Tutorial on Building a web Application with Jdeveloper using EJB, JPA and Java Server Faces

By Phaninder Surapaneni

This Tutorial covers:
1. Building the DataModel using EJB3.0.
2. Creating a MasterDetail JSF Page.
4. Customizing the layout.

In this tutorial model is provided by EJB components, view and controller are provided by JavaServer Faces. We will be building the user interface using ADF Faces set of JSF compatible component.

Establishing DataBase Connection

Step1: Open Jdeveloper, click on the connections tab and right click on the Database and click on the New Database Connection.

Step2: Enter the username and password and then enter the Host Name = oracle.cis.ksu.edu
And SID = ORACLE and click on next.
After the connection is established, test whether the connection works or not and click on finish.

**Building a Data Model with EJB 3.0**

Step 1: Click on the Applications tab and right click on the applications and select New Application. Then
a create Application dialog box will appear.

Enter the Application name as Ejbapp and enter oracle as Application Package info and select Web Application [JSF, EJB]. Now click on Manage Templates.

Step 2: In the Application templates select the view and controller in web Application [JSF, EJB] and in the view and controller pane enter the project name to UserInterface.jpr.

Step 3: In the Application templates select the Data Model in web Application [JSF, EJB] and in the Data Model Pane enter the project name to EJBModel.jpr. Then click on ok.
Creating the Persistence Model

Step 1: In the Application Navigator, right click on the EJBModel node and select new

Step 2: In the New Gallery dialog box, in the Business Tier, Click on the EJB node then, select Entities from Tables (JPA/EJB 3.0) and click on next.

Step 3: Create Entities from Tables wizard, select sphani as the connection name. click on next
Step 4: In “sphani” Schema and the Tables Objects Types, click the query button. Then select the Department and Employee tables from the Available list and add to selected list. Then click on next and finish.

Step 5: Right click on the Ejbmodel node and click on make
Make sure that there are no errors present in the Errors Messages tab at the bottom.

**Creating the Data Model**

Step 1: Right-click the EJBModel project node in the Applications Navigator, and then click on the New node.
Step 2: Select the Session Bean in the EJB category and click on next, make sure that you enter HRFacade as the EJB name in the next window.

Click next and click on finish.

Step 3: Right-click the HRFacadeBean node in the Applications Navigator and select Create Data Controls from context
Step 4: In the Choose EJB Interface dialog, select local, and click ok.
Step 5: Right-click the EJBModel project and select the Make option to compile your project. Make sure that Messages - Log window does not report any error.

**Creating the Page Flow**

Step 1: In the Applications Navigator, right-click the UserInterface node and select Open JSF Navigation.
We will be getting a empty diagram after selecting the JSF Navigation

Step 2: In the JSF Navigation Diagram page of the Component Palette, select JSF Page drop it in the diagram. Rename the page deptEmployee.
Step 3: Now drag and drop a JSF Page in the diagram next to deptEmployee and rename it as editEmployee.

Step 4: By selecting the JSF Navigation from the component palette connect the deptEmployee and editEmployee.

Rename the label ‘success’ as ‘edit’
Step 5: Now select the JSF navigation and select the editEmployee and select the deptEmployee.

Save the diagram.

Creating a Master-Details Java Server Faces Page.

Step 1: In the page flow diagram double click on the deptEmployees icon to launch create JSF wizard.

Step 2: Click on the next button to skip the first page and then select the JSP Document (*.jspx) button.

Step 3: In the next page select the Automatically Expose UI Components option
Step 4: In the next window select the available libraries JSF Core 1.0, JSF HTML 1.0, ADF Faces Components, ADF Faces

Click on next and finish to create new JSF page.
Step 5: Now from the component palette in ADF Faces Core library, drag and drop ADF Faces **panelPage** to the empty page.

After dropping, change the title for the panelPage to Department/Employee.

Step 6: Open the Data Control palette, expand the HRFacadeLocal | queryDepartmentsFindAll() | Departments and select the employeeList node.

Now drag and drop the department collection on the page and it appears as follows.
Step 7: Select the Submit button and in the Property pane, change the Text value to edit. Use the dropdown list for the Action and select edit.

Step 8: Open the page diagram again and right click on the deptEmployees.jspx and click on run
The Master Detail layout opens in the browser as follows

Department/Employee

Department

Department Employee List

<table>
<thead>
<tr>
<th>Select name</th>
<th>employee</th>
<th>firstname</th>
<th>lastname</th>
<th>phone type</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="mailto:Missy@acme-hospital.com">Missy@acme-hospital.com</a></td>
<td>101</td>
<td>Missy</td>
<td>Jewel</td>
<td>1001 Registered Nurse</td>
</tr>
<tr>
<td><a href="mailto:Tara@acme-hospital.com">Tara@acme-hospital.com</a></td>
<td>102</td>
<td>Tara</td>
<td>Monss</td>
<td>1002 Registered Nurse</td>
</tr>
<tr>
<td><a href="mailto:luc@acme-hospital.com">luc@acme-hospital.com</a></td>
<td>103</td>
<td>lucy</td>
<td>Boop</td>
<td>1003 Cardiovascular Technologist</td>
</tr>
<tr>
<td><a href="mailto:sphani@ksu.edu">sphani@ksu.edu</a></td>
<td>104</td>
<td>Phaninder</td>
<td>surapen</td>
<td>1004 Cardiovascular Technologist</td>
</tr>
</tbody>
</table>

Check the navigation buttons and close the browser.

**Creating an Employee Edit Java Server Faces Page**
Step 1: Open the page flow diagram, and double click on the editEmployee node. Now this opens a wizard. In the next window select the desired libraries JSF Core 1.0, JSF HTML 1.0, ADF Faces Components, ADF Faces HTML.

Click on finish and an empty page displayed in the editor.

Step 2: Now drag and drop panelpage component from the ADF Faces Core Library from the component palette on to the page design.

Step 3: In the property pane change the title of the panelpage to Employee Details. Now select the employee list from query departments node from the Data Control palette. Drag and drop the employeeelist on the page and select the create forms | ADF form.
In the edit form that appears select the Include Submit button and click on ok. Now the page should appear as follows:

Step 4: Select the mergeEntity node in the Data Controls Pallete. Now drag and drop the mergeEntity node on the submit button and select the Bind Existing CommandButton option.
Select the button and change the text to commit and action to commit in the property pane.

Now the page should like this:
Running the JSF Pages

Step 1: In order to test the page, we need to right click on editEmployees.jspx in the Applications Navigator and click on the run from the menu.
The page will be loaded in the browser and will appear as follows:
Be sure that the navigation buttons are working properly.

Step 2: From the employee list, select one employee and click on the edit button.

Step 3: In the above window the phone no appear to be 1000, now we are changing the number to 12345 and click on the commit button in order to make the changes.
Step 4: Now changes are displayed in the window clearly.

Customizing the Layout

Step 1: Select the deptEmployee.jspx and double click on it from the applications navigator in order to open the page.
Step 2: From the structure pane expand the h:form | af:panelPage | af:panelGroup | af:panelHeader Department Employees List and right click on table1 node and select the properties from menu.

Step 3: click on the column summary tab in the window and reorder the columns in the following order:

Email, Employeeno, firstname, lastname, phone, type.
Step 4: replace the header in each column with the following names:

Email, EmployeeNo, First name, last name, phone, type.
Step 5: Now click on the column details tab and set Is Sortable for columns lastname and employeeno.
Step 6: Now right click on the page and click on run
The window opens in the browser and is follows:
Now by clicking on last name and employeeno we can change to ascending order and descending order. Here by clicking on employee no the above window appears to be as follows;
Close the browser.