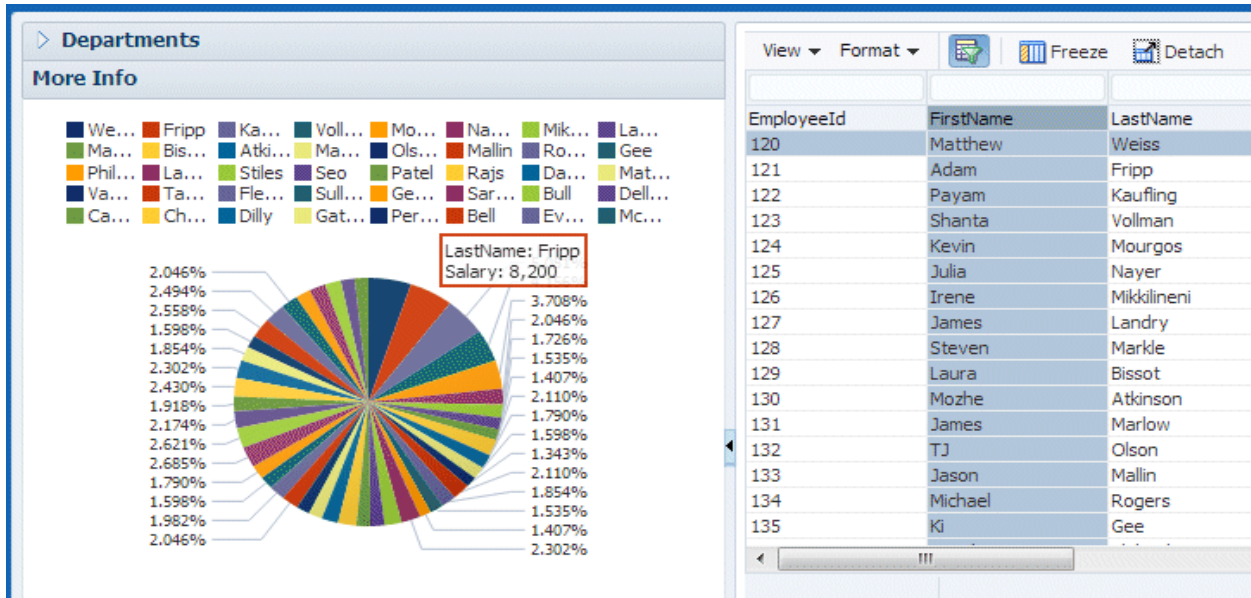


# How-To Develop Oracle JDeveloper User Interface

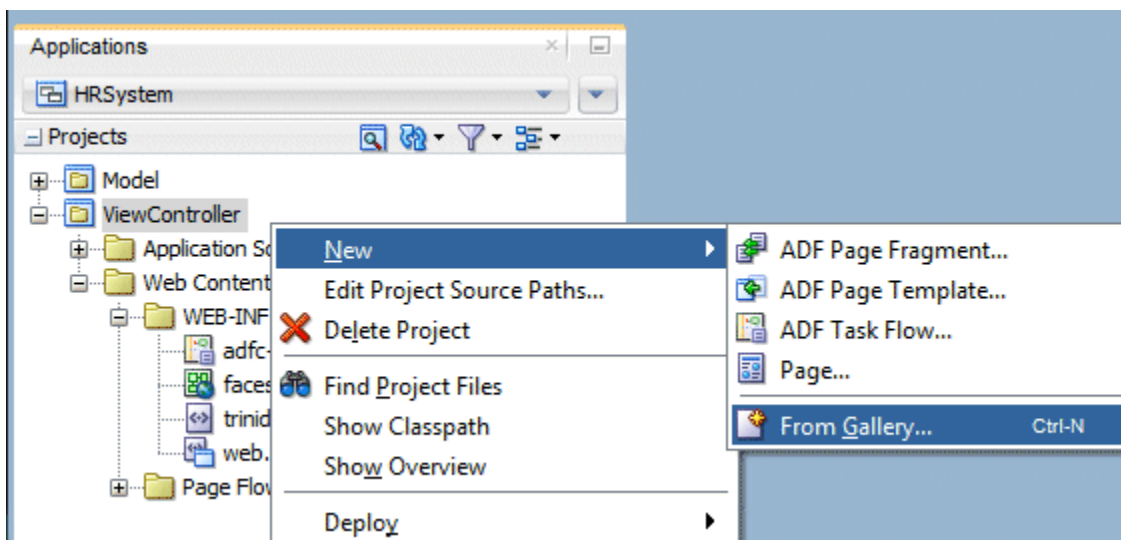
JavaServer Faces (JSF) is a standard Java EE technology that simplifies Web development. In this part of the tutorial you create a JSF page to access the business components that you created in the previous part.

You create a form that can be used to display and modify data. You use a master-detail relationship to display information about employees in a department. To build the page you'll use Oracle ADF Faces Rich Client components. These components allow you to build an Ajax-based rich Web UI without writing low-level HTML and Javascript code.



## Step 1: Create a JSF Page

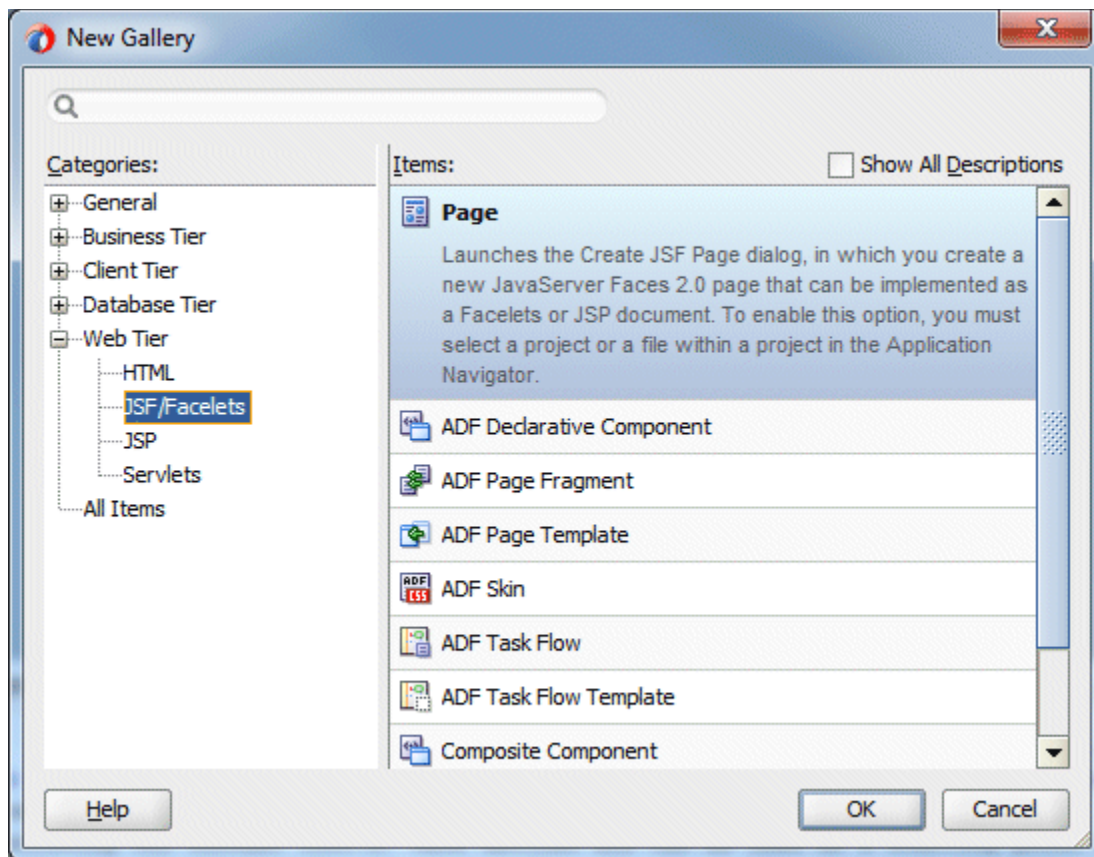
1. Right-click the **ViewController** project node in the Applications window, and select **New > From Gallery...**



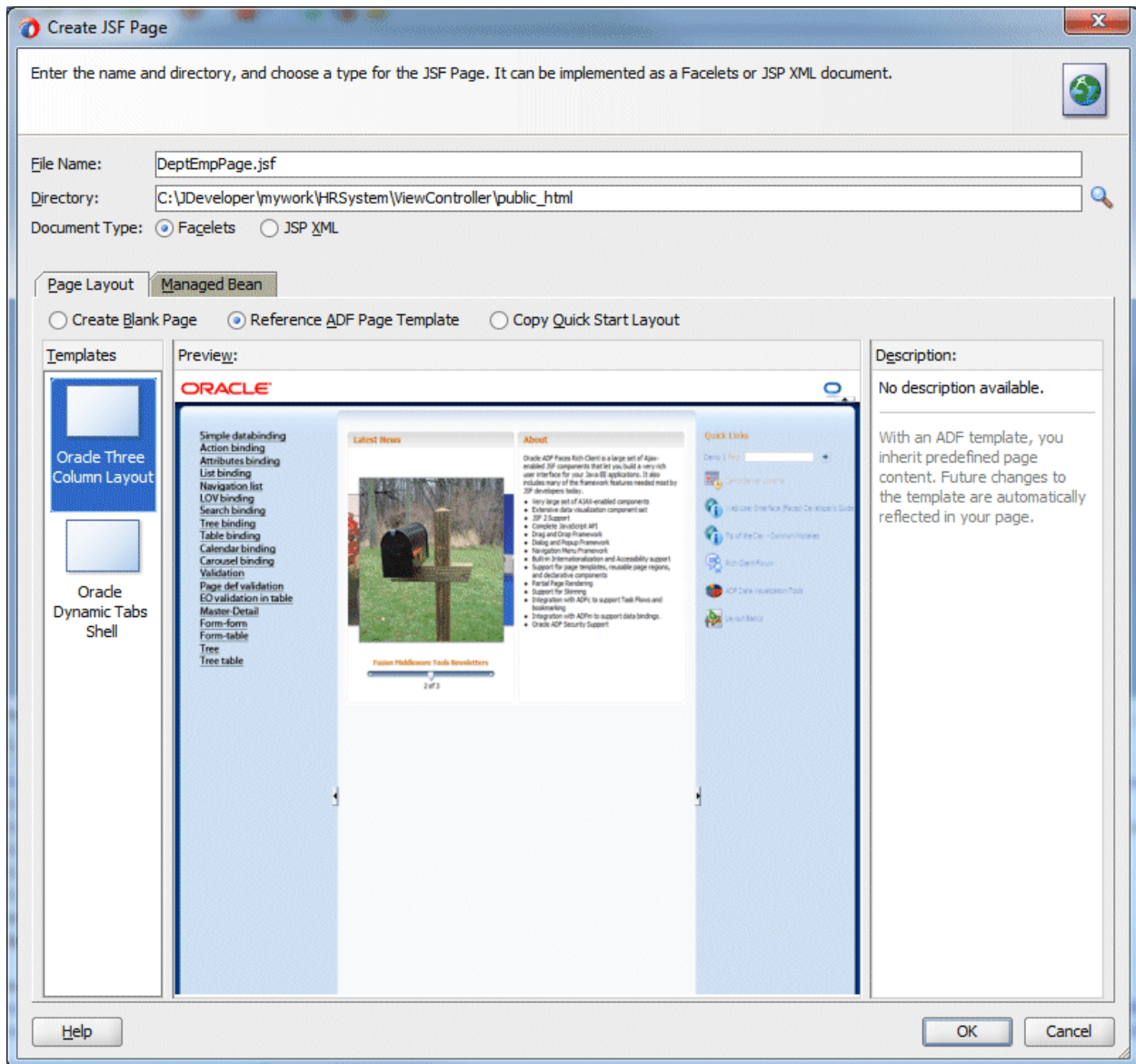
The Web part of the application is developed in a separate project called the ViewController project, which you created in the previous part of this tutorial when you created the Fusion Web application.

Maintaining a separation between the Model layer and the user interface by developing them in separate projects, makes the business services layer more reusable.

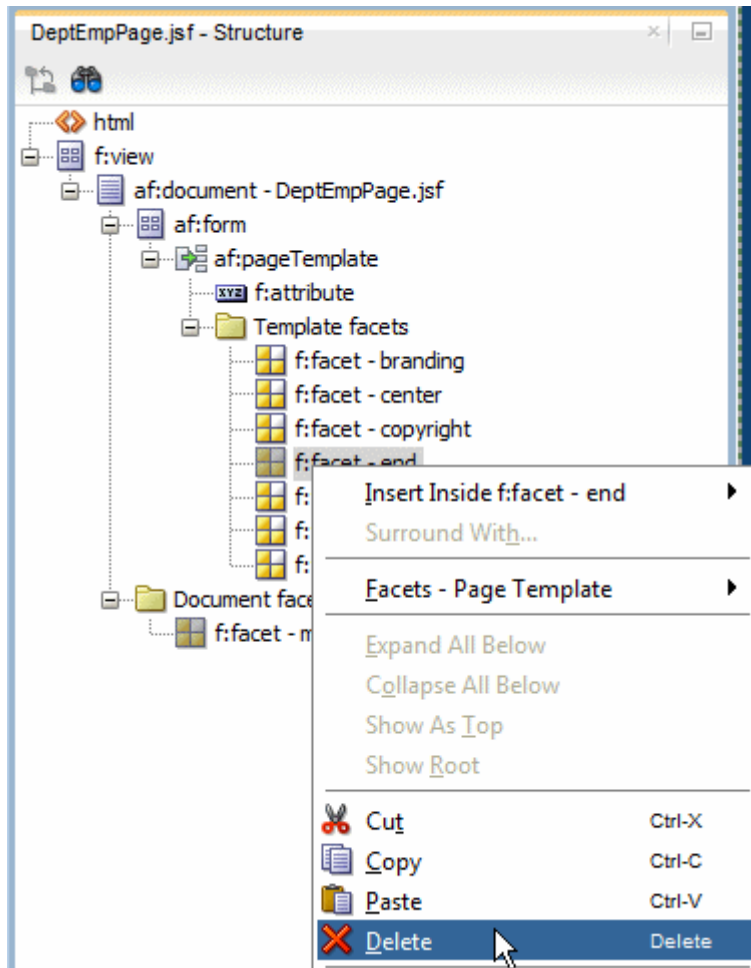
2. In the New Gallery, navigate to **JSF/Facelets** under the **Web Tier** node, and choose **Page** in the Items pane. Click **OK**.



3. In the Create JSF Page dialog, rename the page to **DeptEmpPage.jsf**; make sure the Document Type radio button is set to **Facelets**, and in the Page Layout tab select the Reference ADF Page Template radio button and choose **Oracle Three Column Layout**.

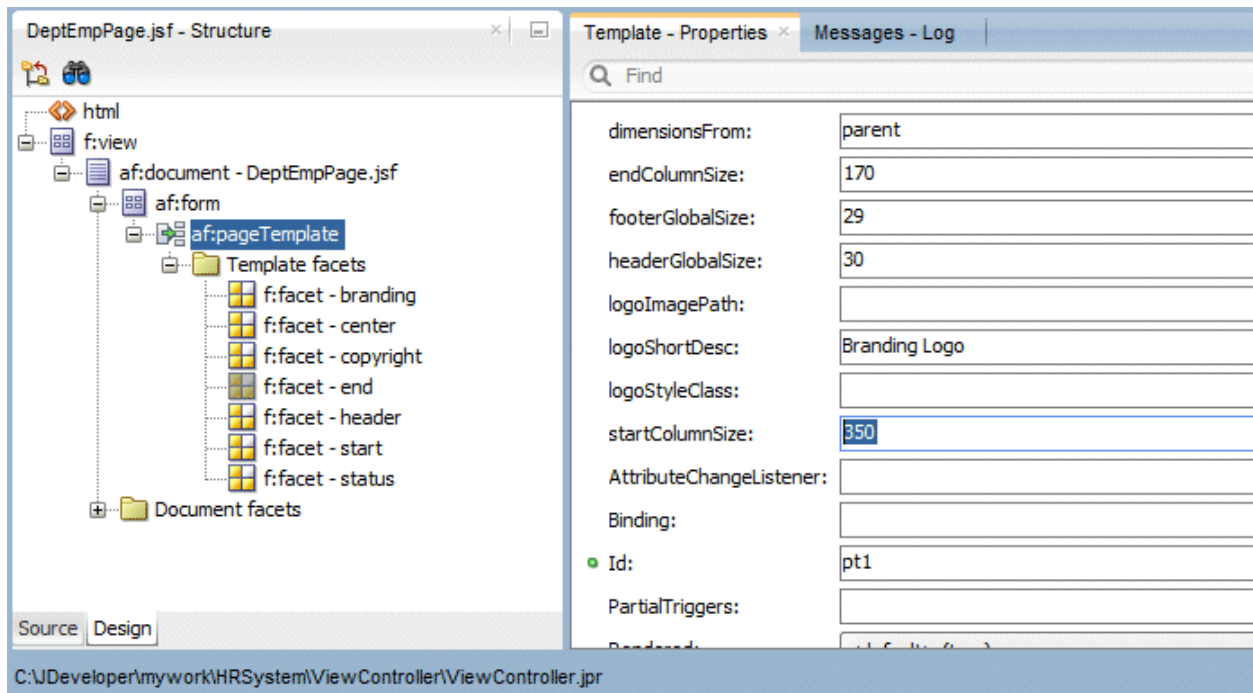


4. Click **OK**.
5. The page displays in the Design Editor. The template has three columns in it, but you only need two in the page, so delete the third one. In the Structure window, expand `af:form > af:pageTemplate > Template facets`. Right-click `af:facet - end` and choose **Delete**. You will notice that the end facet is not deleted from the list of page template facets. It is however removed from the page usage.



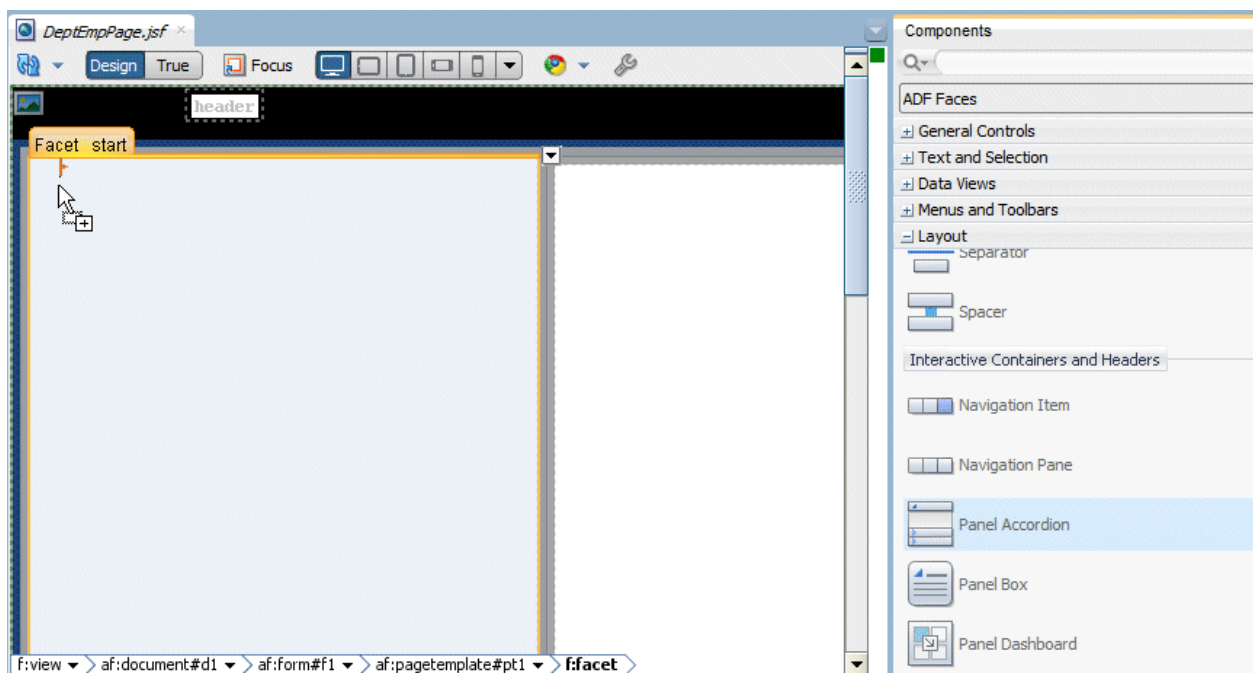
6. With **af:pageTemplate** selected in the Structure window, in the Properties window set the **startColumnSize** property to **350**.



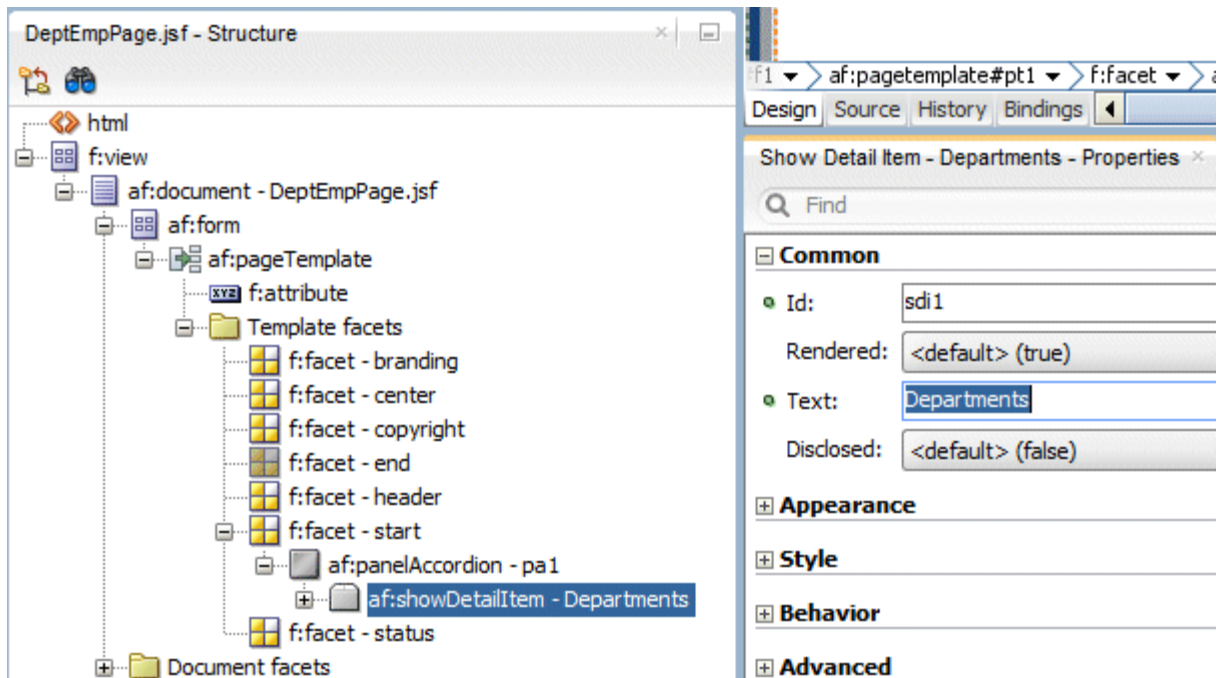


Next, add some layout components to the page.

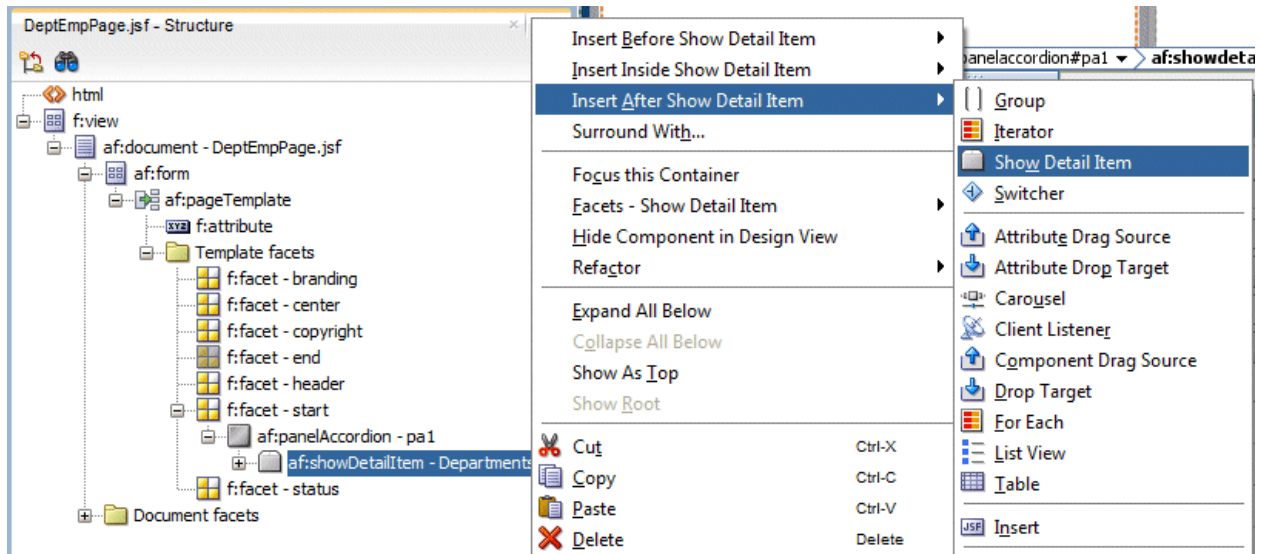
7. In the Components window expand the **Layout** group of components and locate the **Panel Accordion** component in the **Interactive Containers and Headers** section. Drag it into the **start** facet of your page.



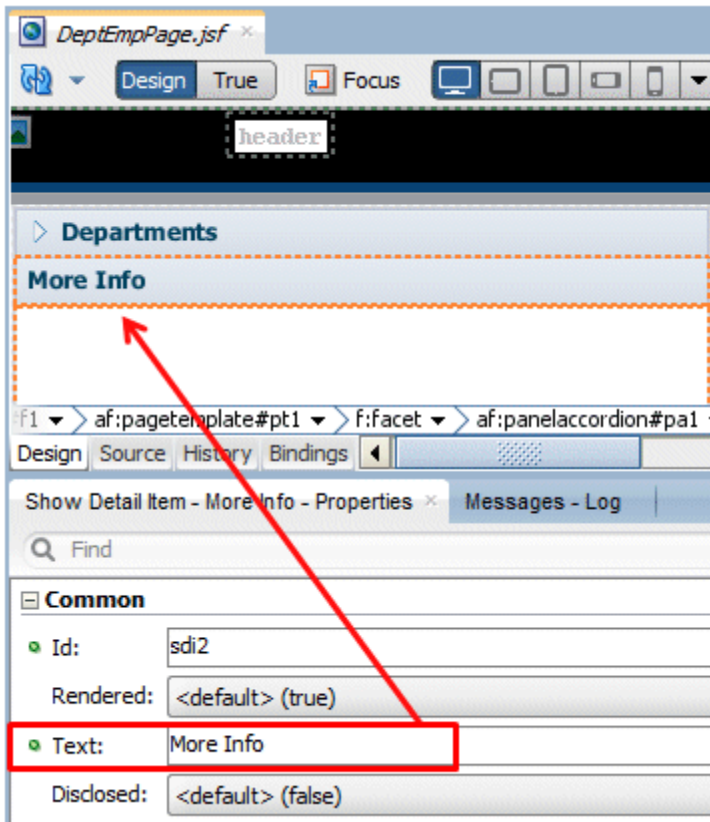
8. Select **af:showDetailItem** (the panel accordion that you just added to the page in the previous step) in the Structure window and in the Properties window change its Text property to **Departments**.



- Right-click the new **Departments Show Detail Item**, and choose **Insert after Show Detail Item - Departments > Show Detail Item**. This adds another accordion to your page.

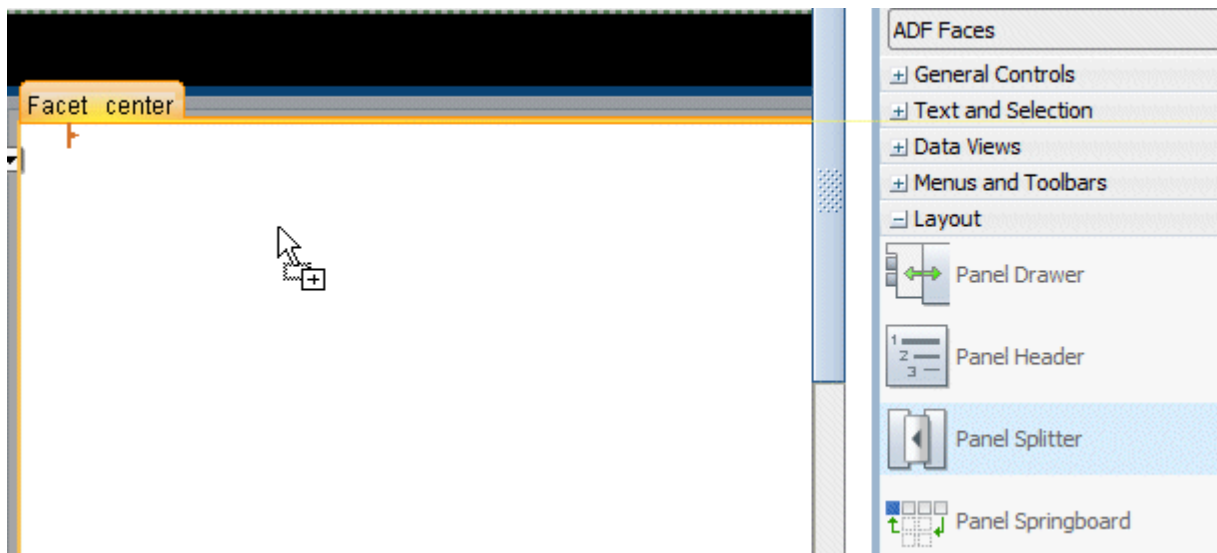


- Change the **Text** property to **More Info**.

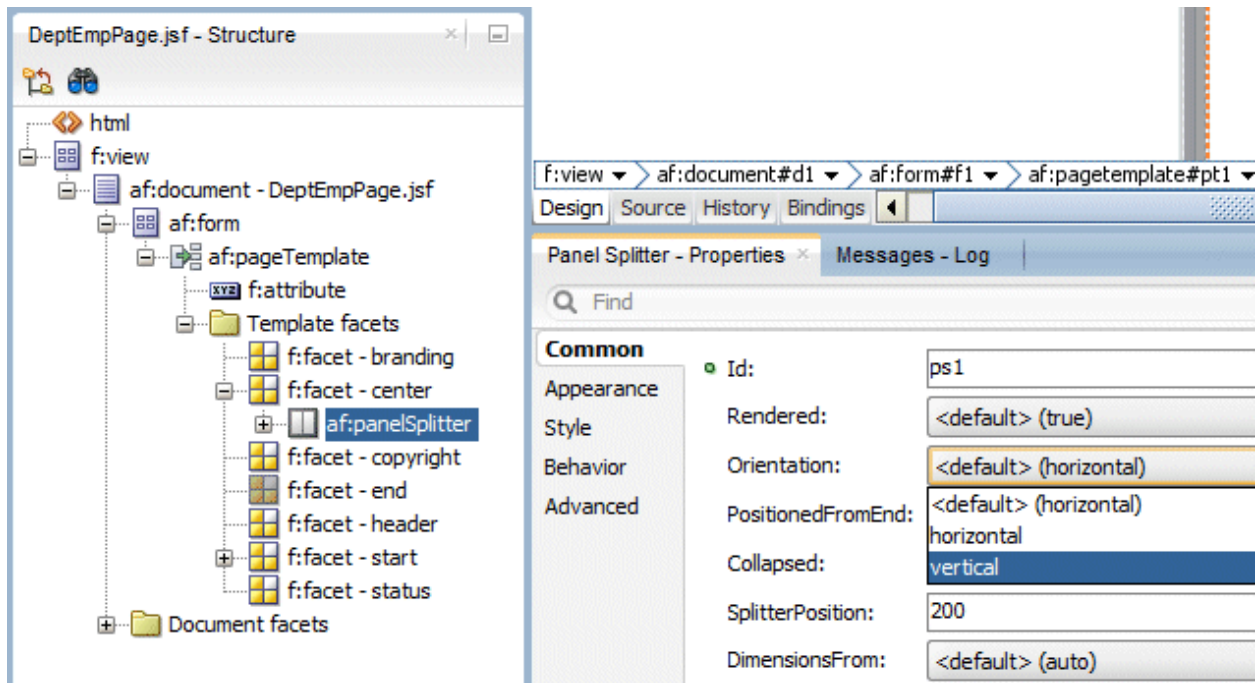


This is another way of adding components to a page using context menus directly inside the Design Editor (instead of dragging and dropping from the Components window).

- From the **Layout** components, click and drag a **Panel Splitter** component onto the **center** facet of your page.

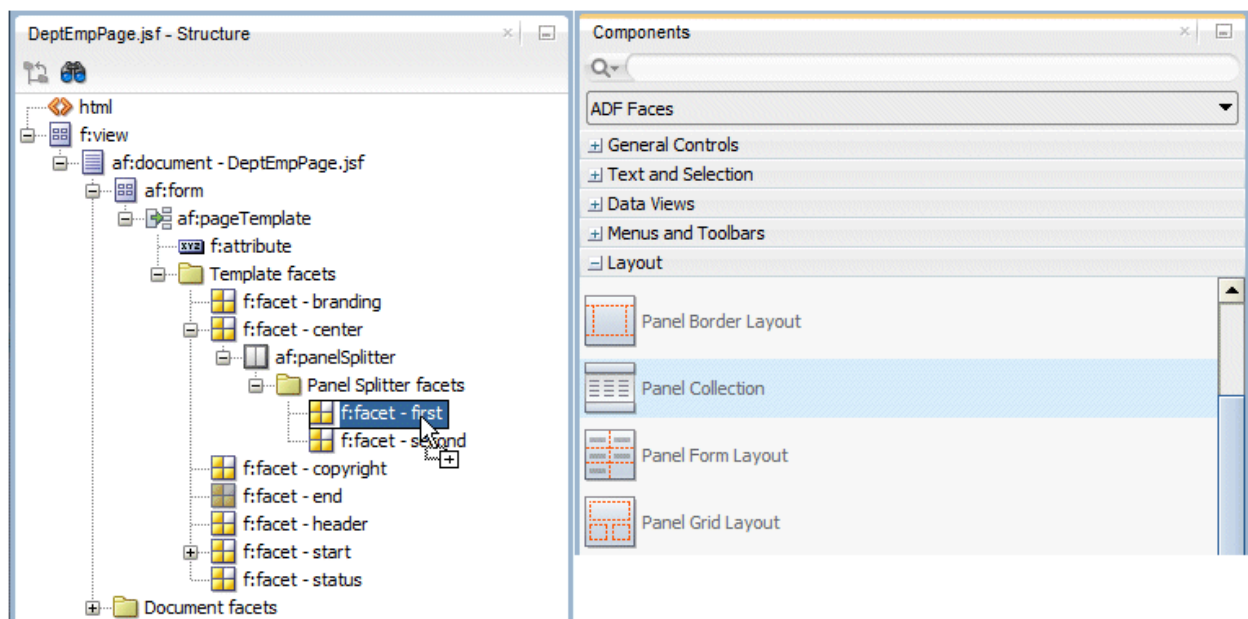


- In the Properties window change the **Orientation** property of the new splitter to **vertical**.



- From the **Layout** components, drag a **Panel Collection** component into the **first** facet at the top of the splitter on your page.

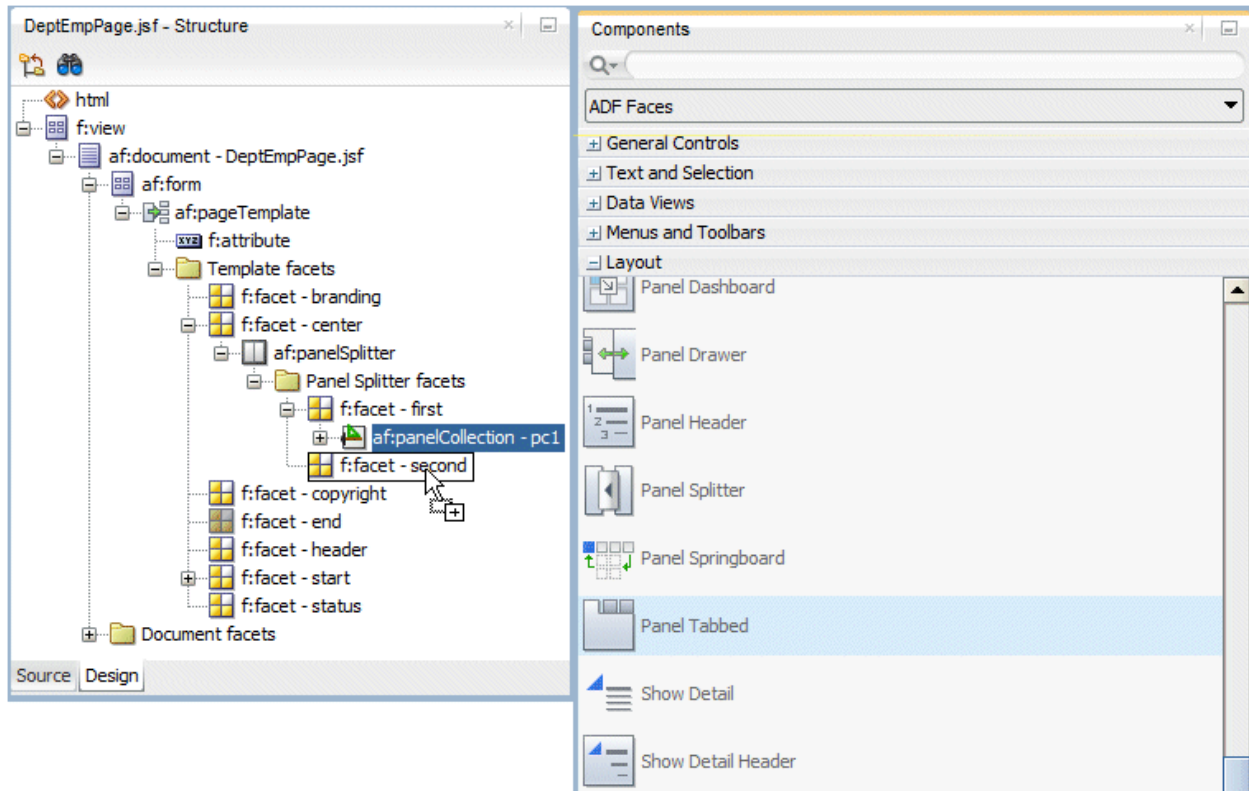
It might be easiest to use the Structure window to drop the component in the correct position.



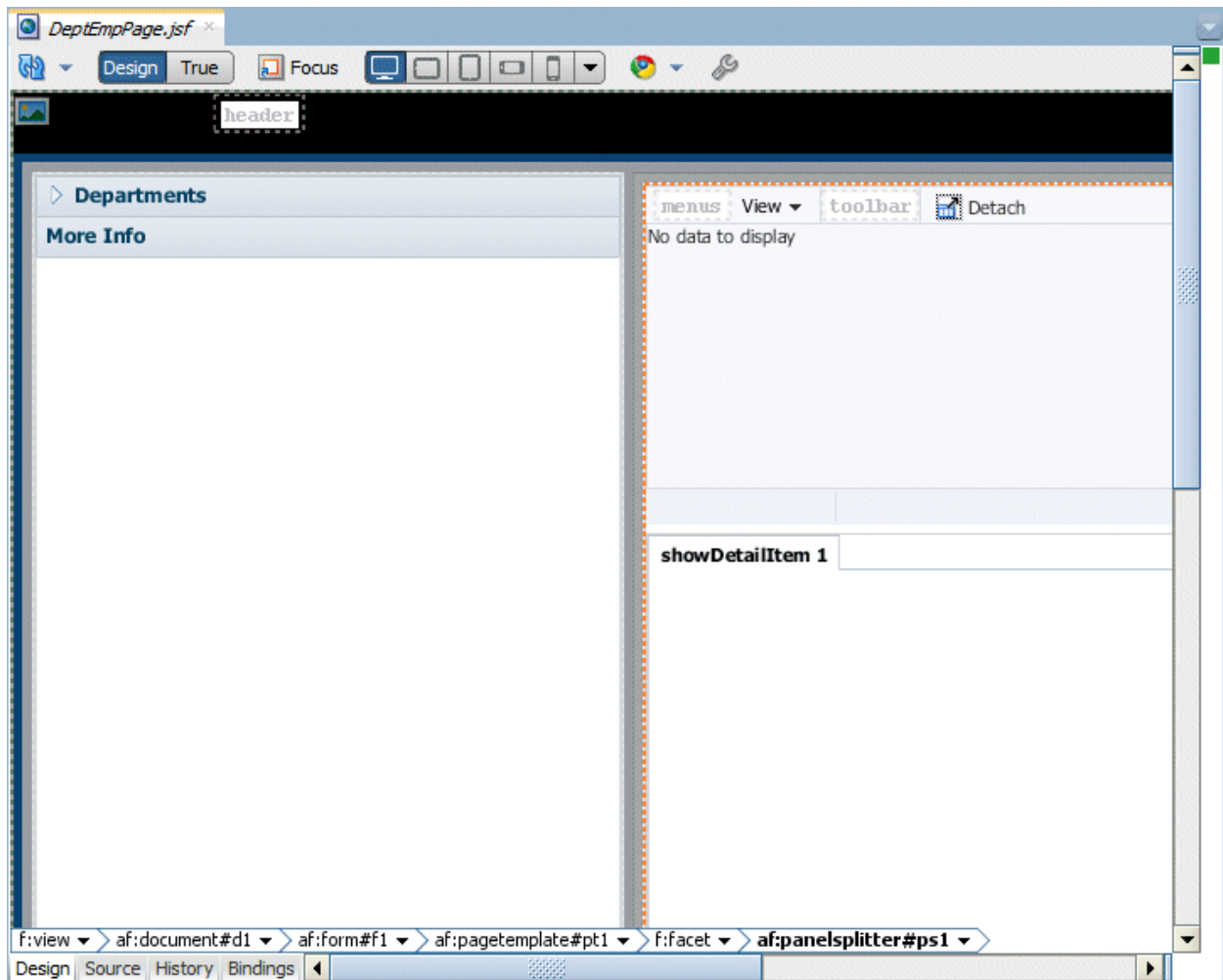
A panel collection is a component that aggregates collection components like table, treeTable and tree to display standard/application menus, toolbars and status bar items.

- Using the Structure window, drag a **Panel Tabbed** component into the **second** facet of the splitter on your page.





15. Double-click the tab **DeptEmpPage.jsf** at the top of the page to view it in its entirety. The page should look like the screen shot below.

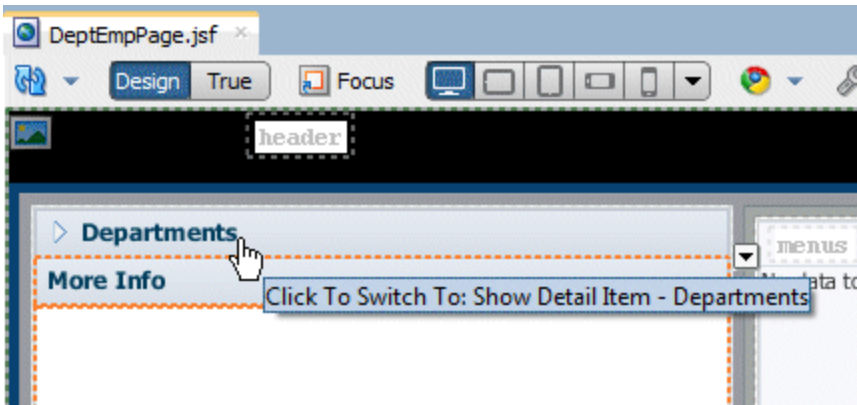


16. Click the **Save All** icon on the JDeveloper menu bar to save your work.

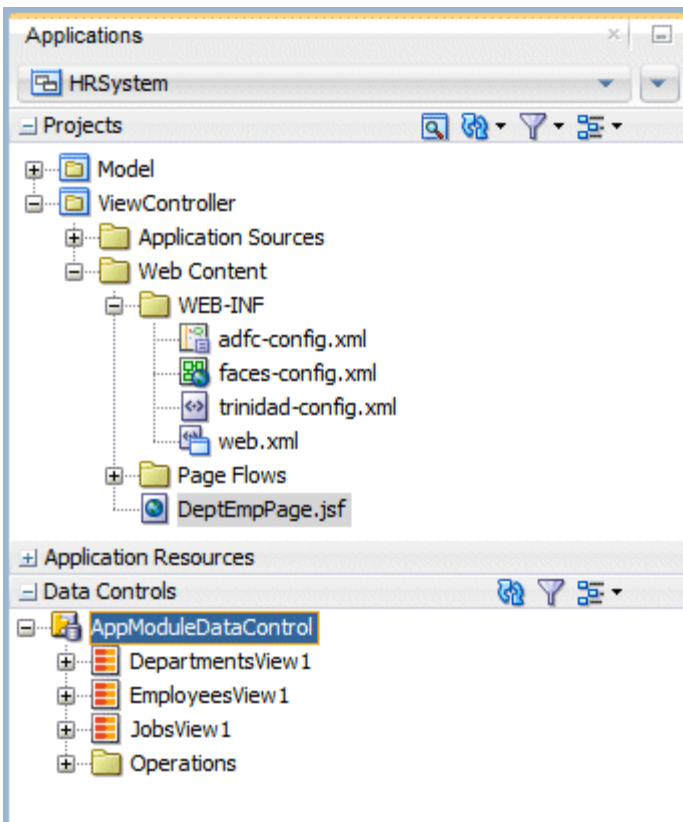
## Step 2: Bind Data Controls to the Page

In this step you bind the business components you created in Part 1 of this tutorial to the user interface. You do this with simple drag and drop operations; behind the scenes the ADF Model layer takes care of binding the data for you.

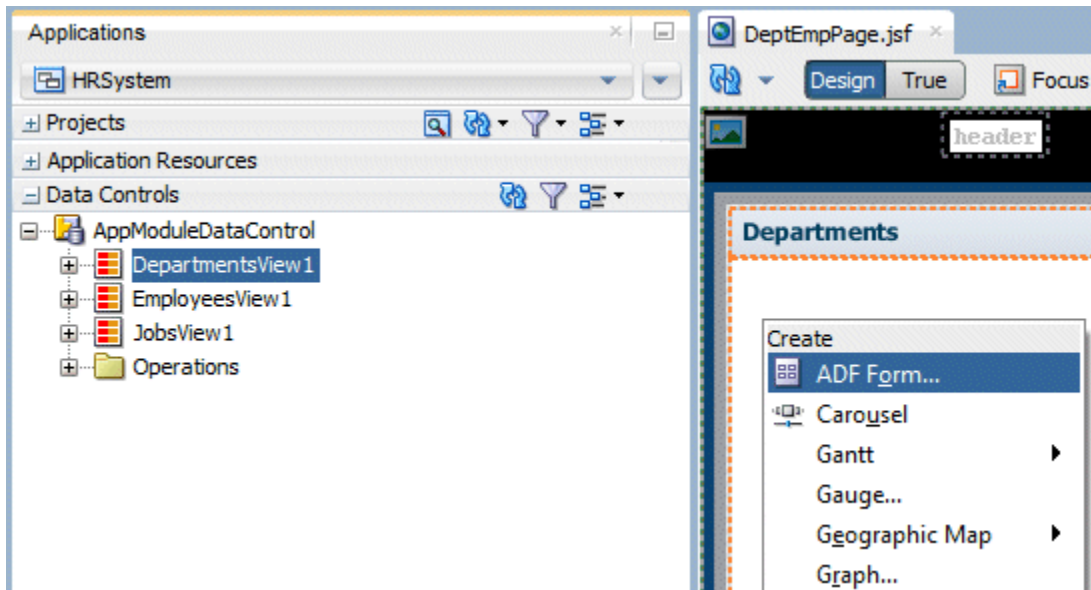
1. In the Design Editor click the **Departments** accordion to expand it.



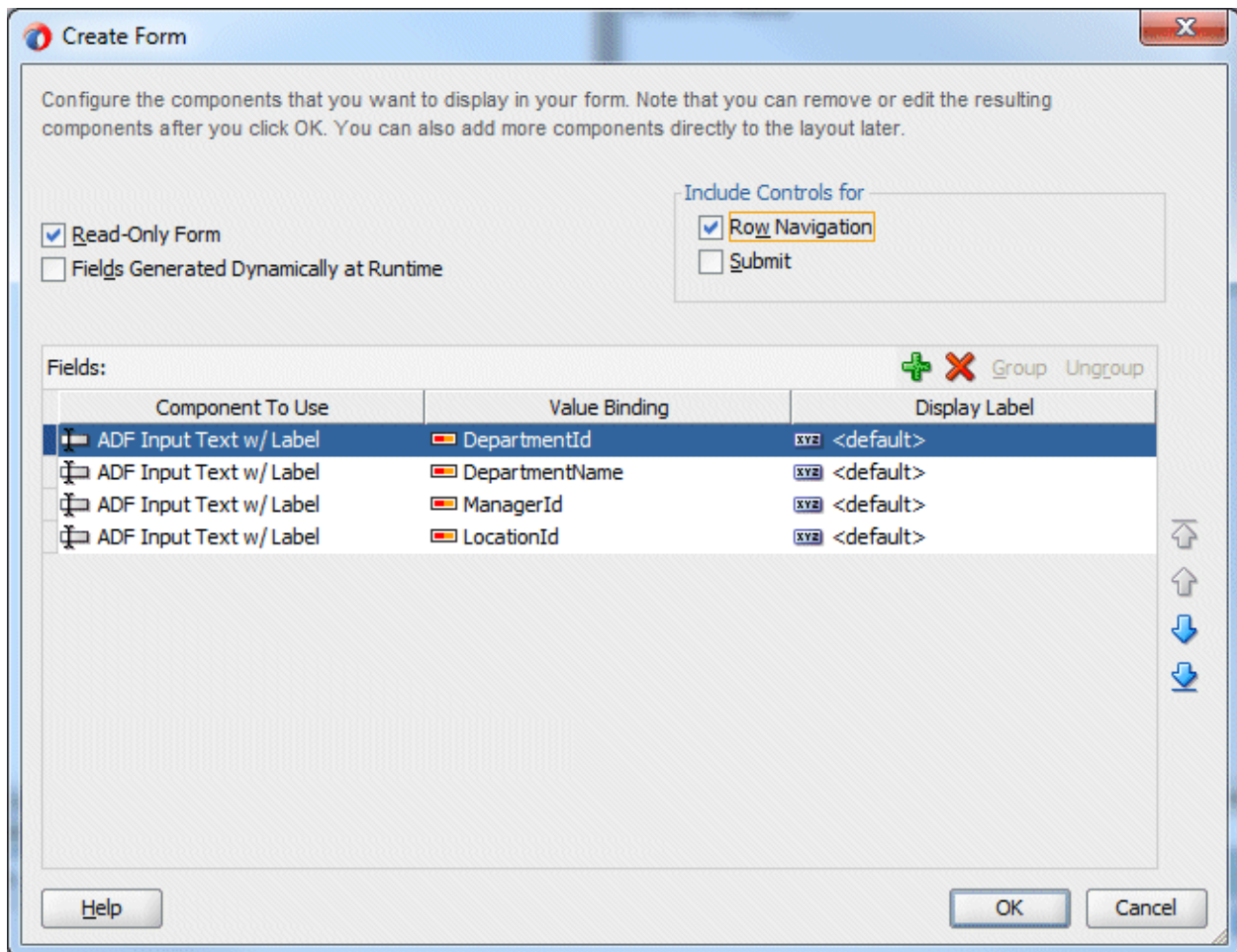
2. In the Applications Navigator expand the **Data Controls** accordion, and in it expand **AppModuleDataControl** to expose the business services you defined in the first part of this tutorial.



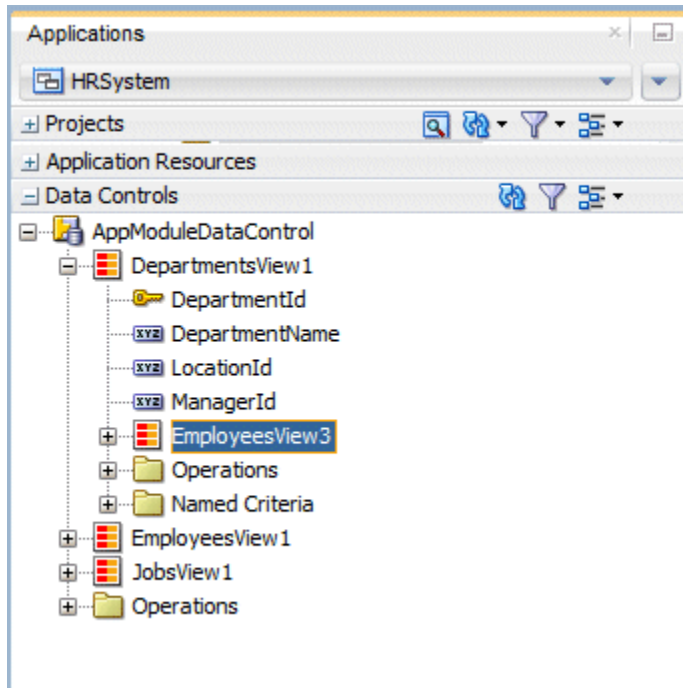
3. Drag the **DepartmentsView1** data control into the **Departments** accordion in your JSF page. When prompted to choose the type of component you want to create, choose **ADF Form**.



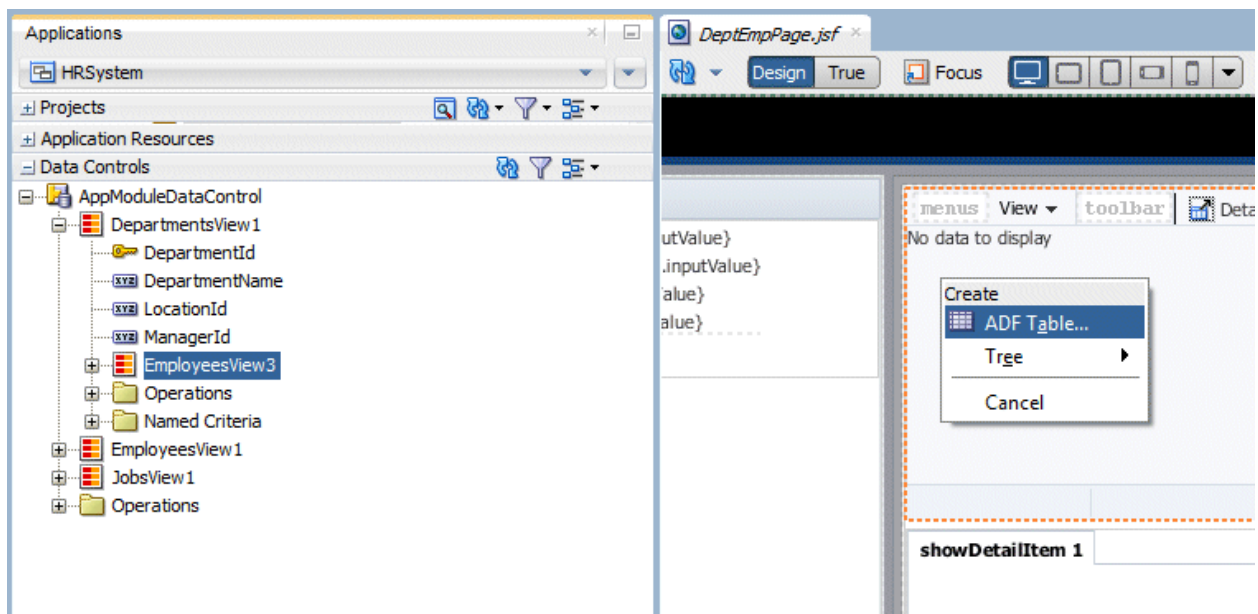
4. In the **Edit Form Fields** dialog, check the **Read-Only Form** and **Row Navigation Controls** checkbox and click **OK**.



- In the Data Controls accordion expand **DepartmentsView1** to expose the fields it contains, as well as the related **EmployeesView3** control for the employees in each department. Note that these are the "detail" employees for the "master" departments - they are linked and therefore at runtime, the employees you see in this view object are the ones who belong to the selected (or viewable) department.

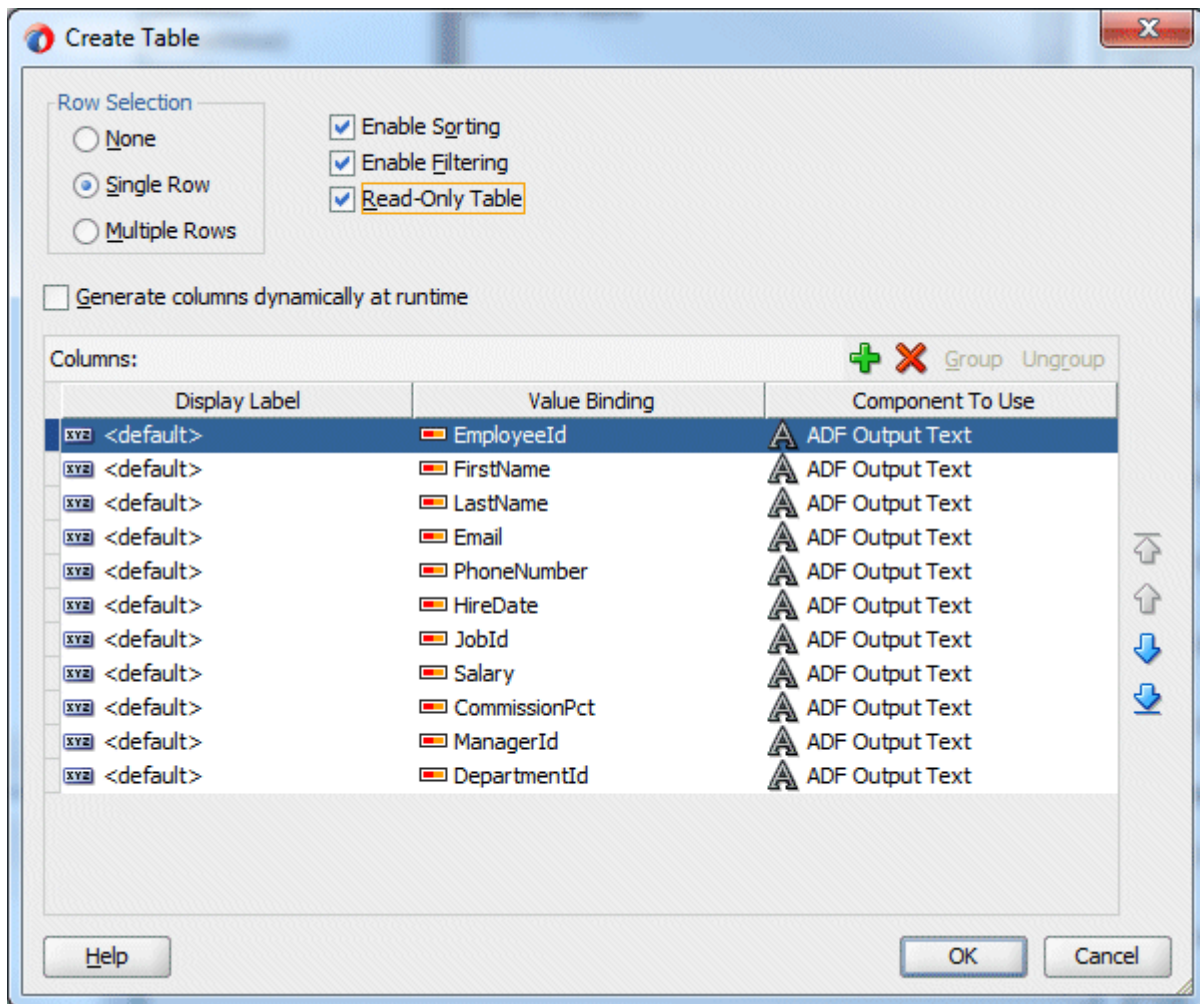


- Drag the **EmployeesView3** data control onto the **Panel Collection** that you created in the first facet of the PanelSplitter. Create it as an **ADF Table...**

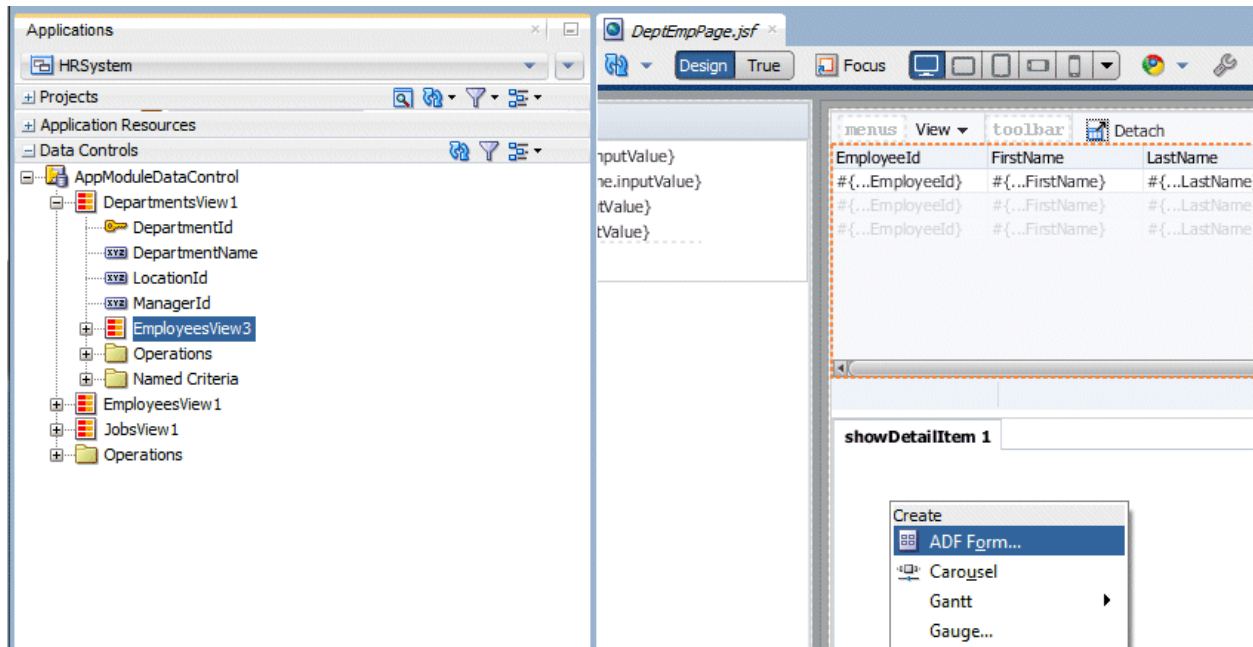


- In the Edit Table Columns dialog, choose **Single Row** in **Row Selection**, and check the checkboxes for **Sorting** and **Filtering and Read-Only Table** checkboxes. Click **OK**.



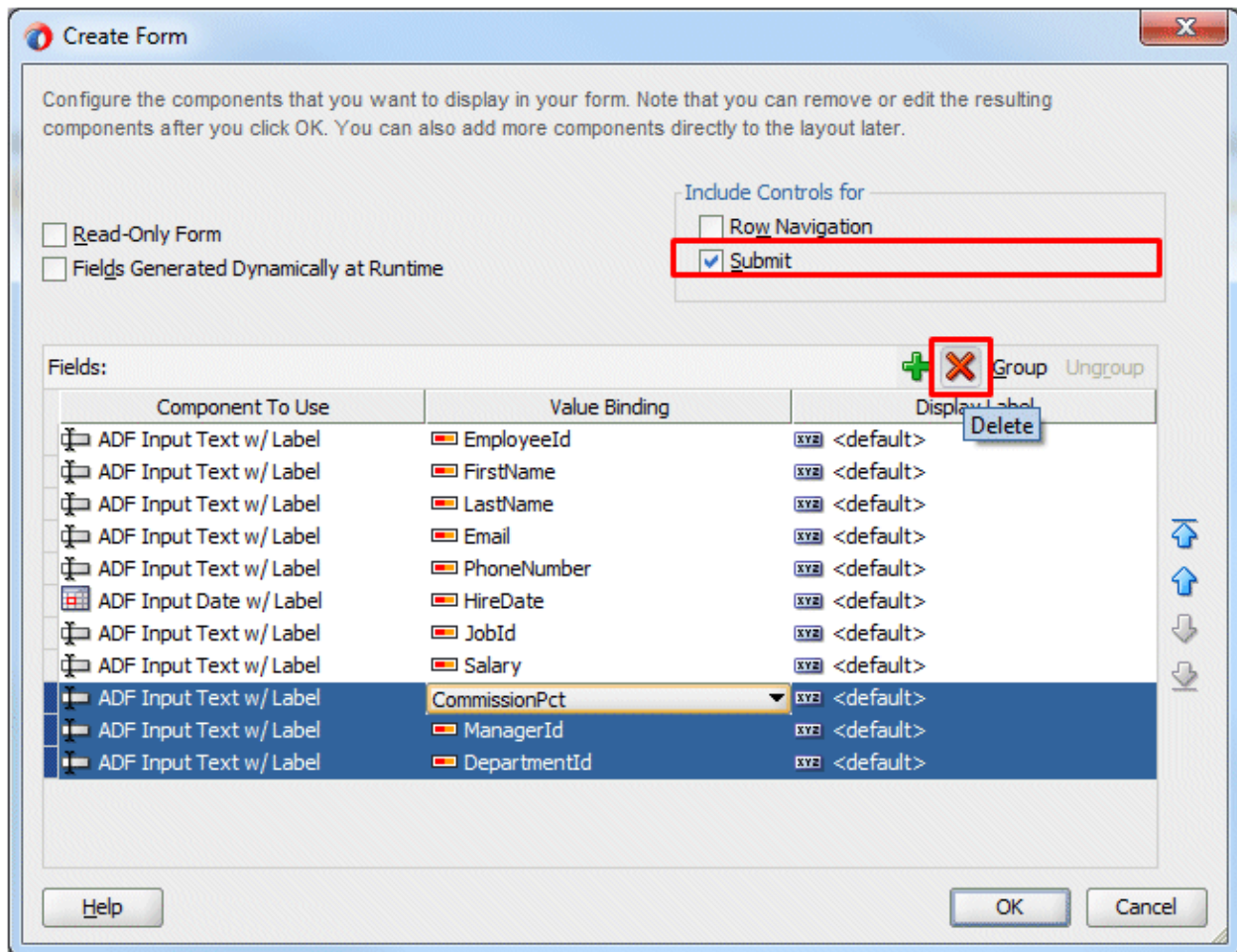



8. Drag the **EmployeesView3** data control again but this time onto **showDetailItem 1** in the **panelTabbed** component in the second facet of the **panelSplitter**. Create it as an **ADF Form**.

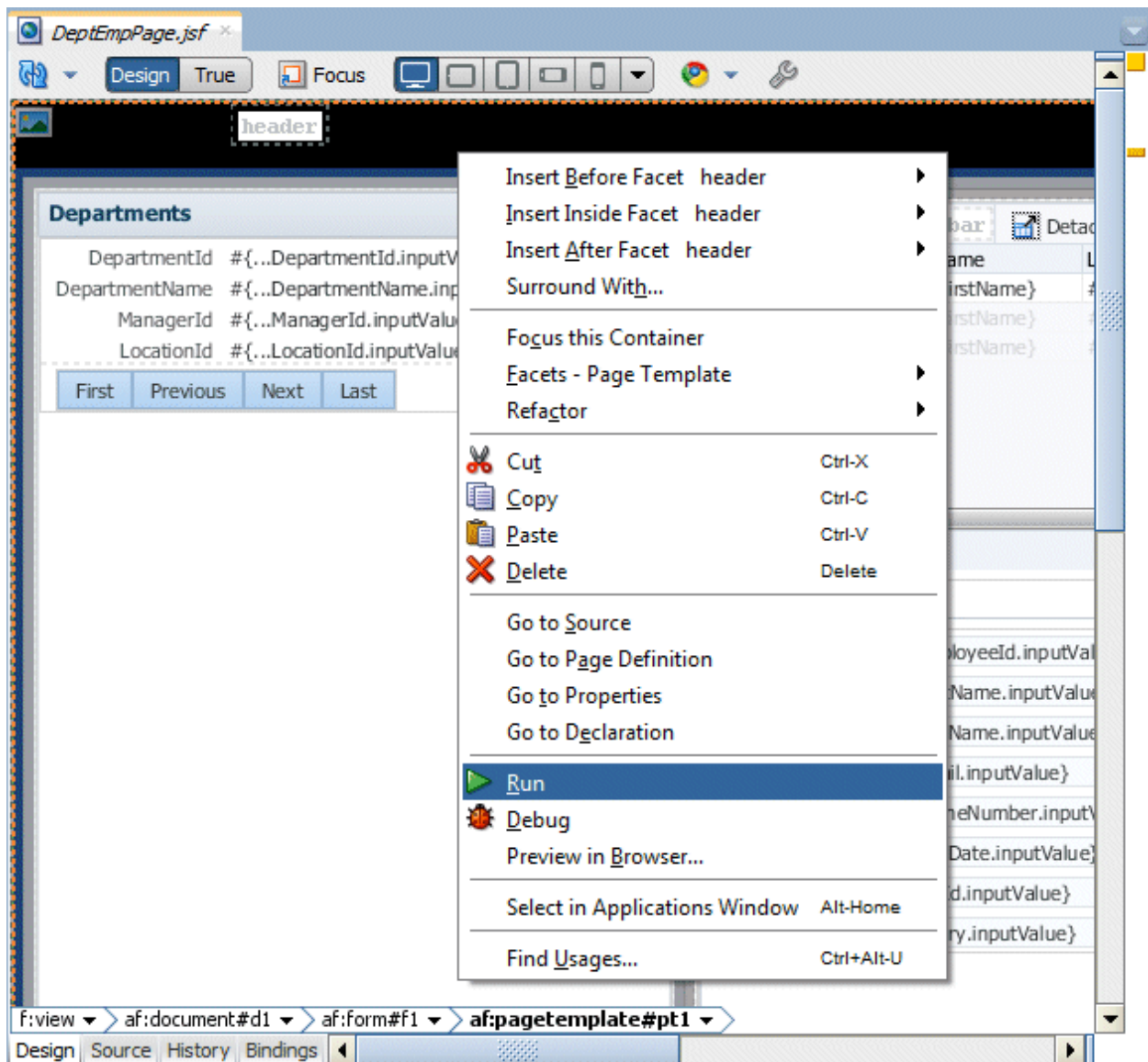


9. In the Edit Form Fields dialog, check the **Include Submit Button** check box. Using your mouse and the Shift key select the bottom three fields from the list: **CommissionPCT**, **ManagerId**, **DepartmentID** and delete them by clicking the **Delete** button .

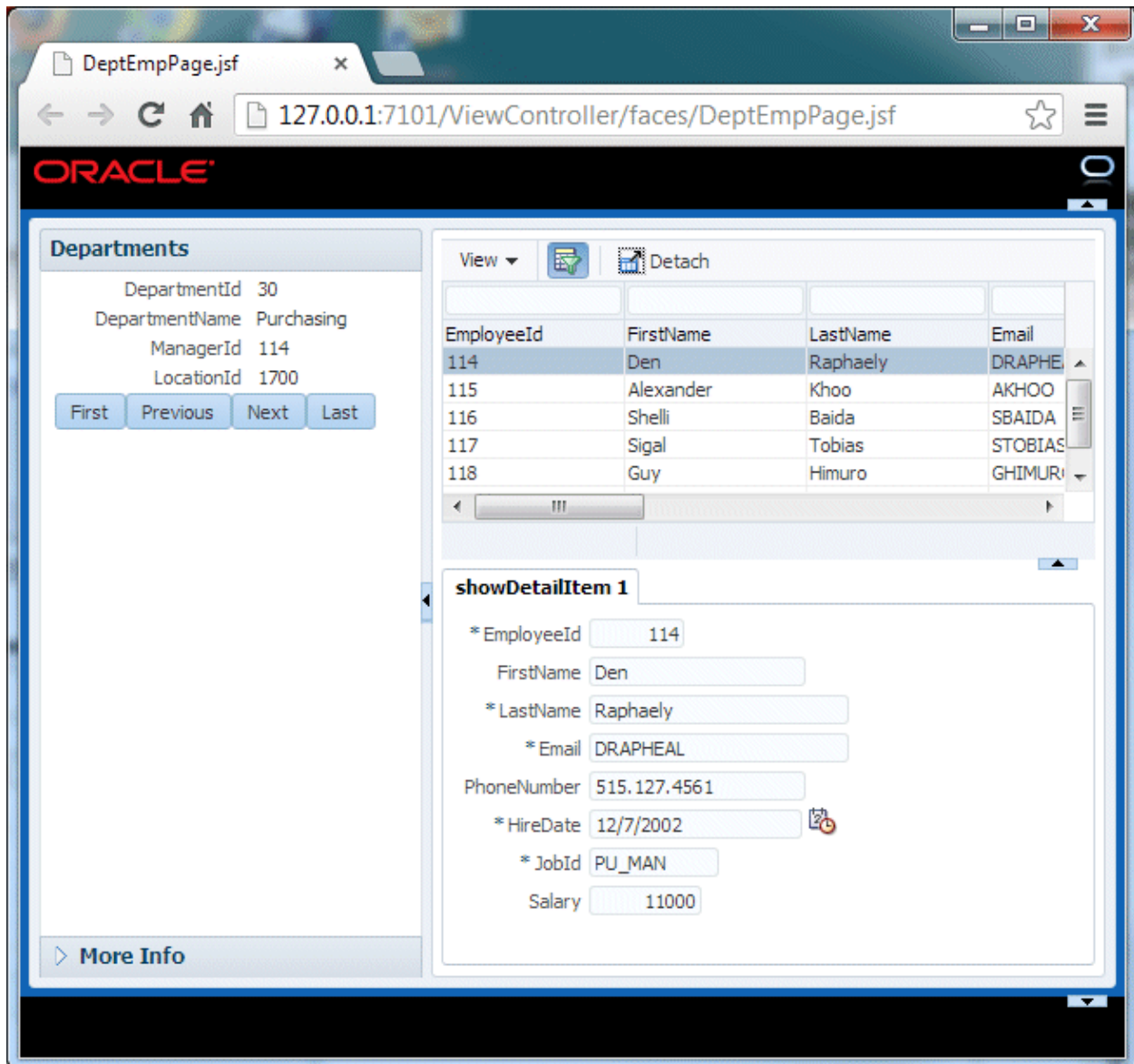
Click **OK**.



10. Click the **Save All**  icon on the JDeveloper menu bar to save your work. Right-click in the page and choose **Run**. This will compile your project, build it, and launch the integrated WebLogic Server to run it. The default web browser opens to display the page. You can follow the progress of these steps in the Log window in JDeveloper.



11. Once the page displays in the browser, resize the page area using the **splitter** to display the department data. Then use the **Next** button to scroll through the departments. Notice that the employee data changes to show the employees within the displayed department. Notice also that the employee form displays the currently selected employee.



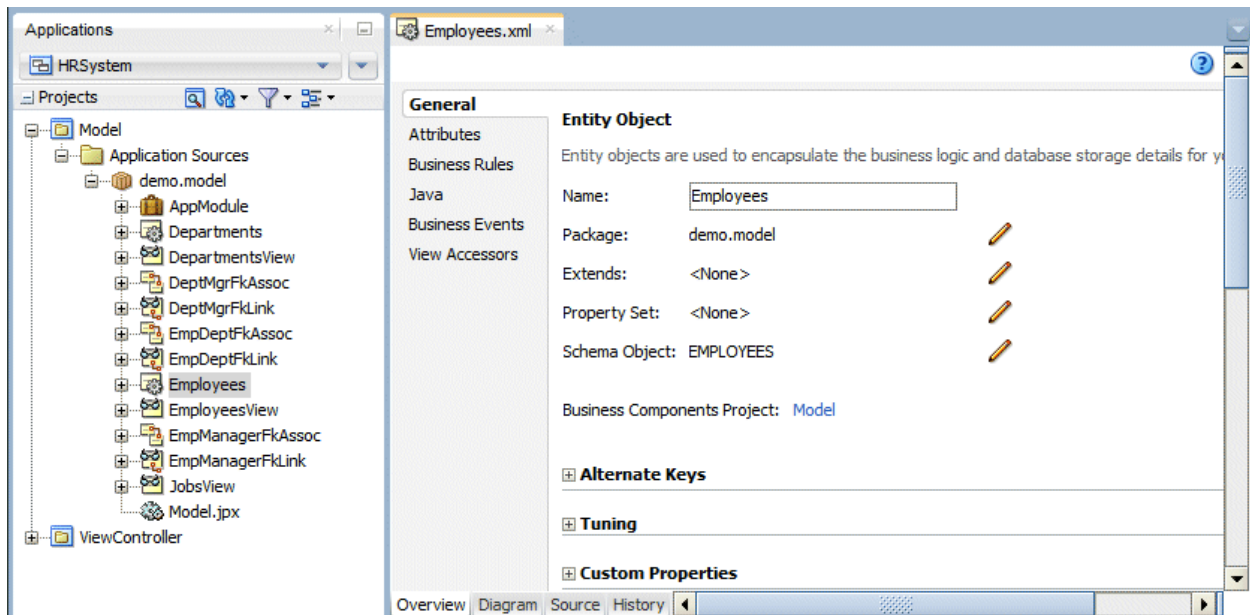
12. Close the browser window.

### Step 3: Refine the Business Services

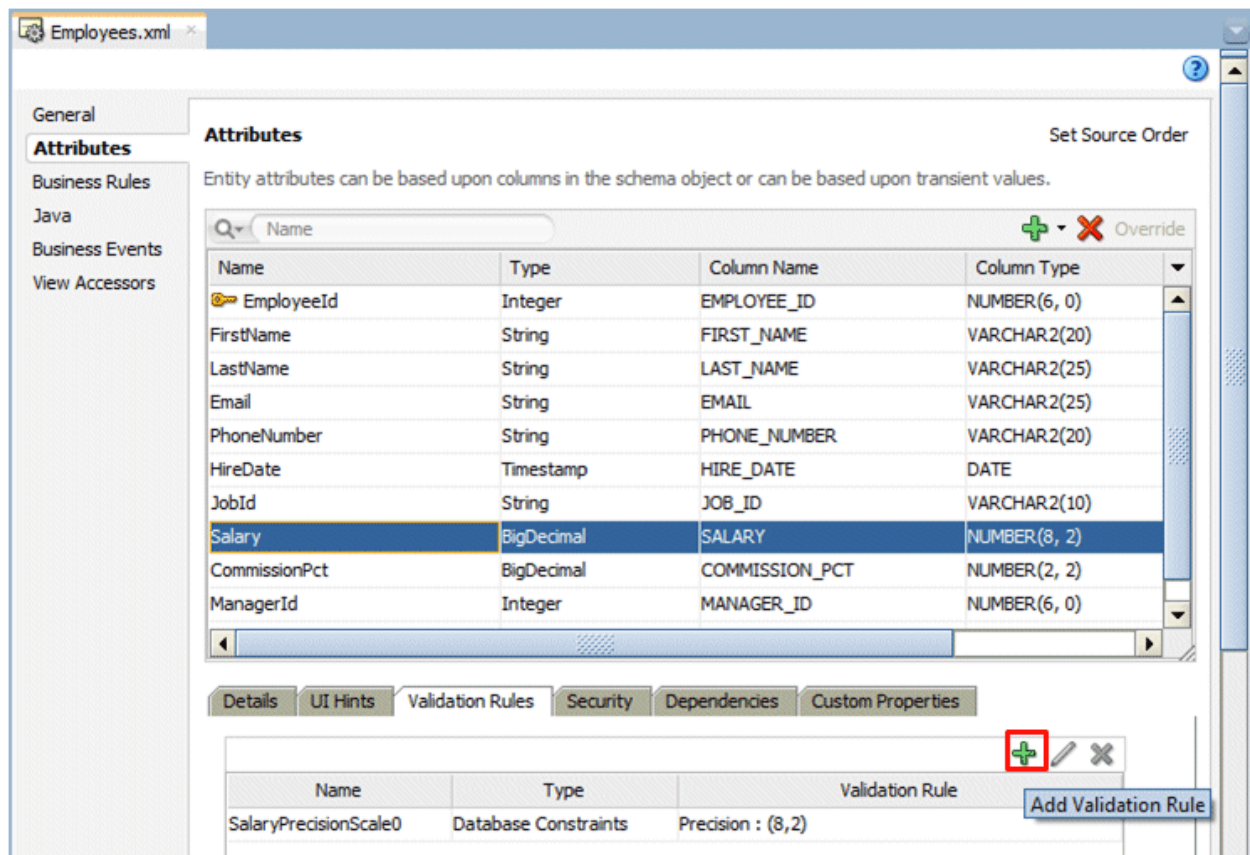
In these steps you refine the business services by adding validation rules, formatting and default values.

1. In the Applications window under the **Model** project, locate the **Employees** entity object and double-click it to open it for editing.





2. In the Employees.xml window, click the **Attributes** finger tab and locate and select the **Salary** attribute. Scroll down in the page to the Validation Rules tab and click the **Add** icon to add a new validation rule.



- Keep the default Name and in the Add Validation Rule dialog choose **Range** from the Rule Type drop down list. Note the various other types of rules you can define here. In the Operator field ensure that the value is set to **Between**, and type **0** for the minimum value and **99000** for the maximum value fields.

Define the Validation you want to perform with this rule and configure the Validation Failure response.

Name:

Description:

Type:

Rule Definition | Validation Execution | Failure Handling

Attribute:

Operator:

Range

Minimum Value:

Maximum Value:

Hint: Enter valid values for the selected attribute's type.

Help OK Cancel

- Click the **Failure Handling** tab to define an error message that will display when the validation fails. In the **Message Text** field type an error message like **"Salary out of range 0 to 99,000"** and click **OK**.

**Add Validation Rule for: Salary**

Define the Validation you want to perform with this rule and configure the Validation Failure response.

Name:

Description:

Type:

Rule Definition    Validation Execution    Failure Handling

Failure Severity  Error     Informational     Warning

Failure Message \_\_\_\_\_

Enter text for the translatable validation failure messages.

Message Text:

Token Message Expressions:

Message Token	Expression

Ignore Detailed Error Messages

Hint: to add a token message expression, include a token of the form {0} in your error message string.

Help    OK    Cancel

- Next you add a default value to the hire date field - so that when a new employee is created the hire date defaults to the current date.  
In the **Employees.xml** window, locate the **HireDate** attribute, and select it. In the **Value** section of the Properties window, locate the **Default Value Type** property and select **Expression**, then set the **Default Value** property to **adf.currentDate**. This will ensure the default value for a new record is set to today's date.

The screenshot shows the Oracle ADF IDE interface. The top window, titled 'Employees.xml', displays the 'Attributes' table. The table lists various attributes with their types and corresponding database column names. The 'HireDate' attribute is highlighted in blue. Below this, the 'Properties' window for 'HireDate' is open, showing configuration options. In the 'Value' section, the 'Mandatory' checkbox is checked, and the 'Updatable' dropdown is set to 'Always'. Under 'Default Value Type', the 'Expression' radio button is selected. The 'Default Value' field contains the expression 'adf.currentDate'. The 'UI Hints' section is currently empty.

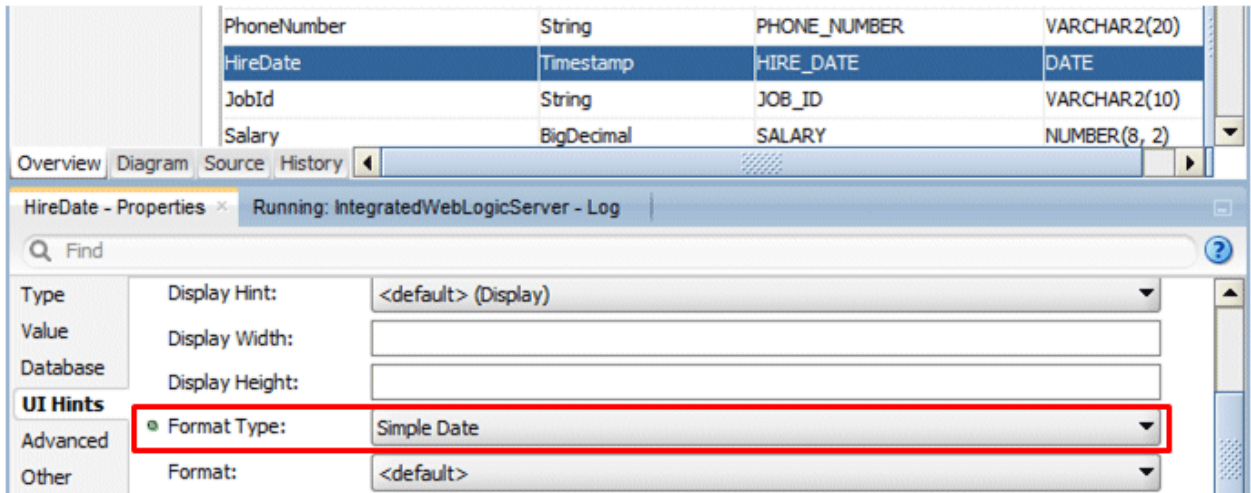
Name	Type	Column Name	Column Type
EmployeeId	Integer	EMPLOYEE_ID	NUMBER(6, 0)
FirstName	String	FIRST_NAME	VARCHAR2(20)
LastName	String	LAST_NAME	VARCHAR2(25)
Email	String	EMAIL	VARCHAR2(25)
PhoneNumber	String	PHONE_NUMBER	VARCHAR2(20)
HireDate	Timestamp	HIRE_DATE	DATE
JobId	String	JOB_ID	VARCHAR2(10)
Salary	BigDecimal	SALARY	NUMBER(8, 2)

Properties for HireDate:

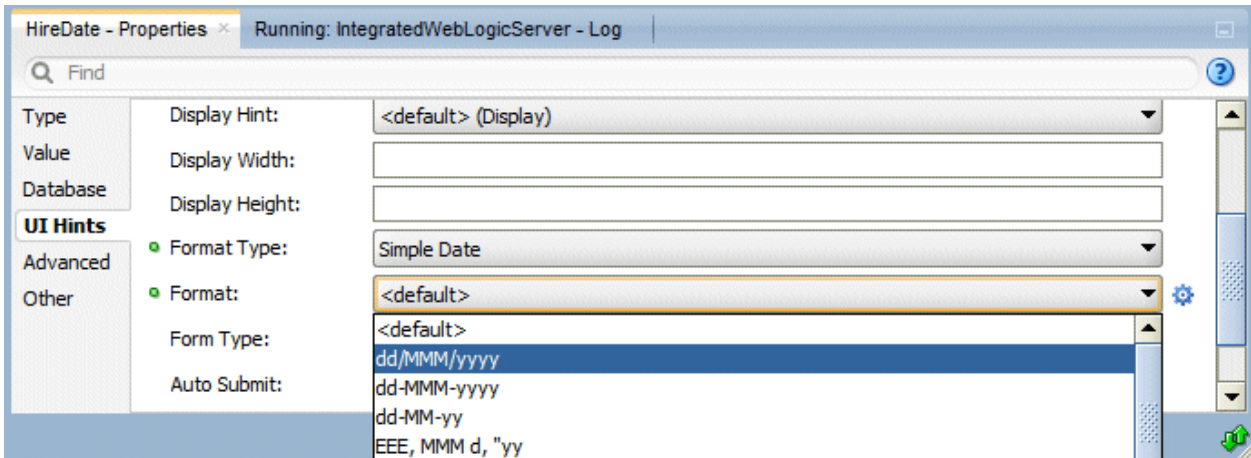
- Type: Value
- Database:  Mandatory    Updatable: Always
- Default Value Type:  Literal     Expression     SQL
- Default Value:
- Refresh Expression Value:
- UI Hints: (Empty)

You can also specify UI Hints for attributes to control how they display by default in the forms and pages that use them. In this step you add a default format mask for the hire date field. You can also specify labels and tooltip help here.

6. In the **UI Hints** section of the Properties window locate the **Format Type** property and select **Simple Date**.

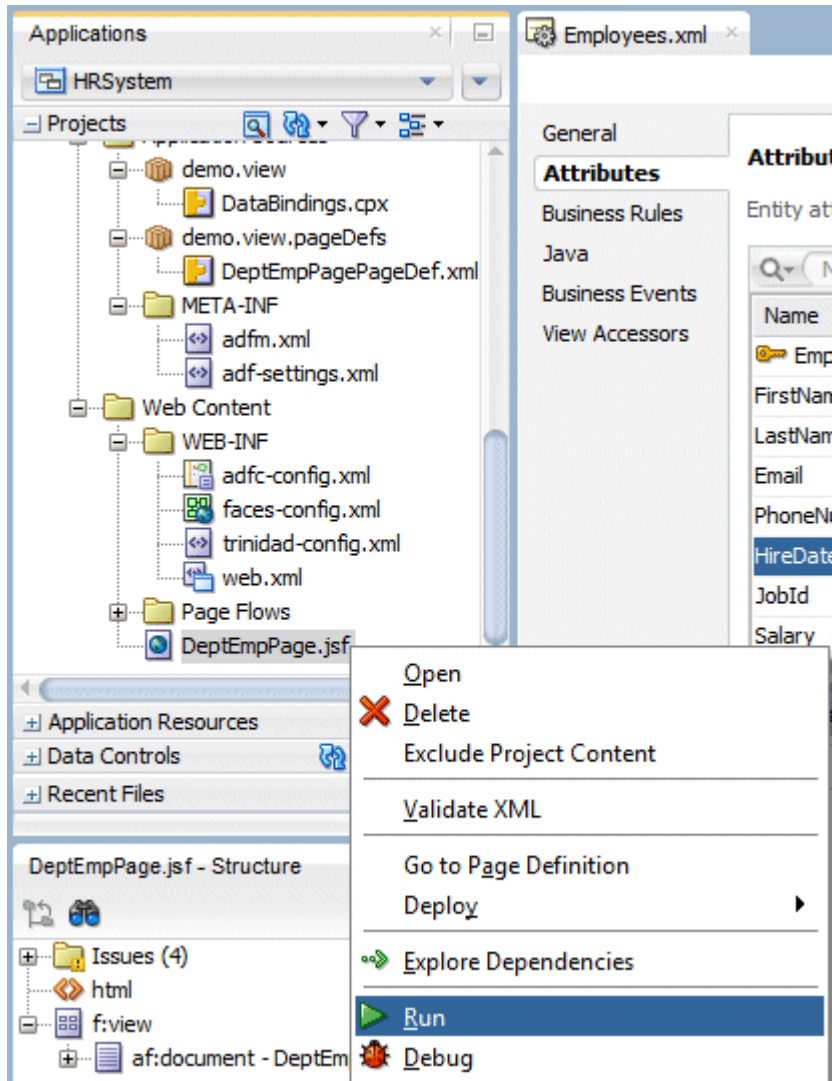


Set the **Format** property to **dd/MMM/yyyy**. You should get a warning message indicating that this will change the format in a resource bundle. Click **Yes** to acknowledge the message.



7. Run the page again to test the refinements you have made.





8. Test the **HireDate** field. Notice that as soon as you click in the field you are provided with an example of the format for the date (as you defined it).

The screenshot shows a web browser window with the URL `127.0.0.1:7101/ViewController/faces/DeptEmpPage.js`. The page title is "DeptEmpPage.jsf". The Oracle logo is visible at the top left. The main content area is divided into two sections:

- Departments:** A summary view showing:
  - DepartmentId: 10
  - DepartmentName: Administration
  - ManagerId: 200
  - LocationId: 1700Navigation buttons: First, Previous, Next, Last.
- showDetailItem 1:** A form for displaying employee details:
  - \*EmployeeId: 200
  - FirstName: Jennifer
  - \*LastName: Whalen
  - \*Email: JWHALEN
  - \*Phone: 515.123.4444
  - \*HireDate: 17/Sep/2003 (highlighted with a callout box)
  - \*JobId: AD\_ASST
  - Salary: 4400

A callout box points to the HireDate field with the text: **Example: 29/Nov/1998**

9. Use the **Clock/Calendar** icon next to the field to display a pop-up calendar and choose a valid date.

The screenshot shows an Oracle APEX application window titled 'DeptEmpPage.jsf'. The browser address bar shows the URL '127.0.0.1:7101/ViewController/faces/DeptEmpPage.jsf'. The application has an 'ORACLE' logo at the top left.

On the left, there is a 'Departments' section with the following details:

- DepartmentId: 10
- DepartmentName: Administration
- ManagerId: 200
- LocationId: 1700

Below these details are four buttons: 'First', 'Previous', 'Next', and 'Last'.

On the right, there is a 'View' dropdown menu and a 'Detach' button. Below them is a table with the following data:

EmployeeId	FirstName	LastName
200	Jennifer	Whalen

Below the table is a 'showDetailItem 1' section with the following form fields:

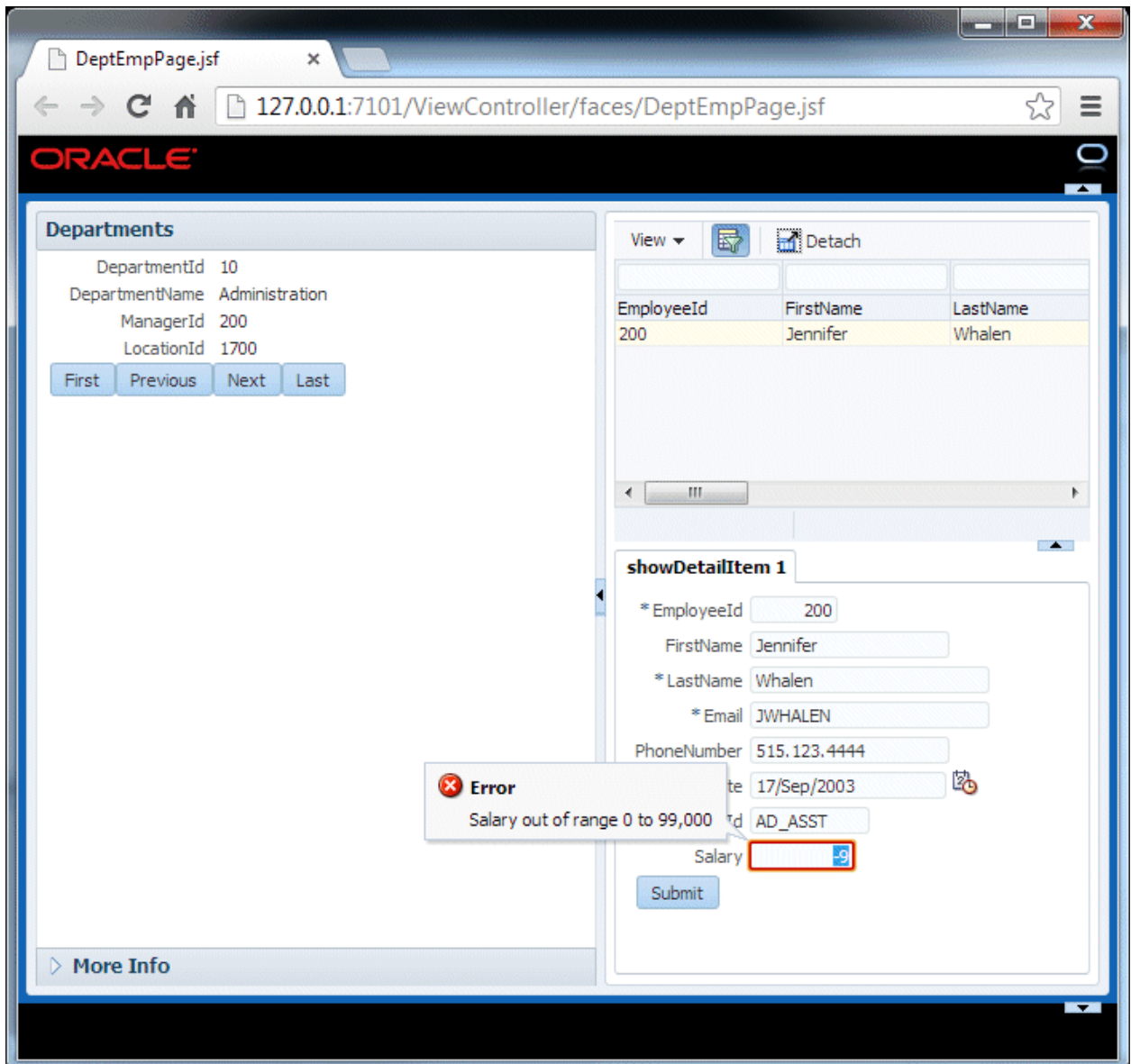
- \* EmployeeId: 200
- FirstName: Jennifer
- \* LastName: Whalen
- \* Email: JWHALEN
- PhoneNumber: 515.123.4444
- \* HireDate: 17/Sep/2003

A calendar is open over the 'HireDate' field, showing the month of April 2013. The calendar grid is as follows:

SUN	MON	TUE	WED	THU	FRI	SAT
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4

At the bottom left of the application window, there is a 'More Info' link. The status bar at the bottom shows the URL '127.0.0.1:7101/ViewController/faces/DeptEmpPage.jsf#'.

- Update the **Salary** field to a value that will break the validation you created earlier (e.g. -9) and click the **Submit** button to see the error message you created.



11. Browse to department **50** and notice that you can now sort the data in the employees table. Click on one of the column headers in the table to sort the data in the table.

View ▾ Detach

EmployeeId	FirstName	LastName	Email	PhoneNumber	HireDate
130	Mozhe	Atkinson	MATKINSO	650.124.6234	30/Oct/2005
192	Sarah	Bell		650.501.1876	04/Feb/2004
129	Laura	Bissot	LBISSOT	650.124.5234	20/Aug/2005
185	Alexis	Bull	ABULL	650.509.2876	20/Feb/2005
187	Anthony	Cabrio	ACABRIO	650.509.4876	07/Feb/2007
188	Kelly	Chung	KCHUNG	650.505.1876	14/Jun/2005
142	Curtis	Davies	CDAVIES	650.121.2994	29/Jan/2005
186	Julia	Dellinger	JDELLING	650.509.3876	24/Jun/2006
189	Jennifer	Dilly	JDILLY	650.505.2876	13/Aug/2005
193	Britney	Everett	BEVERETT	650.501.2876	03/Mar/2005
197	Kevin	Feeney	KFEENEY	650.507.9822	23/May/2006
181	Jean	Fleaur	JFLEAUR	650.507.9877	23/Feb/2006
121	Adam	Fripp	AFRIPP	650.123.2234	10/Apr/2005
190	Timothy	Gates	TGATES	650.505.3876	11/Jul/2006
135	Ki	Gee	KGEE	650.127.1734	12/Dec/2007
183	Girard	Geoni	GGEONI	650.507.9879	03/Feb/2008
199	Douglas	Grant	DGRANT	650.507.9844	13/Jan/2008

12. Click a column heading in the table and drag it to reposition it in the table.

View ▾ Detach

EmployeeId	FirstName	LastName	PhoneNumber	HireDate	JobId
130	Mozhe	Atkinson	650.124.6234	30/Oct/2005	ST_CLERK
192	Sarah	Bell	650.501.1876	04/Feb/2004	SH_CLERK
129	Laura	Bissot	650.124.5234	20/Aug/2005	ST_CLERK
185	Alexis	Bull	650.509.2876	20/Feb/2005	SH_CLERK
187	Anthony	Cabrio	650.509.4876	07/Feb/2007	SH_CLERK
188	Kelly	Chung	650.505.1876	14/Jun/2005	SH_CLERK
142	Curtis	Davies	650.121.2994	29/Jan/2005	ST_CLERK
186	Julia	Dellinger	650.509.3876	24/Jun/2006	SH_CLERK
189	Jennifer	Dilly	650.505.2876	13/Aug/2005	SH_CLERK
193	Britney	Everett	650.501.2876	03/Mar/2005	SH_CLERK
197	Kevin	Feeney	650.507.9822	23/May/2006	SH_CLERK
181	Jean	Fleaur	650.507.9877	23/Feb/2006	SH_CLERK
121	Adam	Fripp	650.123.2234	10/Apr/2005	ST_MAN
190	Timothy	Gates	650.505.3876	11/Jul/2006	SH_CLERK
135	Ki	Gee	650.127.1734	12/Dec/2007	ST_CLERK
183	Girard	Geoni	650.507.9879	03/Feb/2008	SH_CLERK
199	Douglas	Grant	650.507.9844	13/Jan/2008	SH_CLERK
105	...	...	650.501.1876	17/Mar/2007	SH_CLERK
106	...	...	650.501.1876	17/Mar/2007	SH_CLERK
107	...	...	650.501.1876	17/Mar/2007	SH_CLERK
108	...	...	650.501.1876	17/Mar/2007	SH_CLERK
109	...	...	650.501.1876	17/Mar/2007	SH_CLERK
110	...	...	650.501.1876	17/Mar/2007	SH_CLERK
111	...	...	650.501.1876	17/Mar/2007	SH_CLERK
112	...	...	650.501.1876	17/Mar/2007	SH_CLERK
113	...	...	650.501.1876	17/Mar/2007	SH_CLERK
114	...	...	650.501.1876	17/Mar/2007	SH_CLERK
115	...	...	650.501.1876	17/Mar/2007	SH_CLERK
116	...	...	650.501.1876	17/Mar/2007	SH_CLERK
117	...	...	650.501.1876	17/Mar/2007	SH_CLERK
118	...	...	650.501.1876	17/Mar/2007	SH_CLERK
119	...	...	650.501.1876	17/Mar/2007	SH_CLERK
120	...	...	650.501.1876	17/Mar/2007	SH_CLERK

The new column order looks as follows:



EmployeeId	Email	FirstName	LastName	PhoneNumber	HireDate	JobId
130	MATKINSO	Mozhe	Atkinson	650.124.6234	30/Oct/2005	ST_CLERK
192	SBELL	Sarah	Bell	650.501.1876	04/Feb/2004	SH_CLERK
129	LBISSOT	Laura	Bissot	650.124.5234	20/Aug/2005	ST_CLERK
185	ABULL	Alexis	Bull	650.509.2876	20/Feb/2005	SH_CLERK
187	ACABRIO	Anthony	Cabrio	650.509.4876	07/Feb/2007	SH_CLERK
188	KCHUNG	Kelly	Chung	650.505.1876	14/Jun/2005	SH_CLERK
142	CDAVIES	Curtis	Davies	650.121.2994	29/Jan/2005	ST_CLERK
186	JDELLING	Julia	Dellinger	650.509.3876	24/Jun/2006	SH_CLERK
189	JDILLY	Jennifer	Dilly	650.505.2876	13/Aug/2005	SH_CLERK
193	BEVERETT	Britney	Everett	650.501.2876	03/Mar/2005	SH_CLERK
197	KFEENEY	Kevin	Feeney	650.507.9822	23/May/2006	SH_CLERK
181	JFLEAUR	Jean	Fleaur	650.507.9877	23/Feb/2006	SH_CLERK
121	AFRIPP	Adam	Fripp	650.123.2234	10/Apr/2005	ST_MAN
190	TGATES	Timothy	Gates	650.505.3876	11/Jul/2006	SH_CLERK
135	KGEE	Ki	Gee	650.127.1734	12/Dec/2007	ST_CLERK
183	GGEONI	Girard	Geoni	650.507.9879	03/Feb/2008	SH_CLERK
199	DGRANT	Douglas	Grant	650.507.9844	13/Jan/2008	SH_CLERK
105	WJONES	Walter	Jones	650.501.1876	17/Mar/2007	SH_CLERK

13. In the Filter field above the **LastName** column, type **B%** and press **Enter** to filter the table to show only those employees whose surname begins with the letter B.

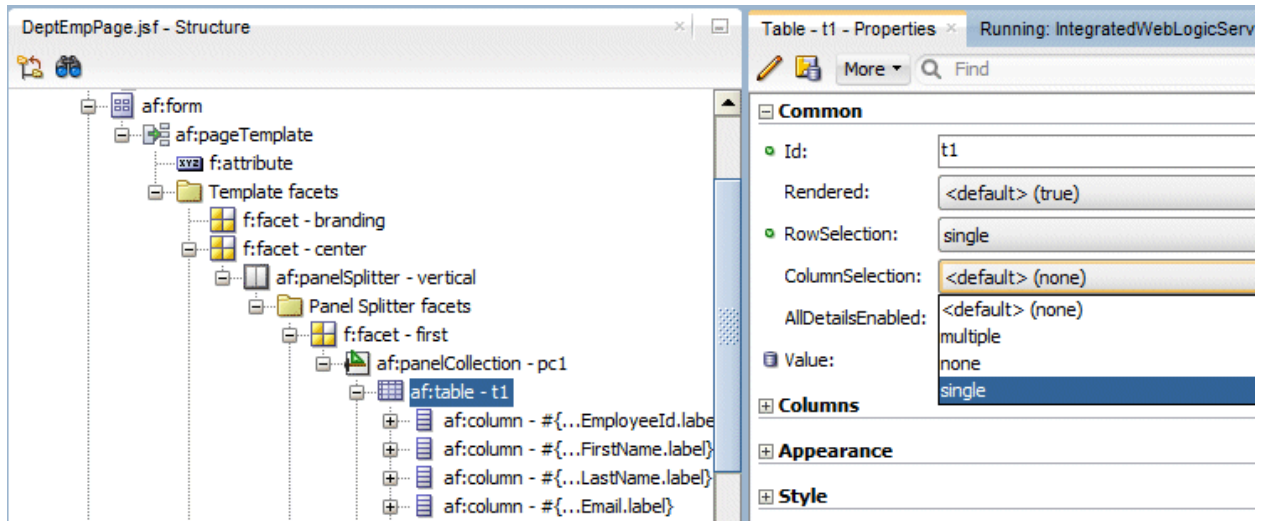
EmployeeId	Email	FirstName	LastName	PhoneNumber	HireDate	JobId
192	SBELL	Sarah	Bell	650.501.1876	04/Feb/2004	SH_CLERK
129	LBISSOT	Laura	Bissot	650.124.5234	20/Aug/2005	ST_CLERK
185	ABULL	Alexis	Bull	650.509.2876	20/Feb/2005	SH_CLERK

14. Experiment with the other menus and buttons of the table to see additional functionality. Once you are finished, close the browser and return to JDeveloper.

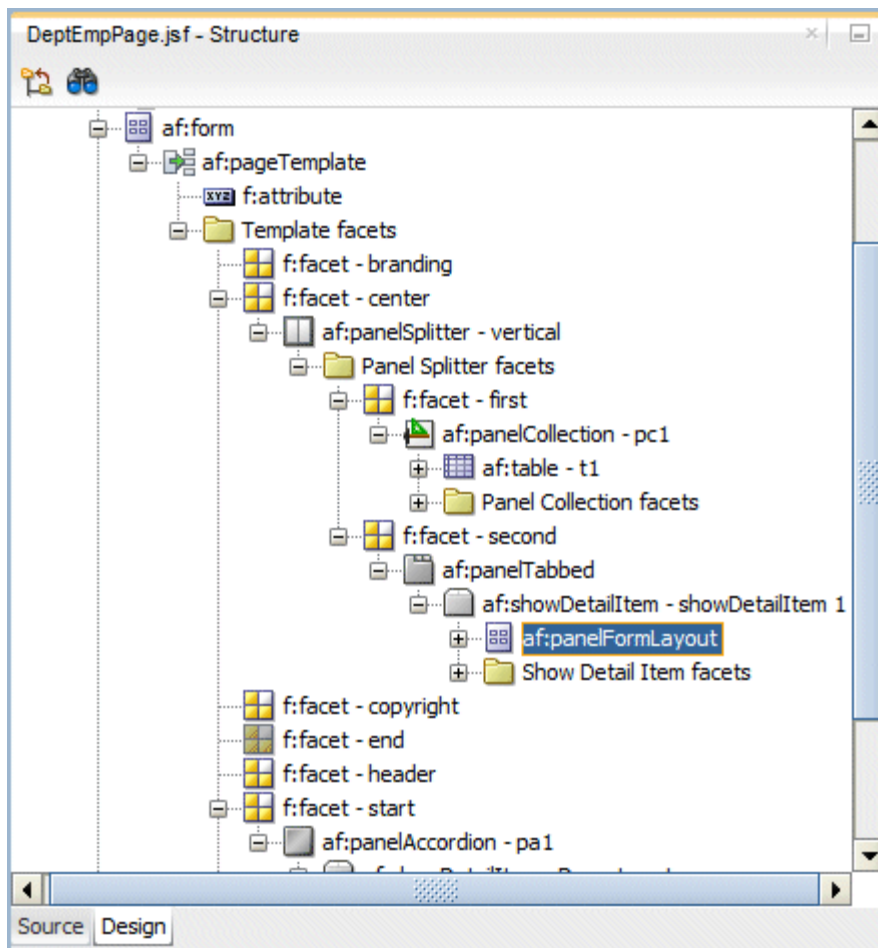
#### Step 4: Enhance the JSF Page

In the next steps you enhance the user interface by modifying the table of employees to add the ability to do column selection, by reordering the fields in the Employee details tab, and by binding the business components to a graph representation. You do this with simple drag and drop operations - behind the scenes the ADF Model layer handles the data binding for you.

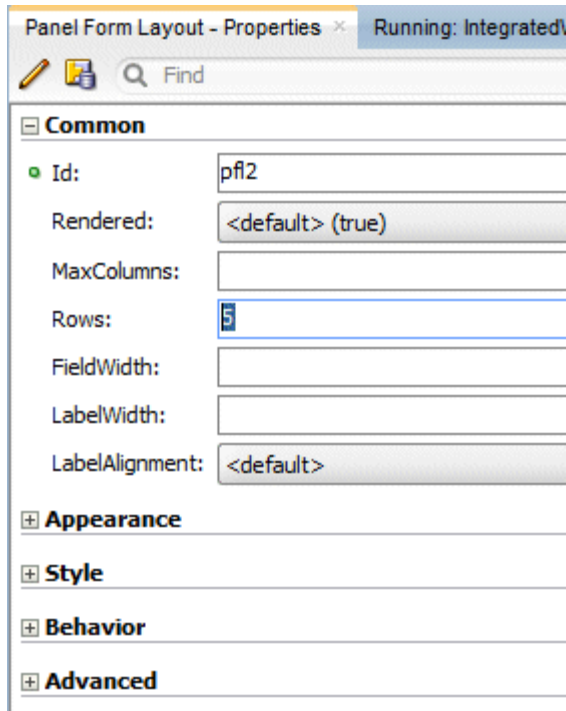
1. Click the **DeptEmpPage.jsf** tab to return to the page in the Design Editor. Select the **Employees** (t1) table in the Design Editor or in the Structure window. Set the **Column Selection** property of the table to **Single**.



2. In the Structure window select the **panelFormLayout** that surrounds the employee details fields in the showDetailItem tab.



3. In the Properties window set the **Rows** property to 5.

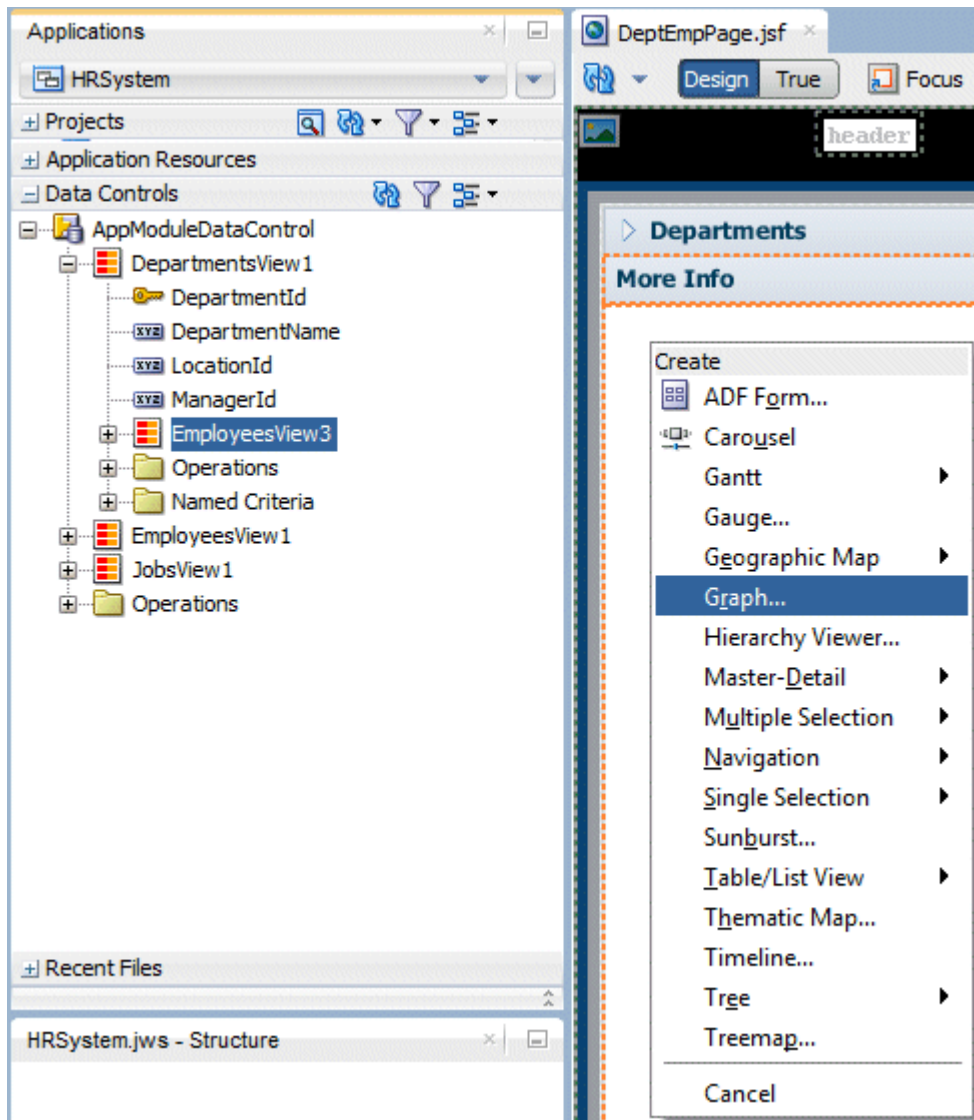


When you run the page in a later step you will see that the fields in the tab are now arranged in two columns.

Notice that you can control layout of the objects within a layout container by setting the properties of the container.

Next you add a graph representation of the data to the page; this is done using the ADF Data Visualization set of JSF components.

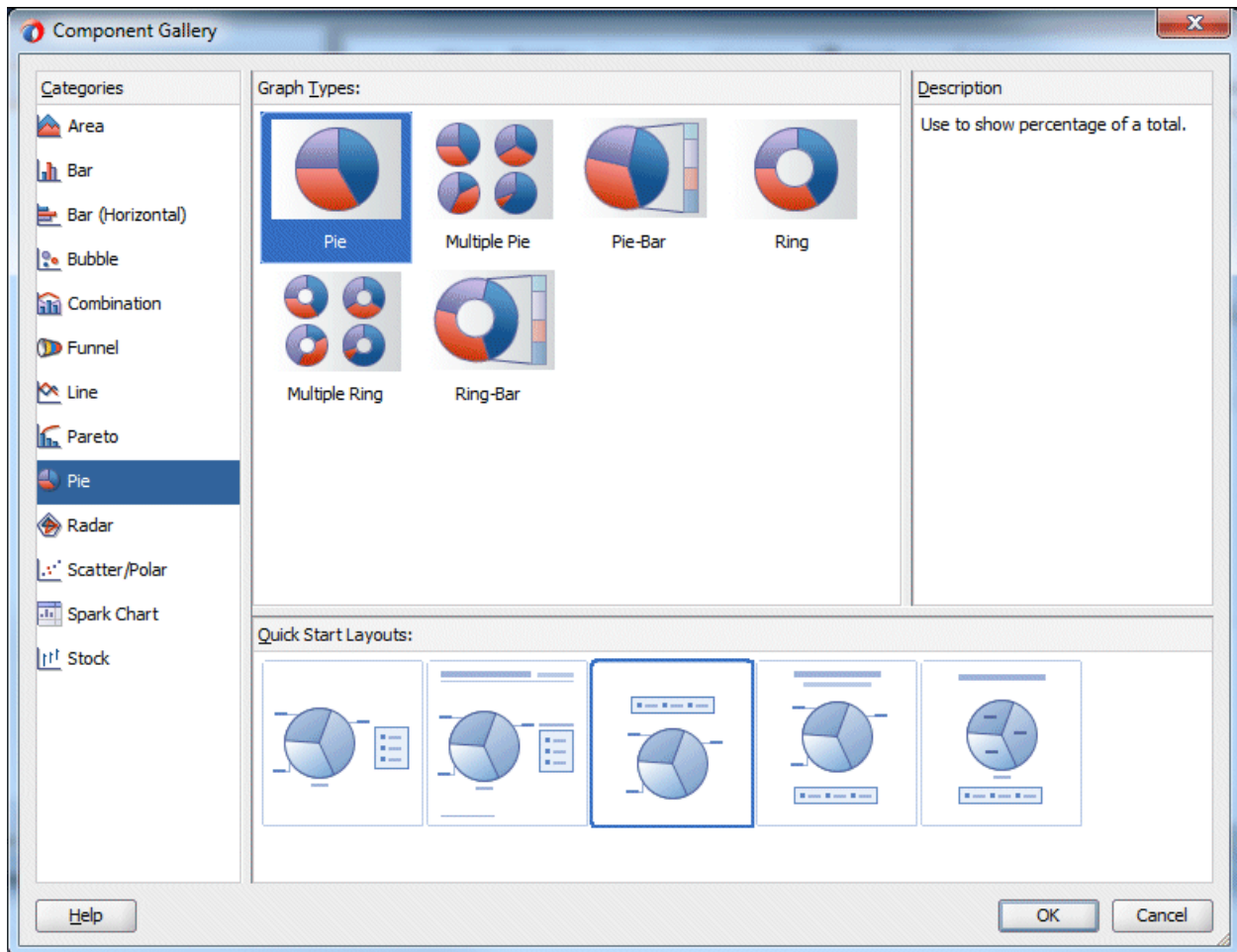
4. In the Design Editor click the **More Info** accordion to expand it. Then drag the **EmployeesView3** data control into the **More Info** accordion and create it as a **Graph**.



The graph can be displayed as static PNG image or an interactive Flash component.

You control these behaviors through the properties of the graph component.

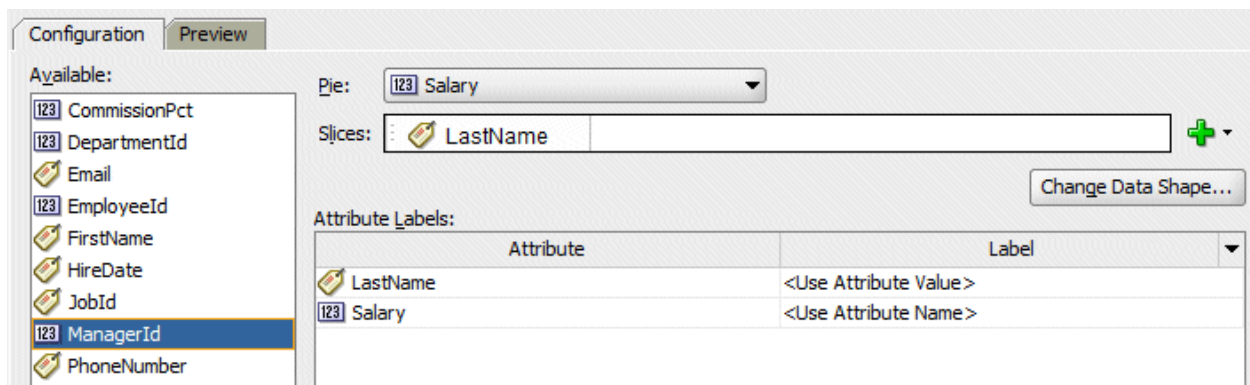
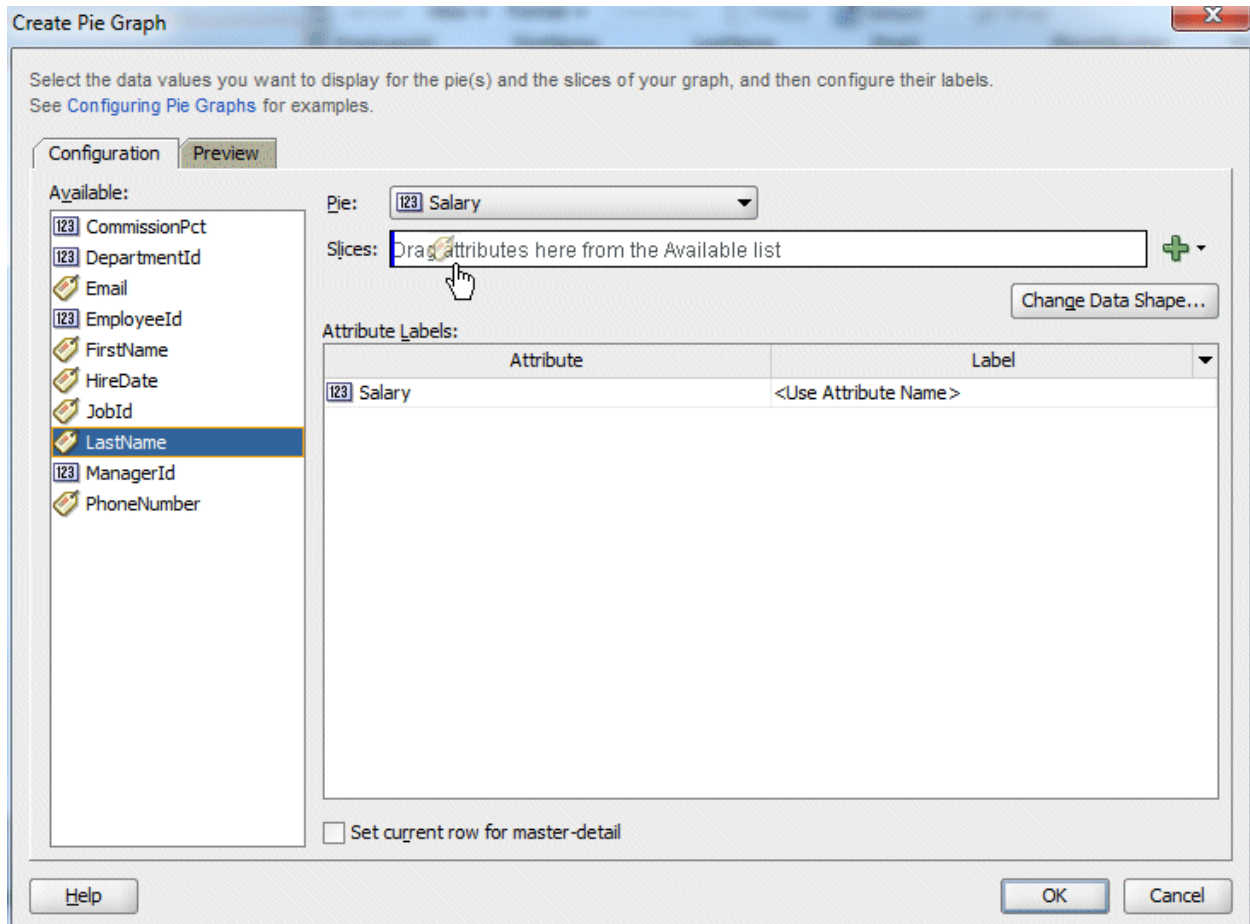
5. In the Component Gallery choose **Pie** from the list of graphs on the left and **Pie** again as the graph type. In the bottom pane, select the **third** Quick Start Layout.




Click the **OK**.

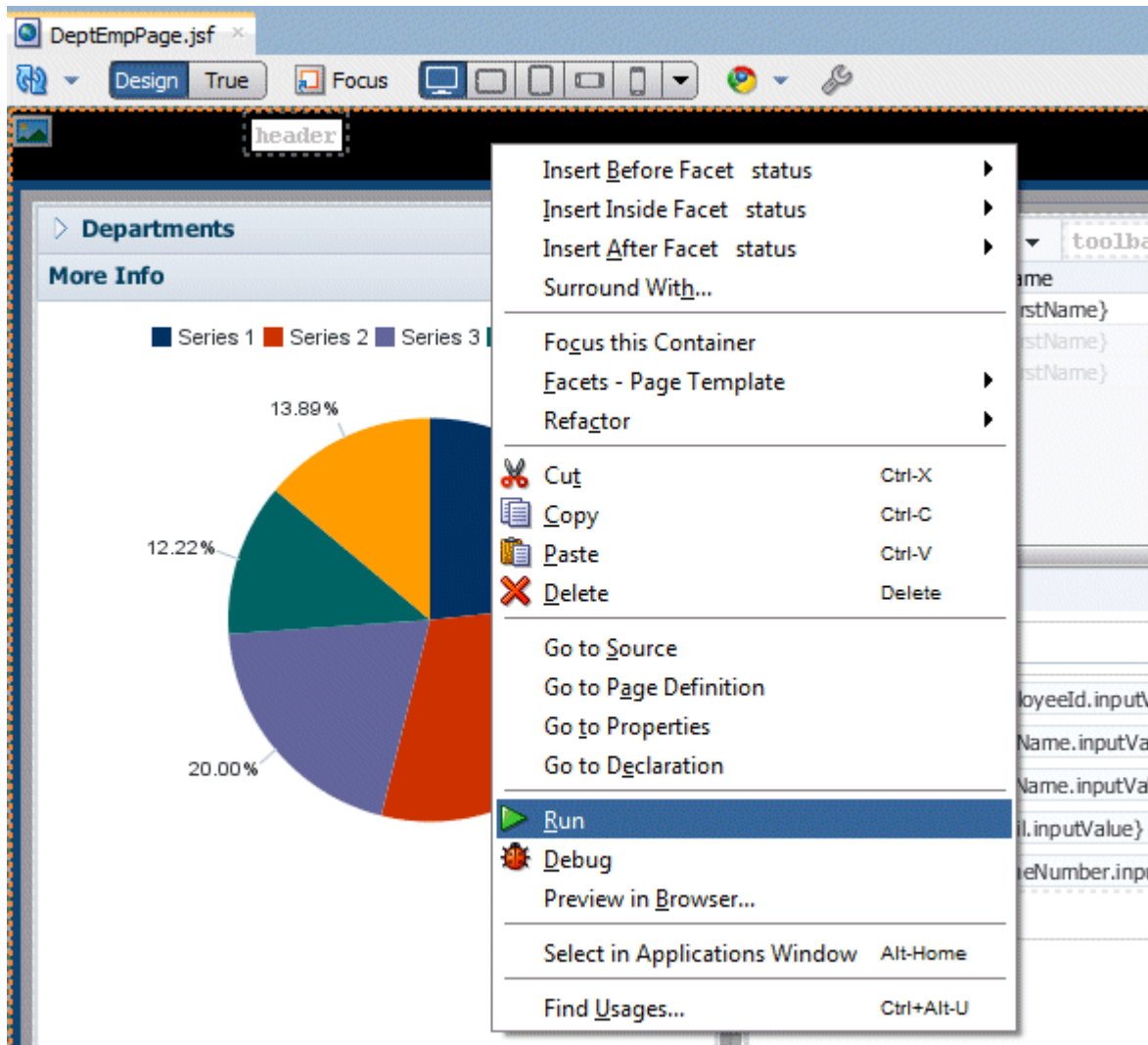
6. In the Create Pie Graph dialog select **Salary** in the Pie drop down and then, drag and drop **LastName** into the Slices field. Click **OK**.





Click **OK**.

- Click the **Save All**  icon on the JDeveloper menu bar to save your work and then right-click within the page and select **Run** from the context menu.



- Browse to department **50**. Because you set the column selection to single, note the additional behaviors for the table that are available once you choose a column - such as **Freeze** and **Wrap**.

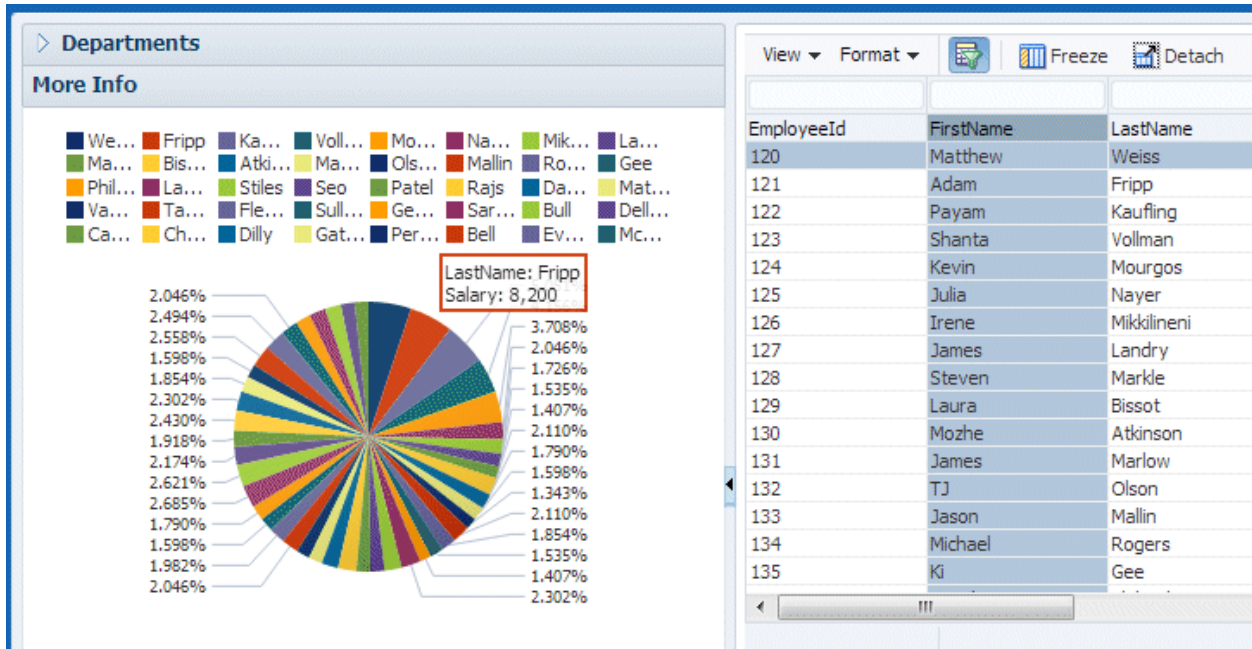
Select a column, then click the **Freeze** button. Now use the horizontal scroll bar to view the right-most columns. The Columns Frozen value is the number of columns (inclusive) to the left of the frozen column.

EmployeeId	FirstName	LastName	Email	PhoneNumber	HireDate	JobId
120	Matthew	Weiss	MWEISS	650.123.1234	18/Jul/2004	ST_J
121	Adam	Fripp	AFRIPP	650.123.2234	10/Apr/2005	ST_J
122	Payam	Kaufing	PKAUFLIN	650.123.3234	01/May/2003	ST_J
123	Shanta	Vollman	SVOLLMAN	650.123.4234	10/Oct/2005	ST_J
124	Kevin	Mourgos	KMOURGOS	650.123.5234	16/Nov/2007	ST_J
125	Julia	Nayer	JNAYER	650.124.1214	16/Jul/2005	ST_J
126	Irene	Mikkilineni	IMIKKILI	650.124.1224	28/Sep/2006	ST_J
127	James	Landry	JLANDRY	650.124.1334	14/Jan/2007	ST_J
128	Steven	Markle	SMARKLE	650.124.1434	08/Mar/2008	ST_J
129	Laura	Bissot	LBISSOT	650.124.5234	20/Aug/2005	ST_J
130	Mozhe	Atkinson	MATKINSO	650.124.6234	30/Oct/2005	ST_J
131	James	Marlow	JAMRLOW	650.124.7234	16/Feb/2005	ST_J
132	TJ	Olson	TJOLSON	650.124.8234	10/Apr/2007	ST_J
133	Jason	Mallin	JMALLIN	650.127.1934	14/Jun/2004	ST_J
134	Michael	Rogers	MROGERS	650.127.1834	26/Aug/2006	ST_J
135	Ki	Gee	KGEE	650.127.1734	12/Dec/2007	ST_J

EmployeeId	FirstName	LastName	JobId	Salary	CommissionPct	Ma
120	Matthew	Weiss	ST_MAN	8000		100
121	Adam	Fripp	ST_MAN	8200		100
122	Payam	Kaufing	ST_MAN	7900		100
123	Shanta	Vollman	ST_MAN	6500		100
124	Kevin	Mourgos	ST_MAN	5800		100
125	Julia	Nayer	ST_CLERK	3200		120
126	Irene	Mikkilineni	ST_CLERK	2700		120
127	James	Landry	ST_CLERK	2400		120
128	Steven	Markle	ST_CLERK	2200		120
129	Laura	Bissot	ST_CLERK	3300		120
130	Mozhe	Atkinson	ST_CLERK	2800		120
131	James	Marlow	ST_CLERK	2500		120
132	TJ	Olson	ST_CLERK	2100		120
133	Jason	Mallin	ST_CLERK	3300		120
134	Michael	Rogers	ST_CLERK	2900		120
135	Ki	Gee	ST_CLERK	2400		120

Columns Frozen 3

- In department 50 expand the **More Info** accordion and notice the graph and the popup that displays when you hover over it with your mouse. Since we plotted salary by last name, each slice of the pie represents the salary amount relative to the other employees. In this case Weiss, Fripp and Kaufing make the most.



10. Scroll over to the **showDetailItem1** tab, and notice how the fields have been reordered into two columns.

The 'showDetailItem1' tab displays a form for employee details. The fields are arranged in two columns:

* EmployeeId	120	* HireDate	18/Jul/2004
FirstName	Matthew	* JobId	ST_MAN
* LastName	Weiss	Salary	8000
* Email	MWEISS		
PhoneNumber	650.123.1234		

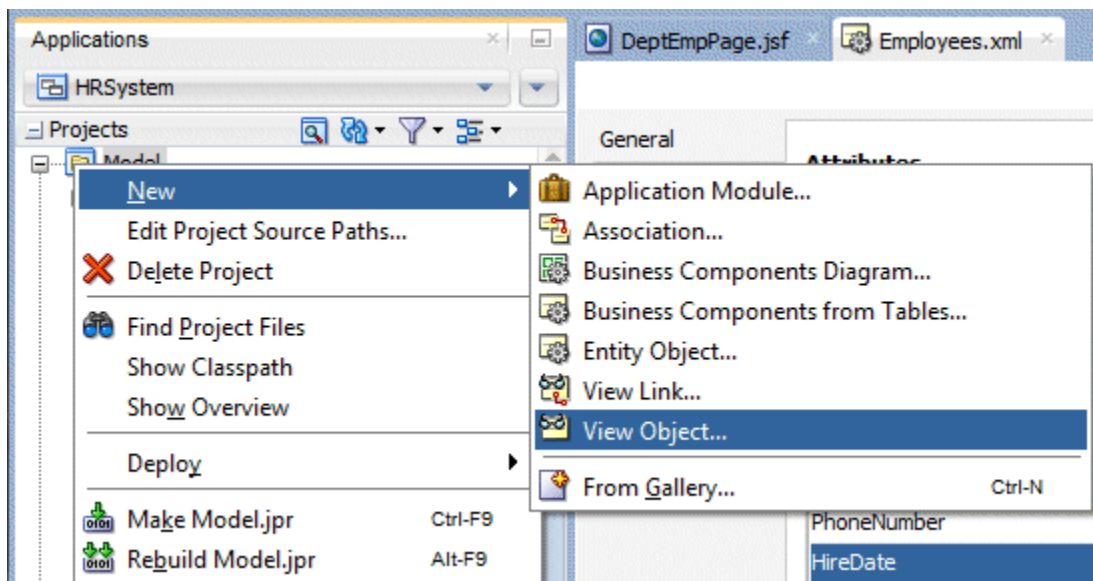
A 'Submit' button is located at the bottom left of the form.

When you are done, close the browser and return to JDeveloper.

## Step 5: Add More Complexity to the Business Services

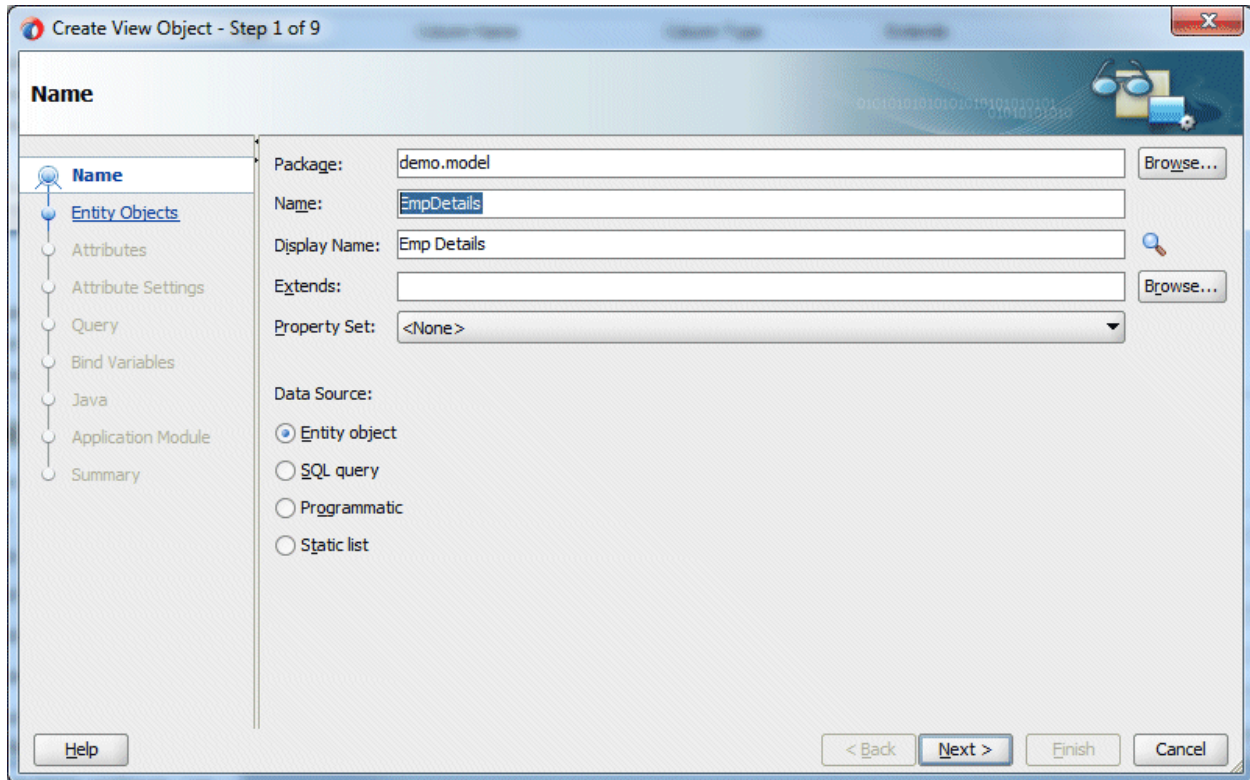
In this step you add a new business service to the application to display a subset of the fields from the Employees table, along with the department name from the Departments table. You also add a field that calculates the annual salary of an employee and a list of values field for department names. Oracle ADF Business Components allows you to create such complex updatable components in a declarative way.

1. In the Applications window right-click the **demo.model** package and choose **New View Object**.



2. In the Name page of the Create View Object wizard, type **EmpDetails** as the Name and retain **Entity object** as the default data source.

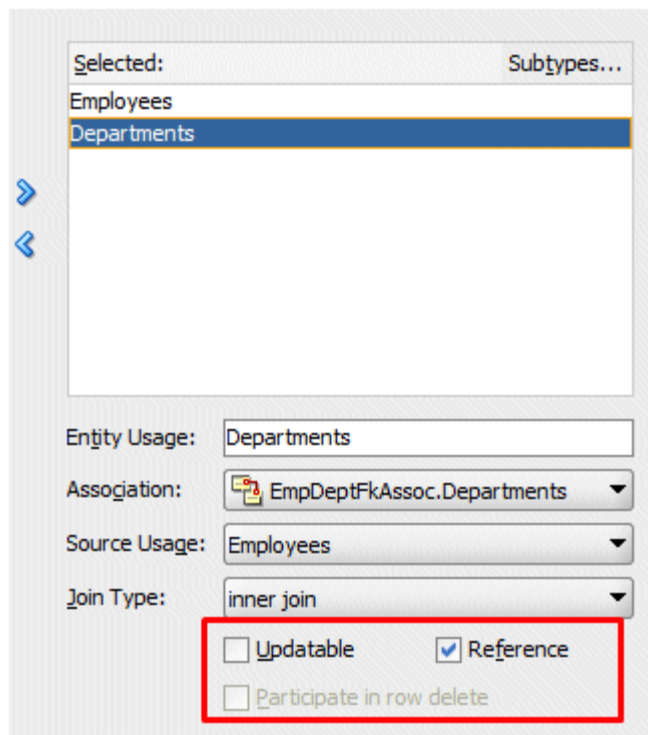
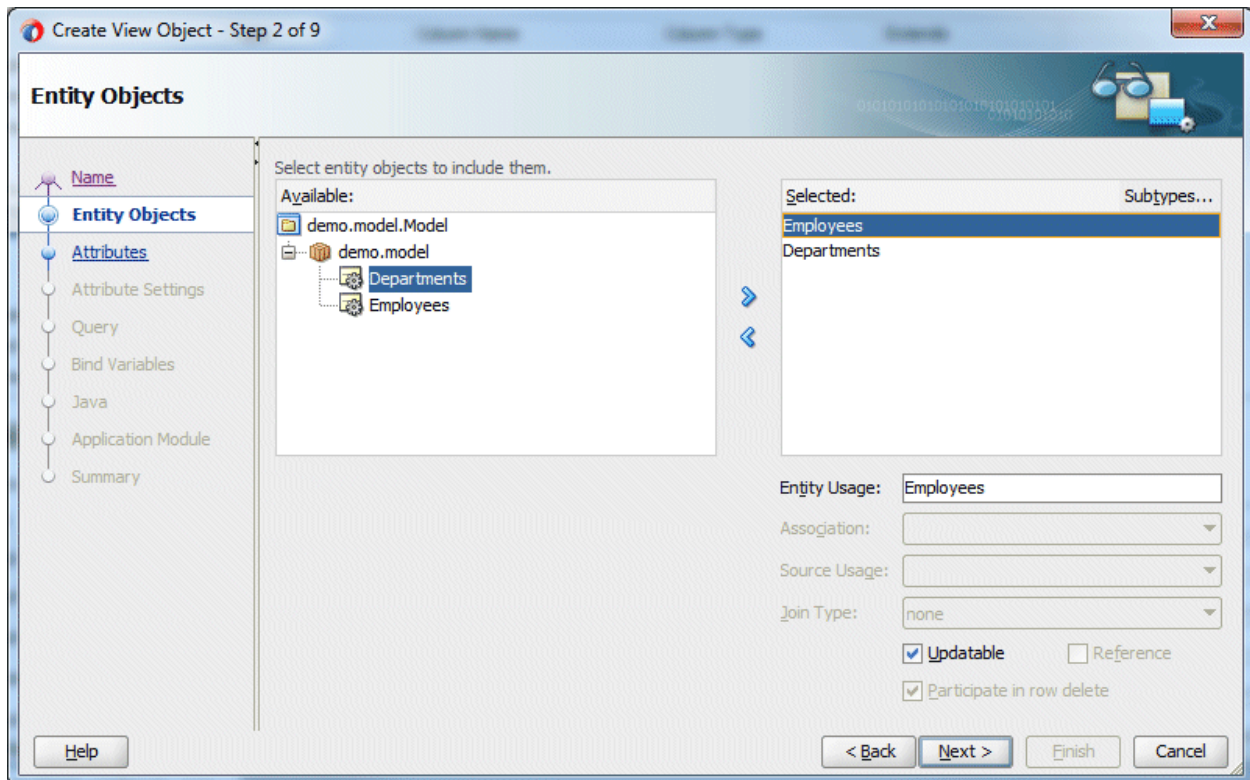




Click **Next**.

3. In the Entity Objects page of the wizard first select **Employees** and shuttle it to the right, and then select the **Departments** entity and add it also to the Selected pane.

Note that only the Employees entity has the updatable checkbox checked - the departments entity is used to read data only.

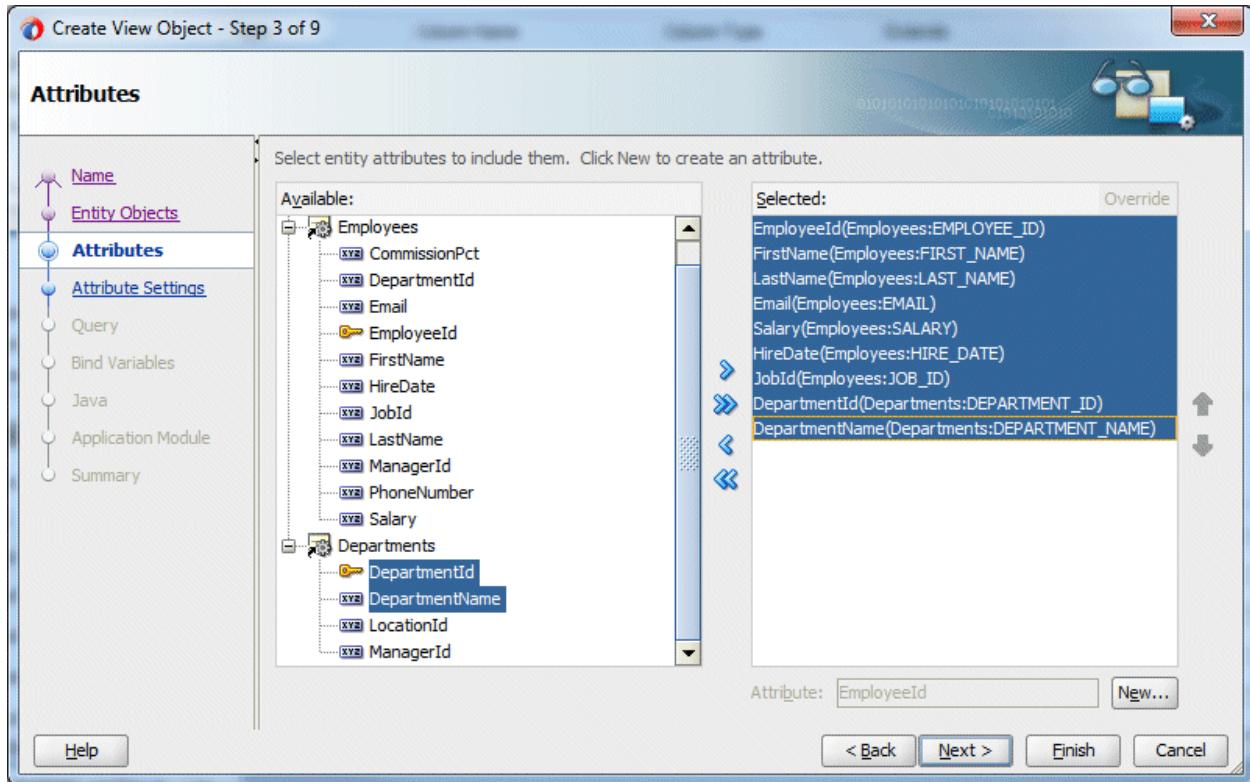


Click **Next**.

4. In the Attributes page of the wizard, shuttle the following **Employees** attributes into the Selected pane:

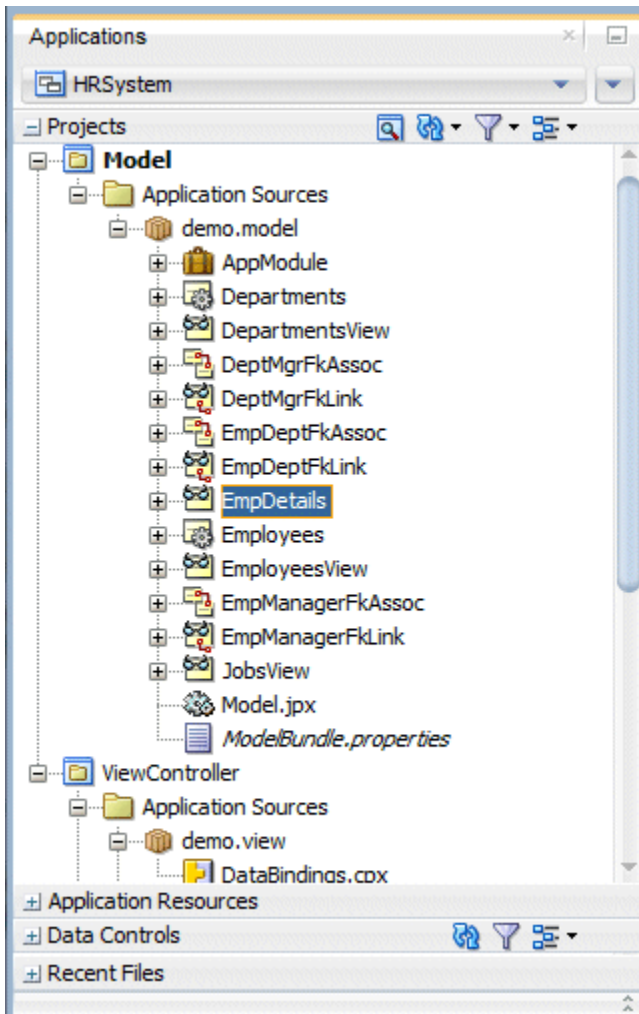
**EmployeeId, FirstName, LastName, Email, Salary, HireDate, JobId.** The order you select the attributes will become the order they're displayed.

And from the **Departments** entity, shuttle the following attributes into the Selected pane:  
**DepartmentId, DepartmentName.**

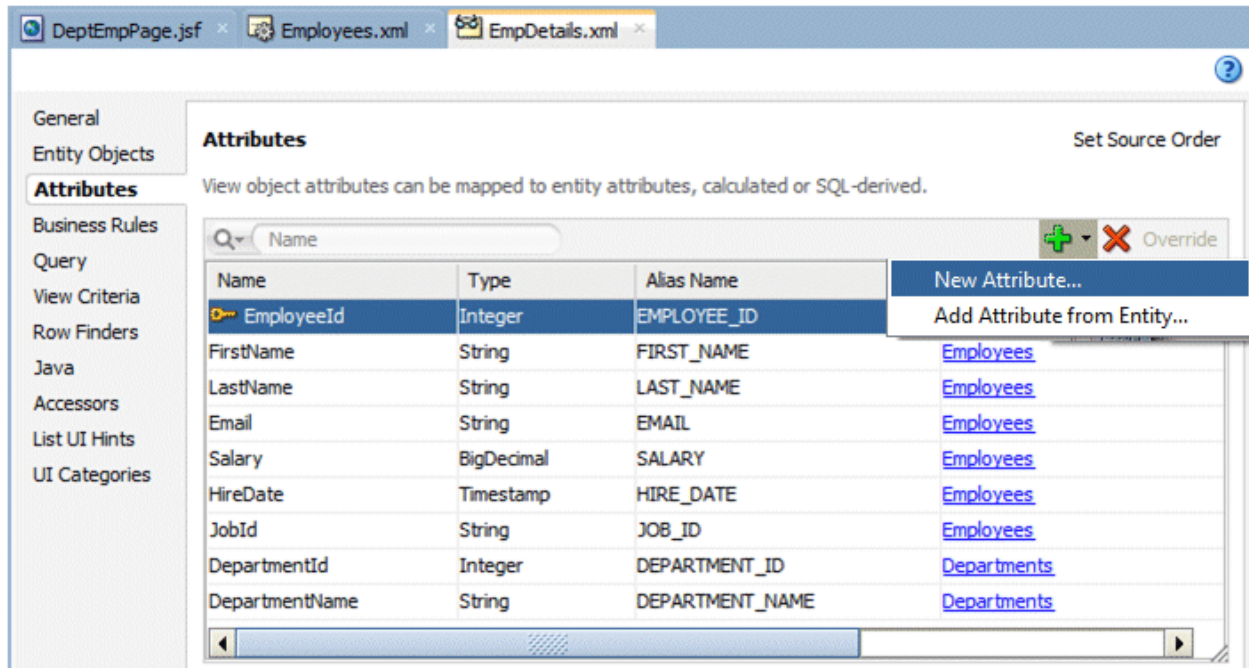


Click **Finish** to complete the view object creation.

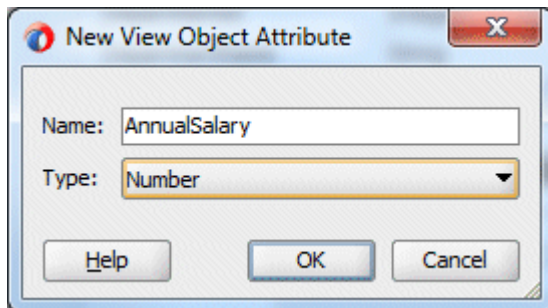
5. Next you add a calculated attribute to the view object, that displays the employee's annual salary. If it is not already open, double-click **EmpDetails** in the Applications Navigator.



6. Click the **Attributes** finger tab and from the drop down list select **New Attribute...**



- In the New View Object Attribute dialog type **AnnualSalary** as the name for the new attribute and **Number** as the Type, then click **OK**.



- In the Details tab at the bottom of the attributes list, set the following properties for the AnnualSalary attribute:

Field	Value
Name	<b>AnnualSalary</b>
Type	<b>Number</b>



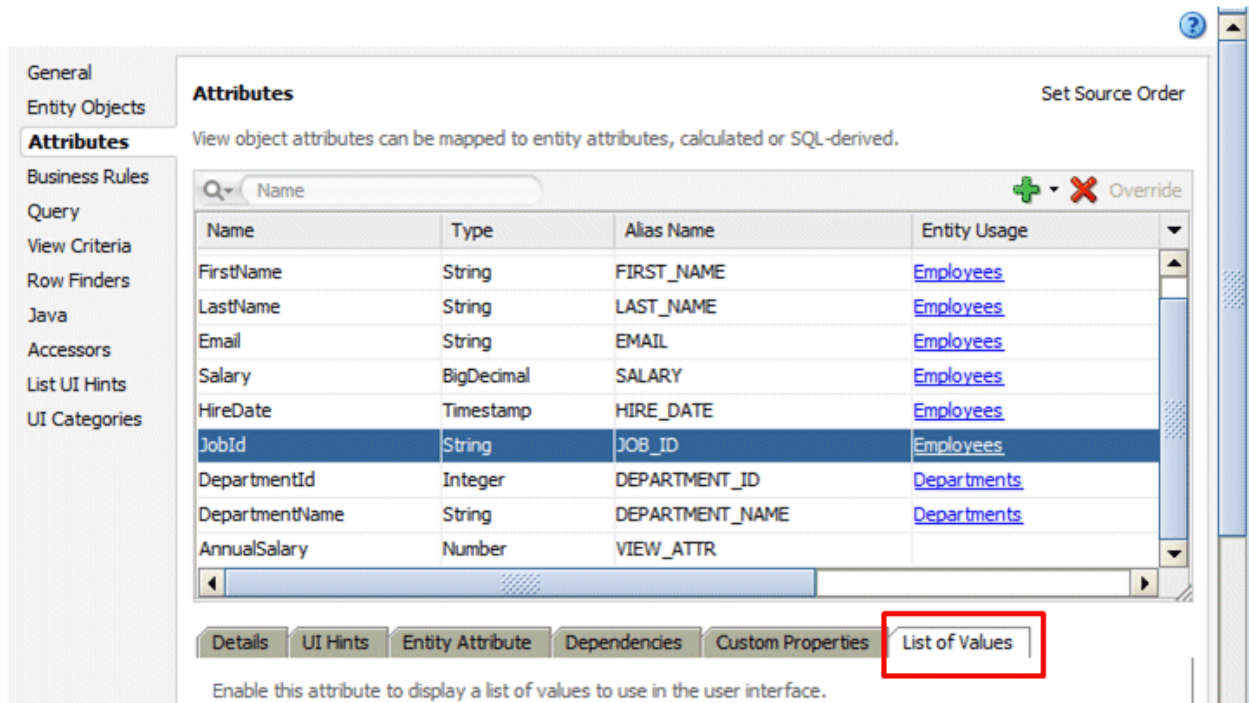
Default Value	<b>Expression (Checked)</b>
Value	<b>(Salary = null ? 0 : Salary) * 12</b>

The screenshot shows the 'EmpDetails.xml' editor interface. At the top, a table lists attributes: Email (String, EMAIL), Salary (BigDecimal, SALARY), HireDate (Timestamp, HIRE\_DATE), JobId (String, JOB\_ID), DepartmentId (Integer, DEPARTMENT\_ID), DepartmentName (String, DEPARTMENT\_NAME), and AnnualSalary (Number, VIEW\_ATTR). The 'AnnualSalary' attribute is selected and highlighted in blue.

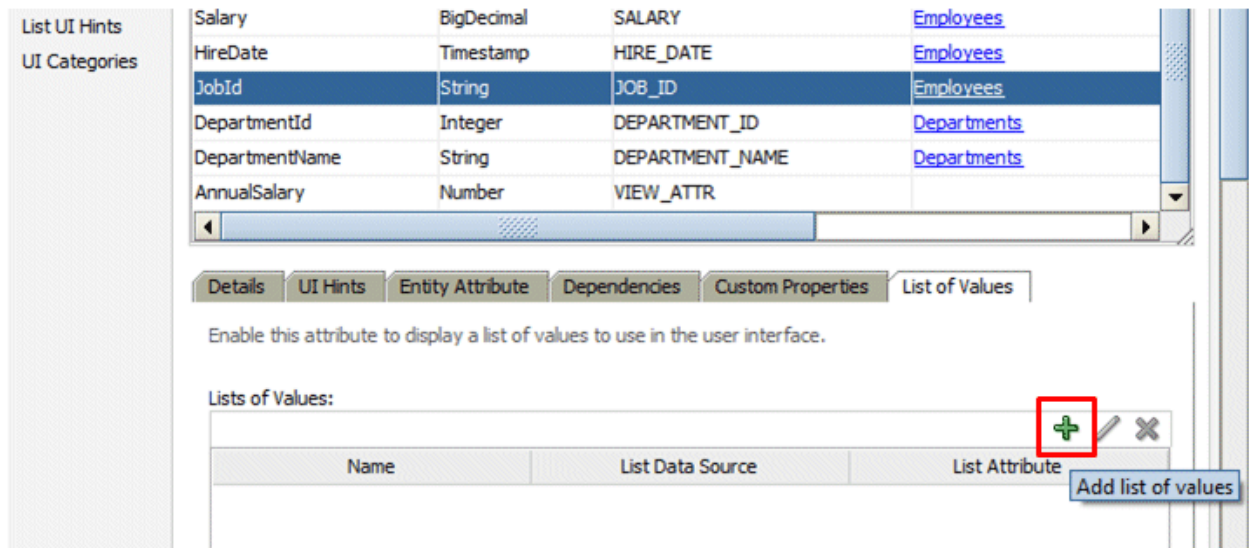
Below the table, the 'AnnualSalary' attribute configuration is shown in the 'UI Hints' tab. The configuration includes:

- Name: AnnualSalary
- Display Name: Annual Salary
- Description: (empty)
- Type: Number
- Property Set: <None>
- Polymorphic Discriminator:
- View/Entity:  View
- Subtype Value: (empty)
- Effective Date: (empty)
- Start Date/End Date: (empty)
- Service: (empty)
- SDO Property:
- Updatable: Never
- Persistent/Transient:  Persistent,  Transient
- Mandatory/Key Attribute:  Mandatory,  Key Attribute
- Selected in Query/Queryable:  Selected in Query,  Queryable
- Passivate:
- Alias: VIEW\_ATTR
- Type: NUMBER
- Default Value:  Literal,  Expression,  SQL
- Expression: (Salary == null ? 0 : Salary) \* 12
- Refresh Expression Value: true

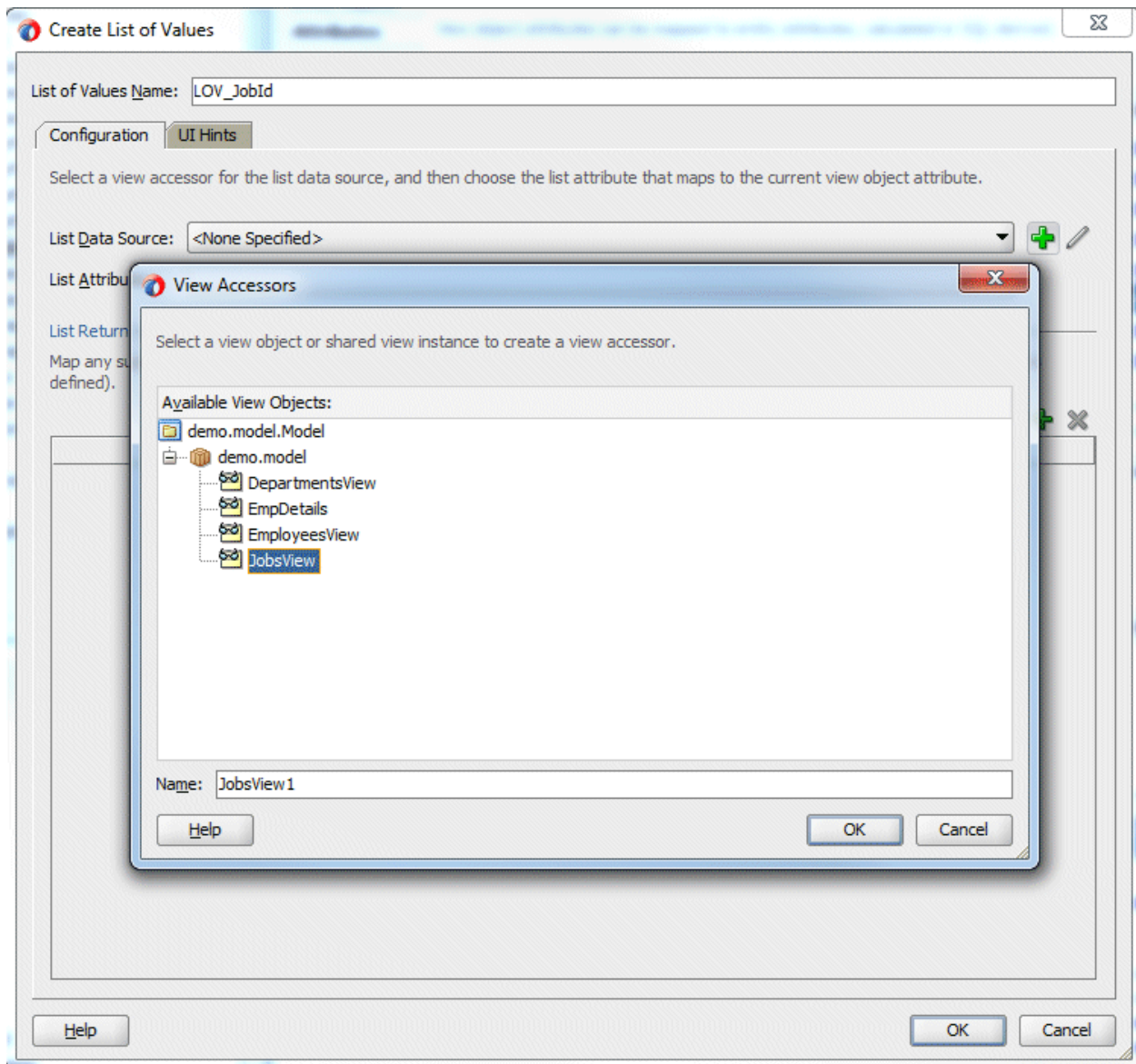
- 9.
10. In the next steps you add a list of values to a field based on a set of values stored in another table. In this case you create a list of values for possible job titles based on the Jobs table.
11. In the **EmpDetails.xml** editor click the **Attributes** tab and select the **JobId** attribute. Click the **List Of Values** tab at the bottom of the attributes list.



12. In the List of Values tab click the **Add** button to create a list of values.

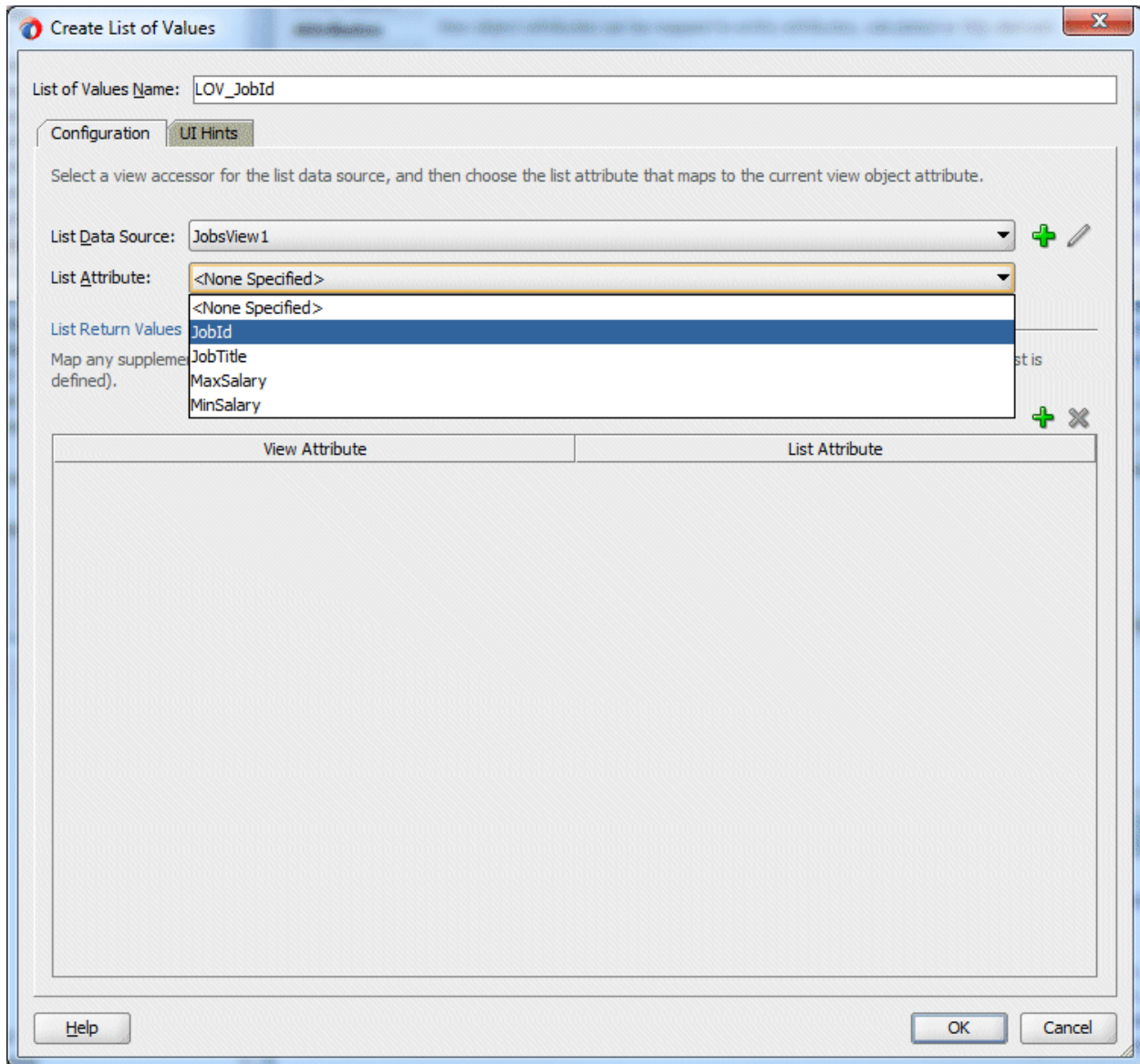


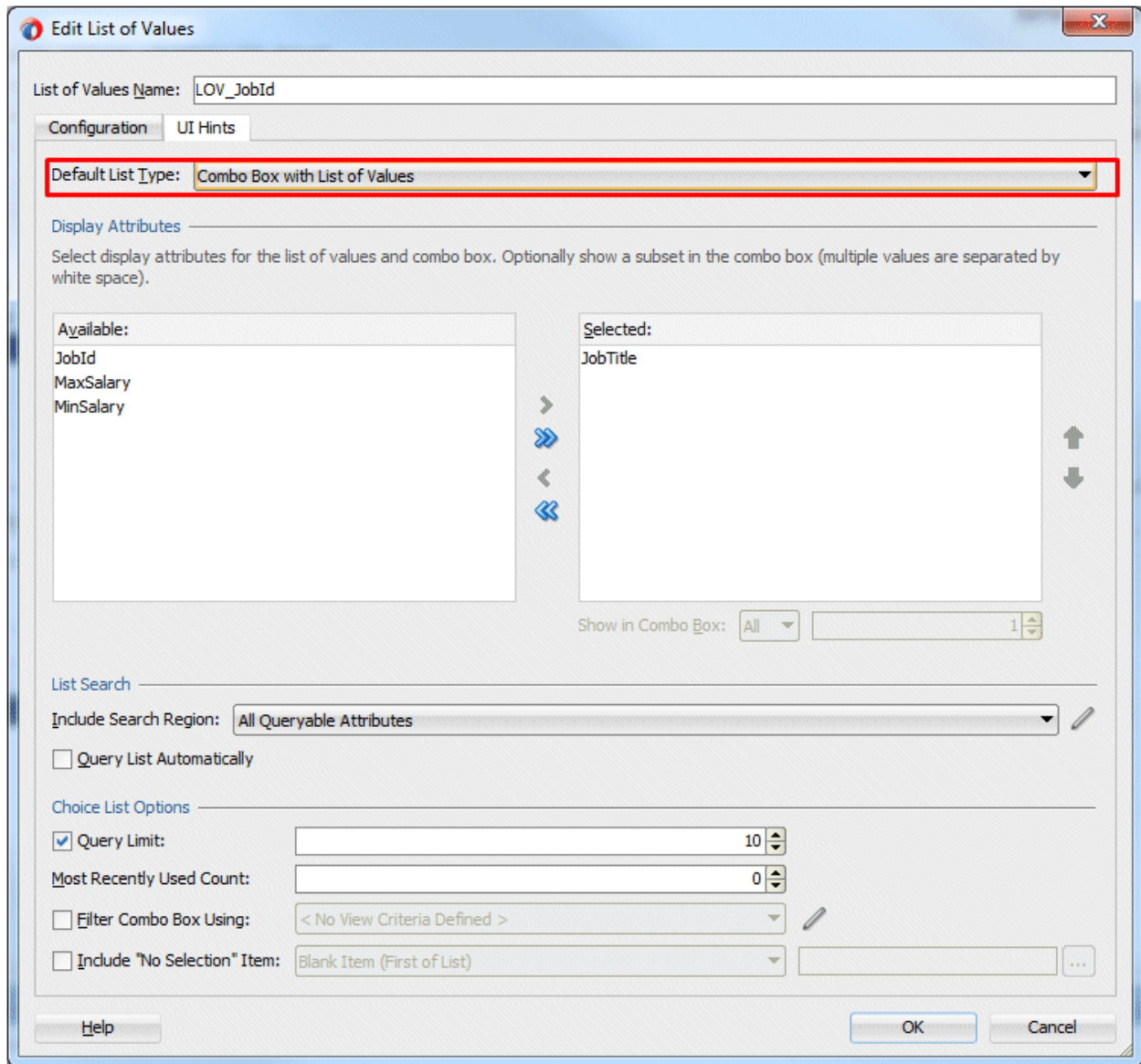
13. In the Create List of Values dialog click the green plus to the right of the **List Data Source** field. In the View Accessors dialog, select **JobsView**.



Click **OK**.

14. Back in the Create List of Values dialog, choose **JobId** as the List Attribute.



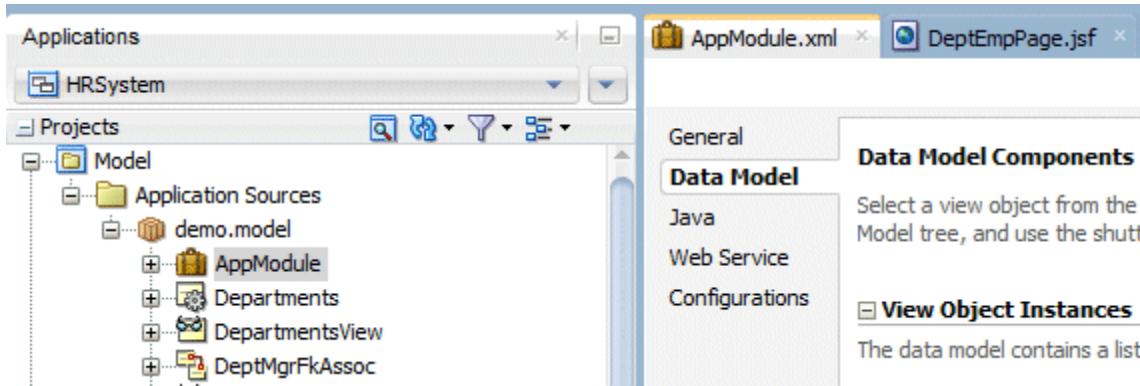


16. Click **OK** and save your work.

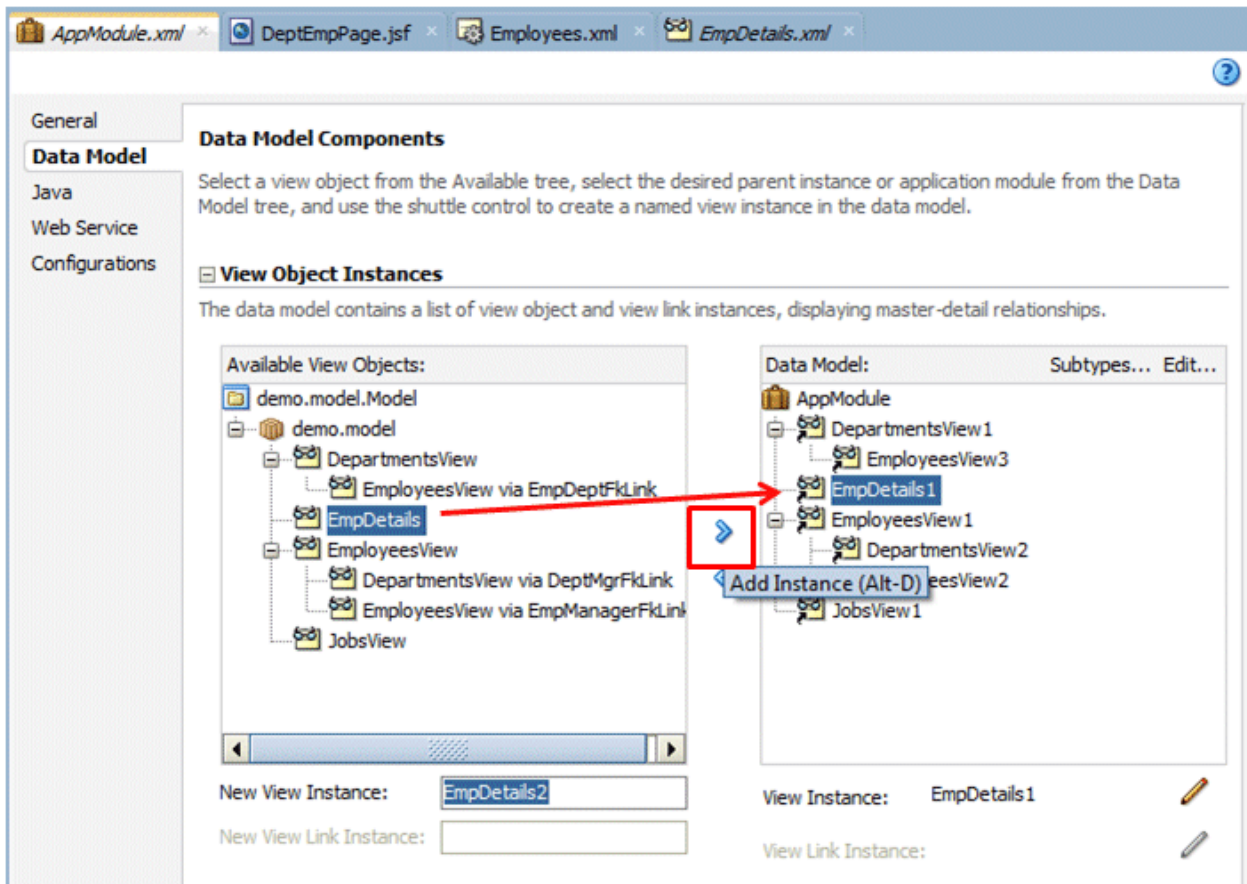
Now you add the new object that you've created to the data model.

17. In the Applications window double-click **AppModule** to open it for editing and choose the **Data Model** tab.

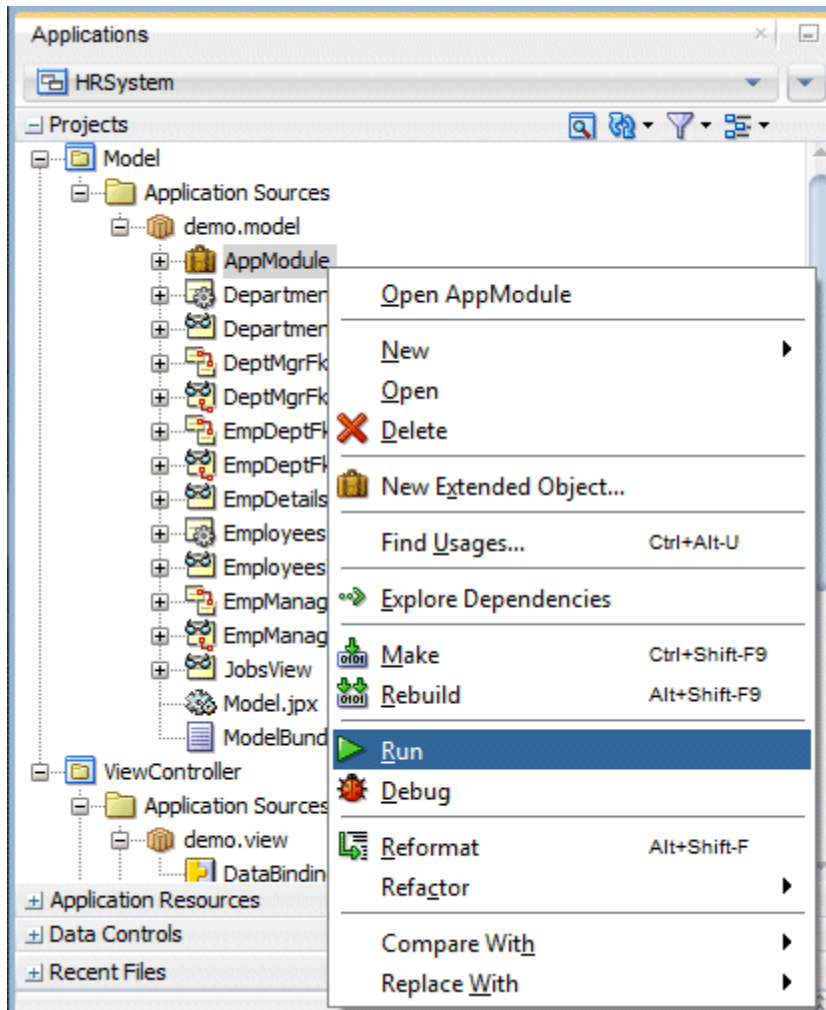




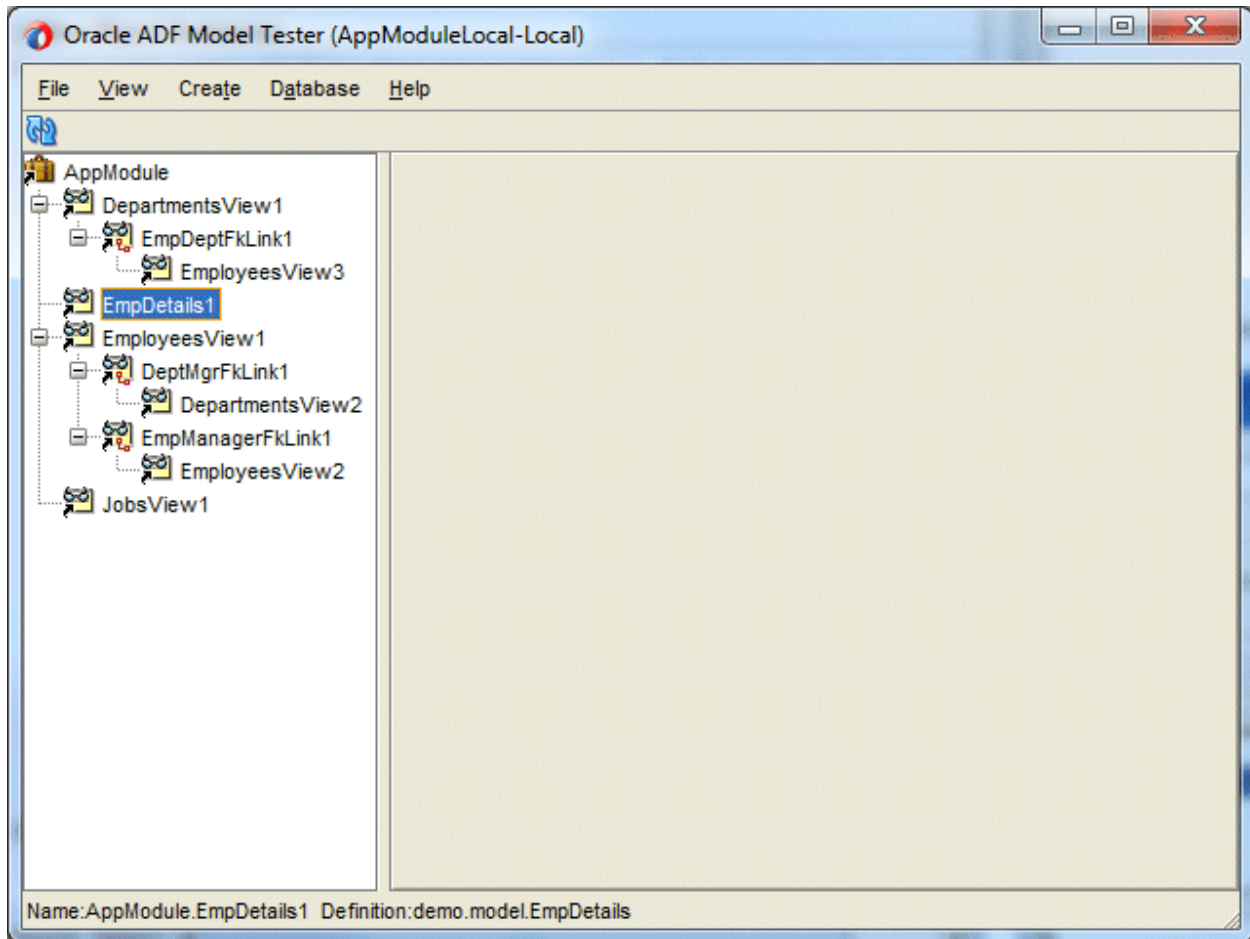
- Click the **EmpDetails** view object and shuttle it into the **Data Model** pane. Notice how the name for the new instance is **EmpDetails1**.  
**Save** your work.



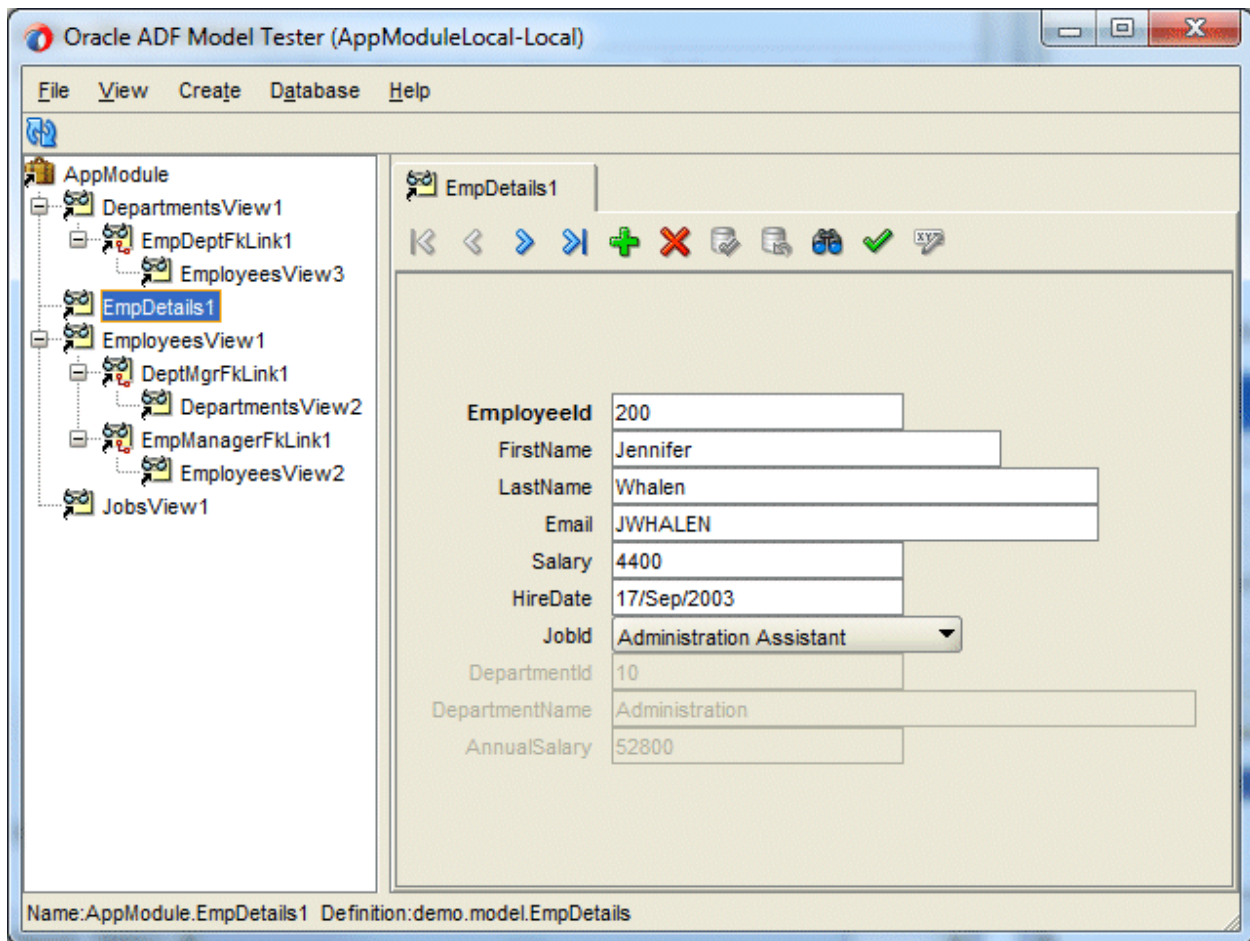
- Use the Business Component Browser to test the new functionality you have just incorporated through the **EmpDetails** view object. In the Applications window right-click **AppModule** and select **Run** to invoke the tester.



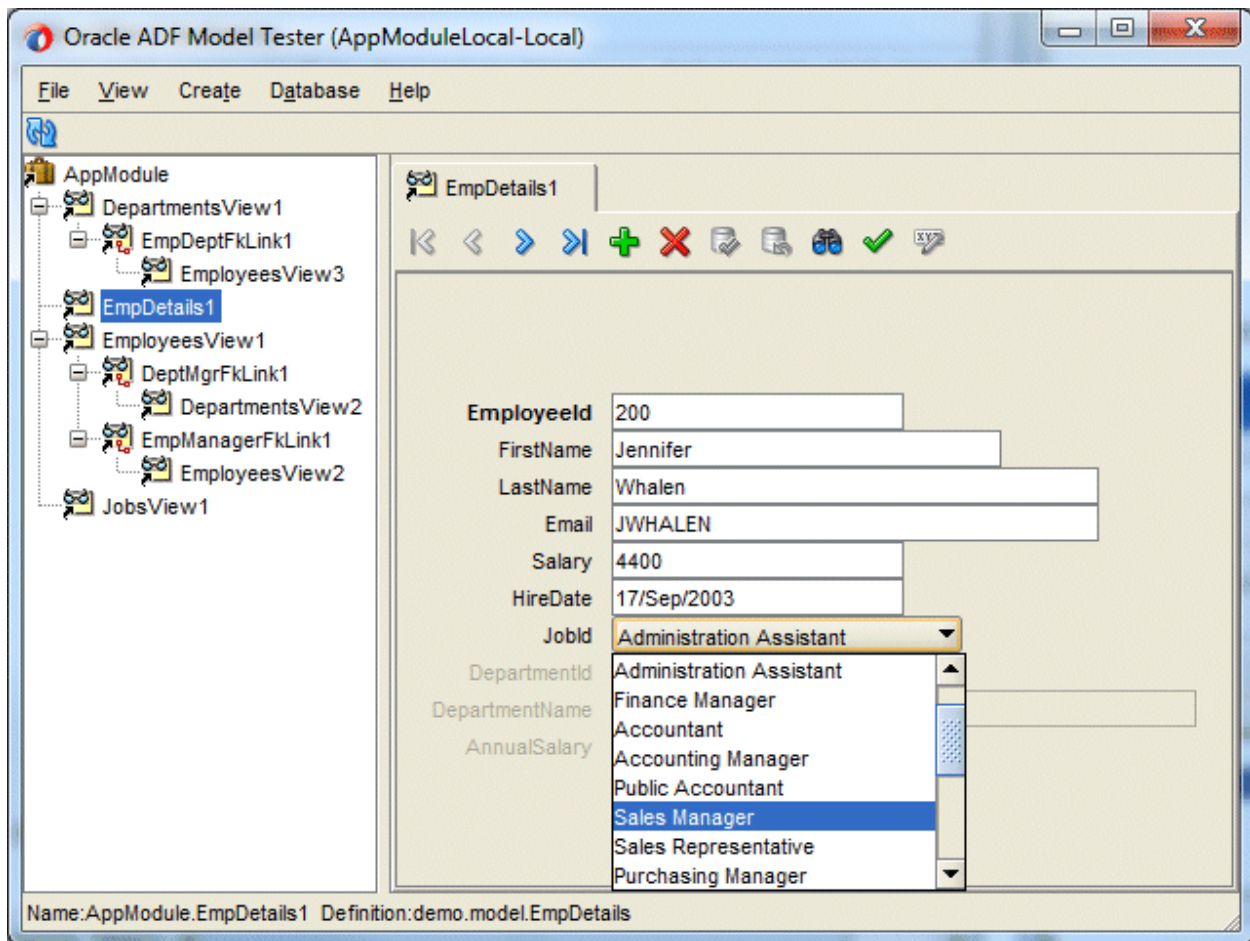
20. Double-click the **EmpDetails1** view object.



21. Notice that the new calculated AnnualSalary field is displayed, along with DepartmentId and DepartmentName. None of the them are updatable here.

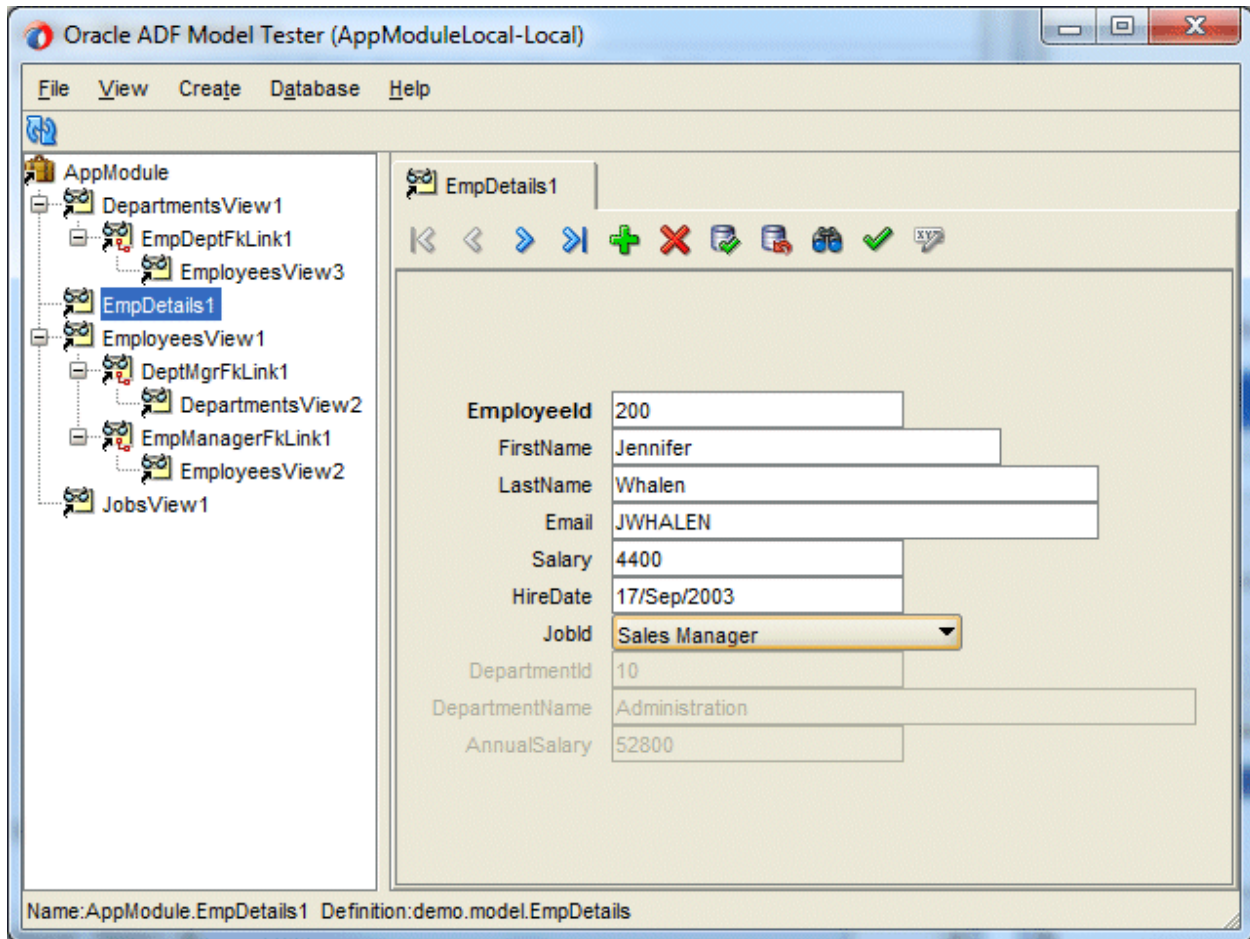


22. Use the drop down list for the **JobId** field to choose a new job.



23. The field is populated with the new job.





24. **Close** the Business Component Browser window without committing the changes.

Courtesy: [https://docs.oracle.com/cd/E37547\\_01/tutorials/tut\\_rich\\_app/tut\\_rich\\_app\\_2.html](https://docs.oracle.com/cd/E37547_01/tutorials/tut_rich_app/tut_rich_app_2.html)

Modified: 2021.10.04.7.45.AM

Dököll Solutions, Inc.