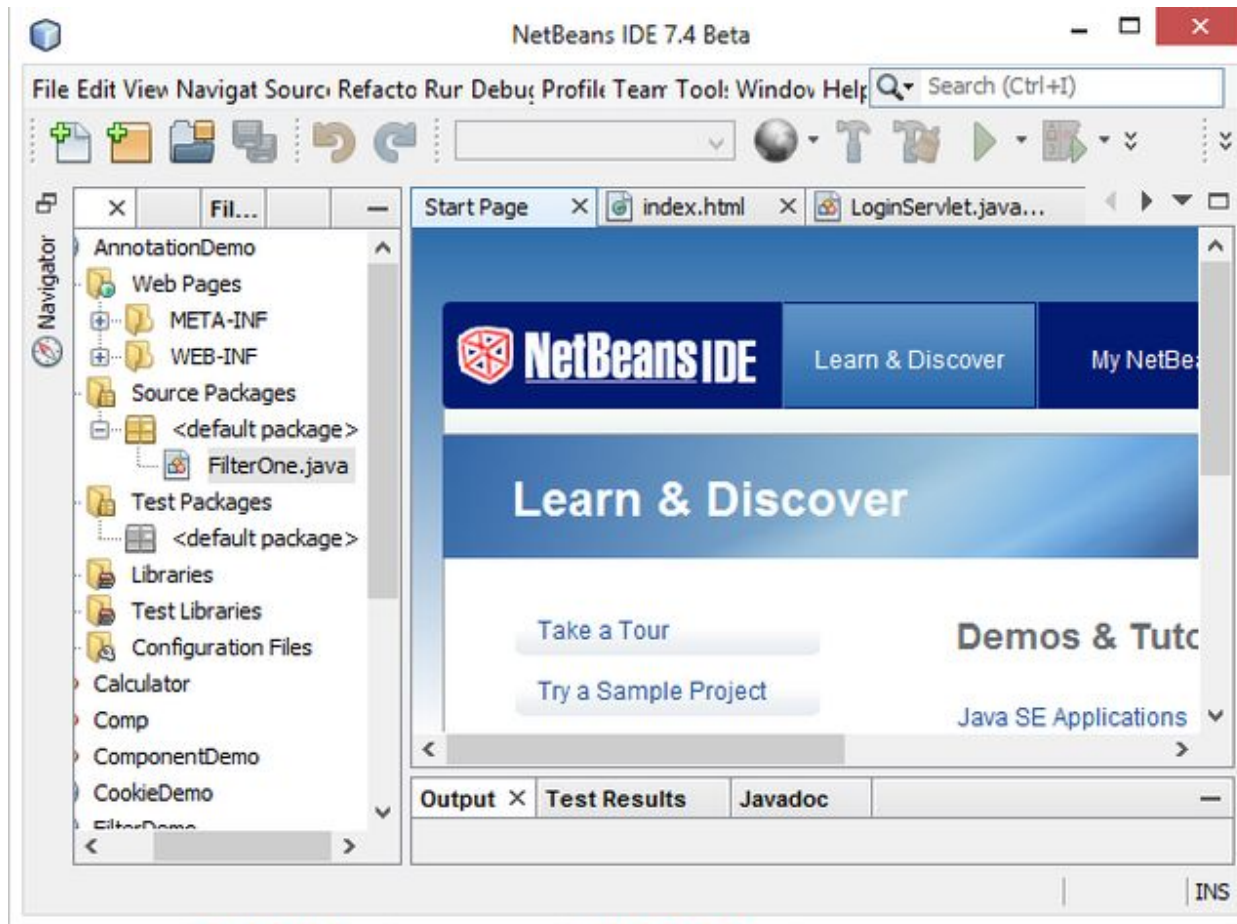
- Chon một tên trong danh sách và bấm Submit



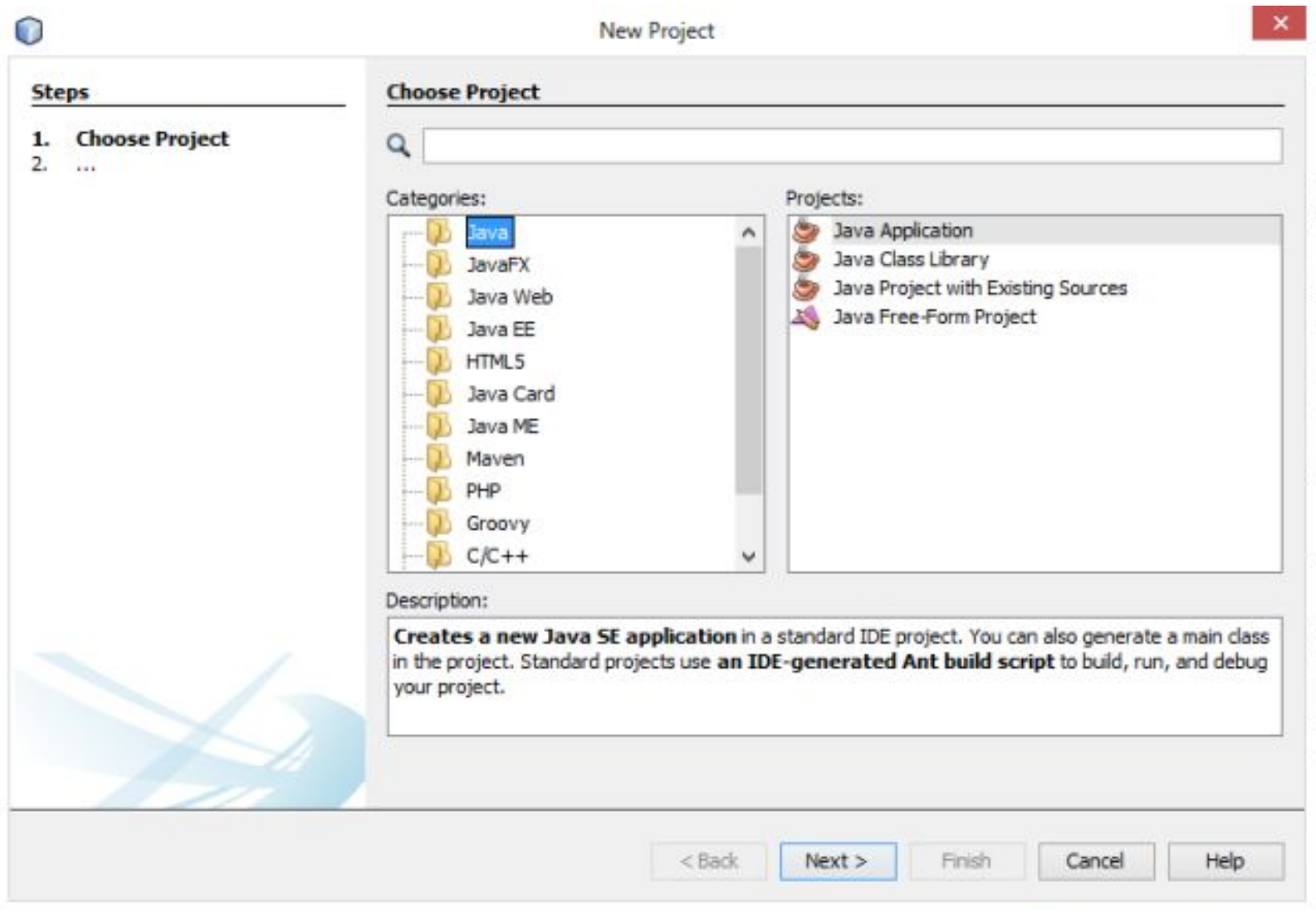
The screenshot shows a Java Swing window titled "JDBC DEMO". On the left side, there is a vertical panel with the text "Select Name:" and a "Submit" button. The main area of the window displays the text "Displaying Emp Data:" in red. Below this text, there are four input fields with labels: "Emp Name:" containing "sandeep", "Emp Email:" containing "sandy05.1991@gmail.com", "Emp pass:" containing "welcome", and "Emp Country:" containing "India".

JTextField và JComboBox

- Thực hiện: Tạo mới một Project Java



JTextField và JComboBox



JTextField và JComboBox

Đặt tên Project là EmpSearchApp

The image shows a 'New Java Application' dialog box with the following fields and options:

- Steps:**
 1. Choose Project
 2. **Name and Location**
- Name and Location:**
 - Project Name: EmpSearchApp
 - Project Location: \\MCSERVER2\UserProfiles\ssharna\My Doc...
 - Project Folder: !\UserProfiles\ssharna\My Documents\NetBea...
 - Use Dedicated Folder for Storing Libraries
 - Libraries Folder: .\lib
 - Different users and projects can share the same compilation libraries (see Help for details).
 - Create Main Class: EmpSearchApp
- Buttons:** < Back, Next >, **Finish**, Cancel, Help

JTextField và JComboBox

Xây dựng giao diện chương trình

```
public class EmpSearchApp extends JFrame implements ActionListener {

    JLabel l, l1, l2, l3, l4, l5;
    JButton b;
    JTextField tf1, tf2, tf3, tf4;
    JComboBox bx;
    String str;

    EmpSearchApp() {
        setVisible(true);
        setSize(700, 700);
        setLayout(null);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setTitle("JDBC DEMO");

        l = new JLabel("Select Name:");
        b = new JButton("Submit");
        bx = new JComboBox();

        tf1 = new JTextField();
        tf2 = new JTextField();
        tf3 = new JTextField();
        tf4 = new JTextField();

        l.setBounds(20, 20, 200, 20);
        b.setBounds(50, 50, 150, 30);
        bx.setBounds(240, 20, 200, 20);

        add(l);
        add(b);
        add(bx);

        tf1.setEditable(false);
        tf2.setEditable(false);
        tf3.setEditable(false);
        tf4.setEditable(false);
    }
}
```


JTextField và JComboBox

- Kết nối CSDL lấy danh sách nhân sự vào JComboBox

```
try {
    String userName="sa";
    String password="sapassword";
    String url="jdbc:sqlserver://localhost:1433;databaseName:emp";
    Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
    Connection con = DriverManager.getConnection("url,userName,password");
    PreparedStatement ps = con.prepareStatement("SELECT uname FROM emp");
    ResultSet rs = ps.executeQuery();
    while (rs.next()) {
        String s = rs.getString(1);
        bx.addItem(s);
    }
} catch (Exception ex) {
    System.out.println(ex);
}
}
```

JTextField và JComboBox

- Xử lý sự kiện nhấn nút, sẽ mở ra 1 cửa sổ mới



```
public void actionPerformed(ActionEvent e) {
    showData();
}

public void showData() {
    JFrame f1 = new JFrame();
    f1.setVisible(true);
    f1.setSize(500, 500);
    f1.setTitle("JDBC DEMO");

    l5 = new JLabel("Displaying Emp Data:");
    l5.setForeground(Color.red);
    l5.setFont(new Font("Serif", Font.BOLD, 20));
    l1 = new JLabel("Emp Name:");
    l2 = new JLabel("Emp Email:");
    l3 = new JLabel("Emp pass:");
    l4 = new JLabel("Emp Country:");
}
```

JTextField và JComboBox

- Xử lý sự kiện nhấn nút, sẽ mở ra 1 cửa sổ mới

```
l5.setBounds(100, 50, 300, 30);
l1.setBounds(20, 110, 200, 20);
l2.setBounds(20, 140, 200, 20);
l3.setBounds(20, 170, 200, 20);
l4.setBounds(20, 200, 200, 20);

tf1.setBounds(240, 110, 200, 20);
tf2.setBounds(240, 140, 200, 20);
tf3.setBounds(240, 170, 200, 20);
tf4.setBounds(240, 200, 200, 20);

f1.add(l5);
f1.add(l1);
f1.add(tf1);
f1.add(l2);
f1.add(tf2);
f1.add(l3);
f1.add(tf3);
f1.add(l4);
f1.add(tf4);
```

JTextField và JComboBox

- Truy vấn và hiển thị thông tin nhân viên đã chọn

```
str = (String) bx.getSelectedItemAt(); //chuyển đổi kiểu dữ liệu
try {
    String userName="sa";
    String password="sapassword";
    String url="jdbc:sqlserver://localhost:1433;databaseName=emp";
    Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
    Connection con = DriverManager.getConnection(url,userName,password);
    PreparedStatement ps = con.prepareStatement("SELECT * FROM emp WHERE uname=?");
    ps.setString(1, str); //Cung cấp giá trị cho prepareStatement tại vị trí dấu ?
    ResultSet rs = ps.executeQuery();
    while (rs.next()) {

        tf1.setText(rs.getString(1));
        tf2.setText(rs.getString(2));
        tf3.setText(rs.getString(3));
        tf4.setText(rs.getString(4));

    }
} catch (Exception ex) {
    System.out.println(ex);
}

}

public static void main(String arr[]) {
    new EmpSearchApp();
}
}
```

JDBC – 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.

JDBC – JTable

- Để hiển thị bảng **emp** ở trên, ta làm như sau

```
//tạo vector chứa tên các cột của Jtable
    Vector rowHeader = new Vector ();
    rowHeader.add ("Emp Name");
    rowHeader.add ("Emp Email");
    rowHeader.add ("Emp pass");
    rowHeader.add ("Emp Country");
//tạo TableModel chứa dữ liệu cho Jtable
    DefaultTableModel model = new DefaultTableModel(rowHeader,0);
```


JDBC – JTable

```
//Kết nối CSDL
try {
    String userName="sa";
    String password="sapassword";
    String url="jdbc:sqlserver://localhost:1433;databaseName:emp";
    Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
    Connection con = DriverManager.getConnection("url,userName,password");
    PreparedStatement ps = con.prepareStatement("SELECT * FROM emp");
    ResultSet rs = ps.executeQuery();

//Lặp để đọc dữ liệu
    Vector rowData;
    while (rs.next()){
        rowData = new Vector() ;
        rowData.add (rs.getString("uname"));
        rowData.add (rs.getString("umail"));
        rowData.add (rs.getString("upass"));
        rowData.add (rs.getString("ucountry"));
        model.addRow(rowData) ;
    }
}
```

JDBC – JTable

```
://hiển thịTableModel lên JTable (hiển thị dữ liệu lên bảng)  
jTable1.setModel(model);
```

JDBC – JTable

```
.....  
//Đếm số hàng
```

```
public int getRowCount() {  
    return rowData.size();  
}
```

```
.....  
//Đếm số cột
```

```
public int getColumnCount() {  
    return rowHeader.size();  
}
```

JDBC – JTable

```
//Đọc giá trị tại một ô
public Object getValueAt(int row, int column) {
    Vector tbData = (Vector) (rowData.elementAt(row));
    return tbData.elementAt(column);
}

//Không cho sửa trực tiếp
public boolean isCellEditable(int rowIndex, int columnIndex) {
return false;
}
```

XIN CẢM ƠN!

