

How-To Develop an ADF Client Using a Web Service Data Control

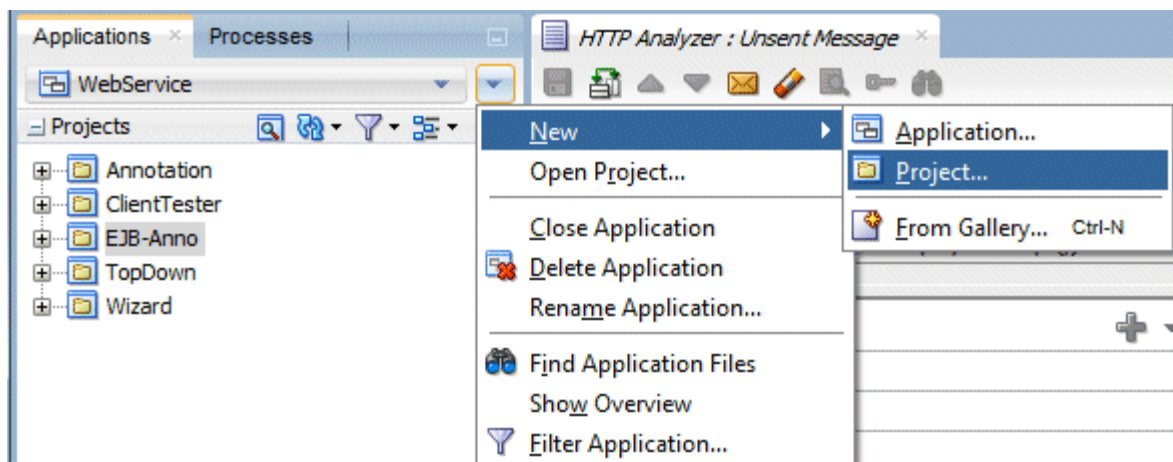
Web services allow enterprises to expose business functionality irrespective of the platform or language of the originating application. The business functionality is exposed in such a way that it is abstracted to a message composed of standard XML constructs that can be recognized and used by other applications. Oracle ADF provides an access layer that can be used to access many types of business or data services, including web services.

The most common way of using web services in an application developed using Oracle ADF, is to create a data control for an external web service. A typical reason for doing this is to add functionality that is readily available as a web service, but which would be time consuming to develop within the application.

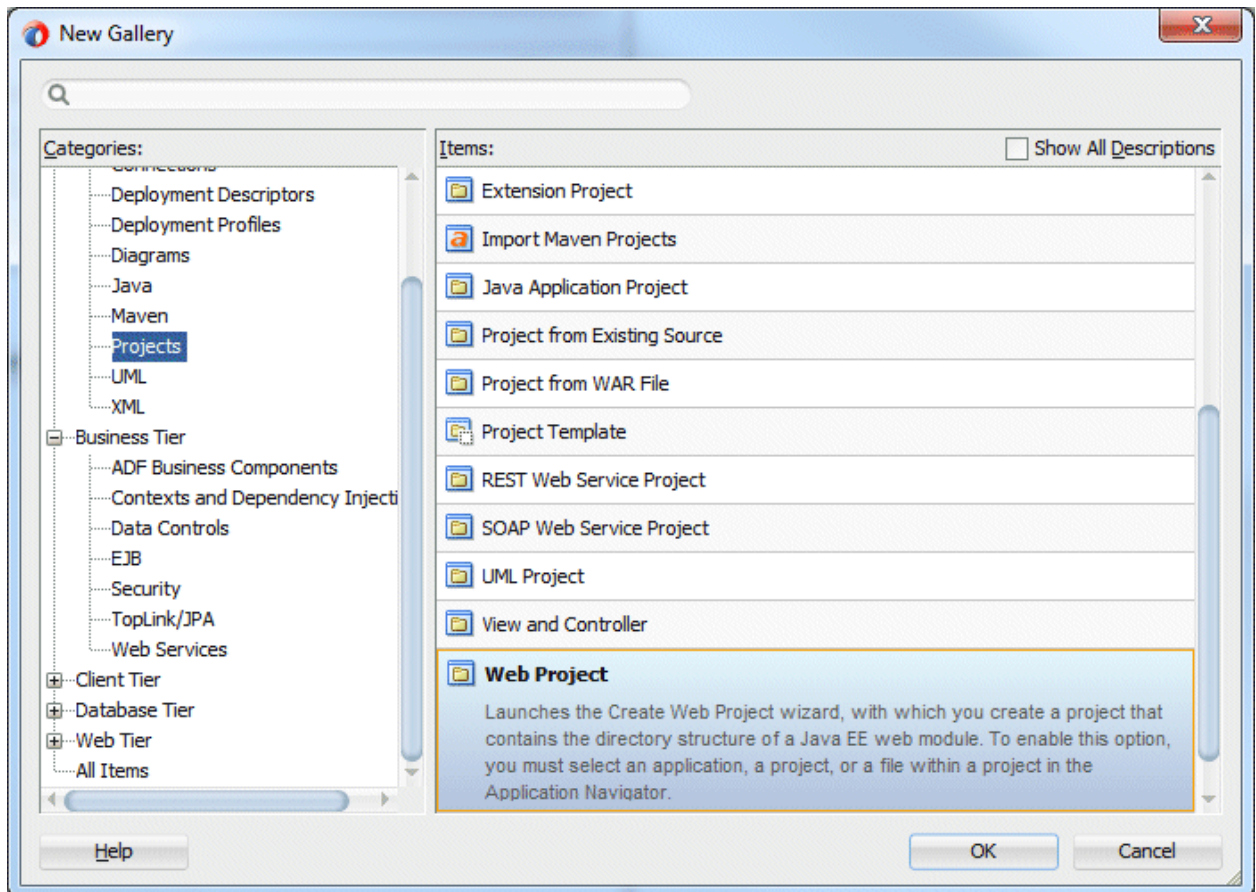
Step 1: Creating the Web Service Data Control

In this section you use the WSDL deployed to the integrated server and create a data control from it. Your application will not contain the original Java code as in the previous scenarios; it creates the data control from the deployed WSDL. Once you create the data control, any ADF client application can use it.

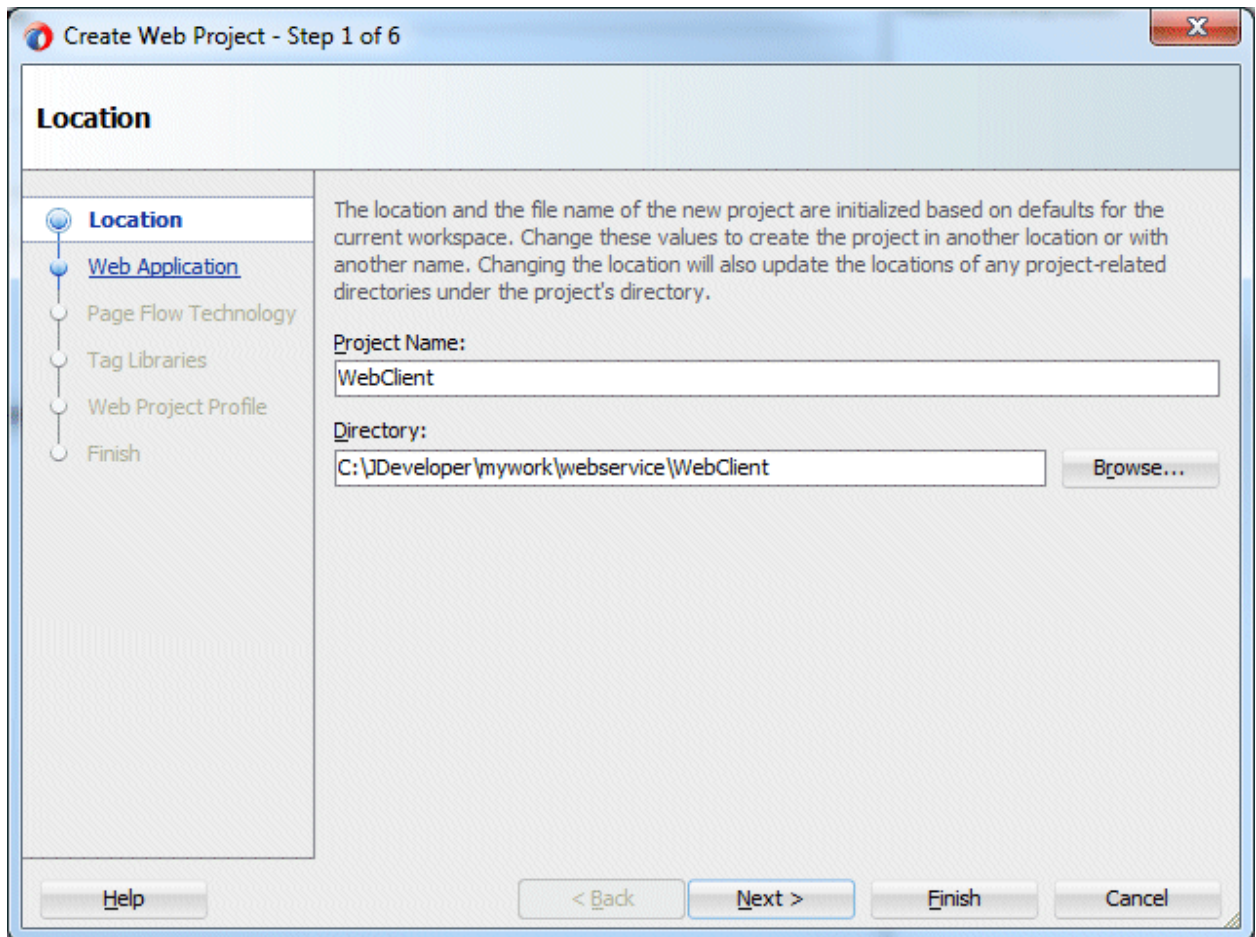
1. In the Applications window, click the **down-arrow** to the far right of the Application name and select **New > Project** from the menu.



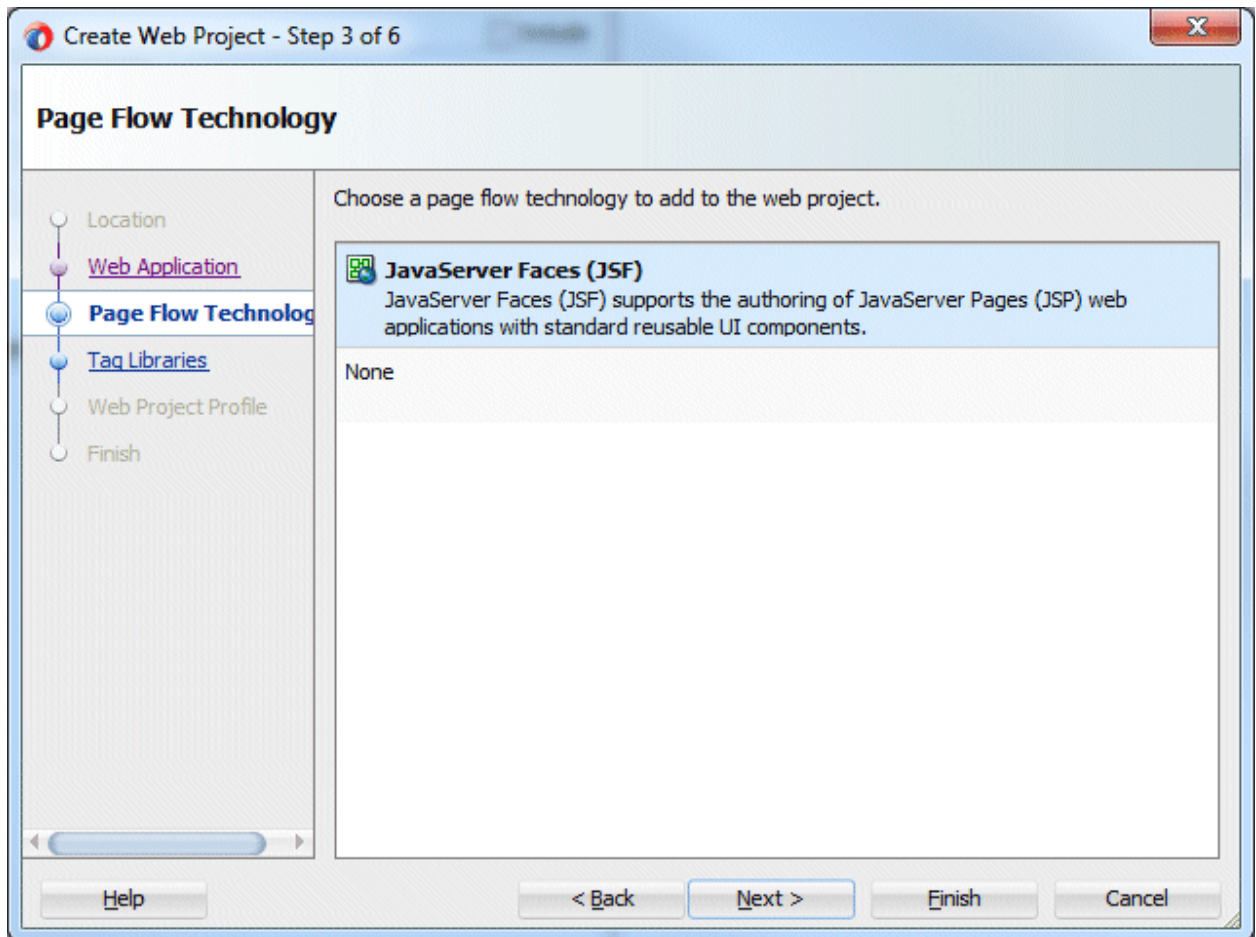
2. In the New Gallery, select **Web Project** and click **OK**.



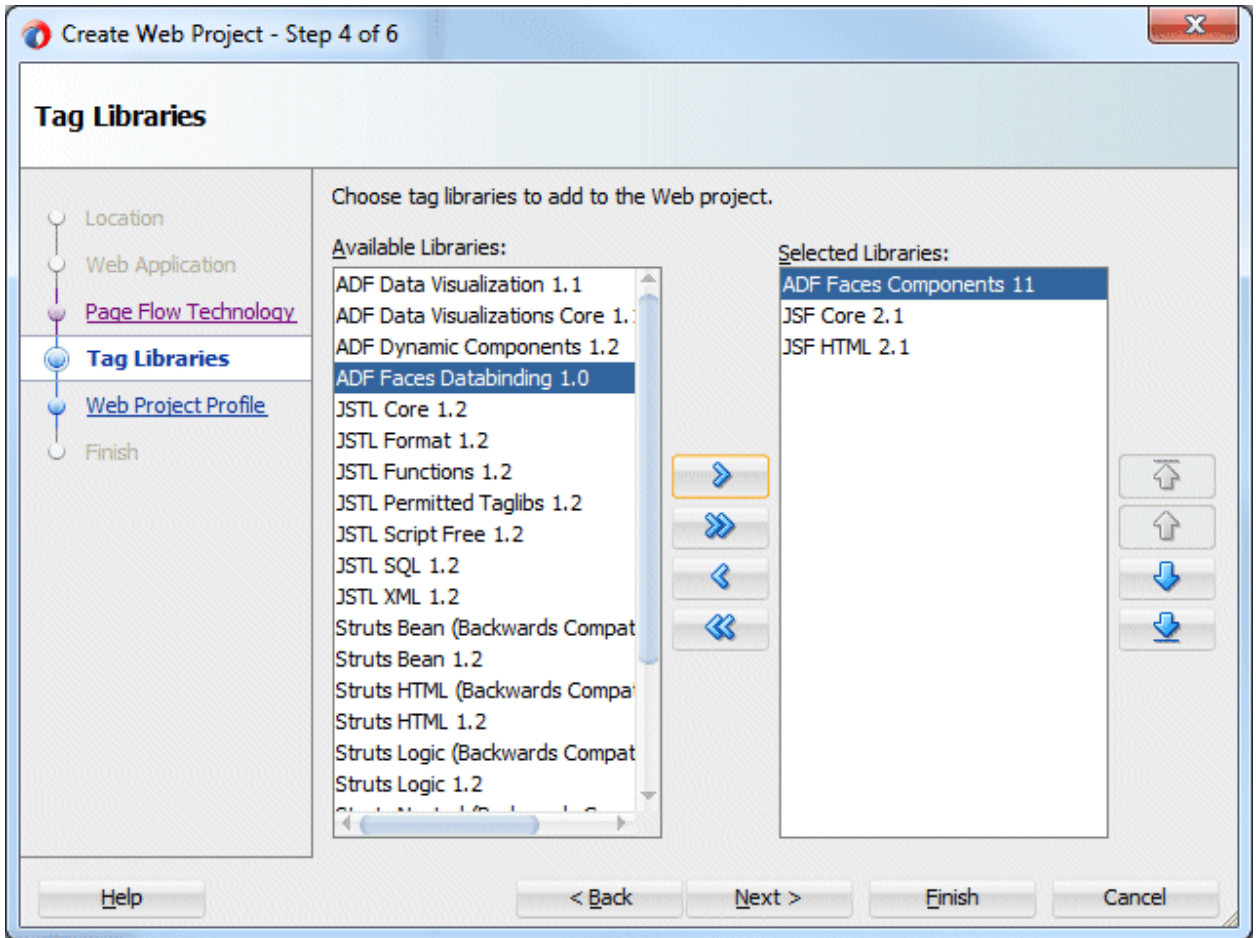
3. In the Create Web Project wizard, name the project WebClient. Click **Next** and **Next** again.



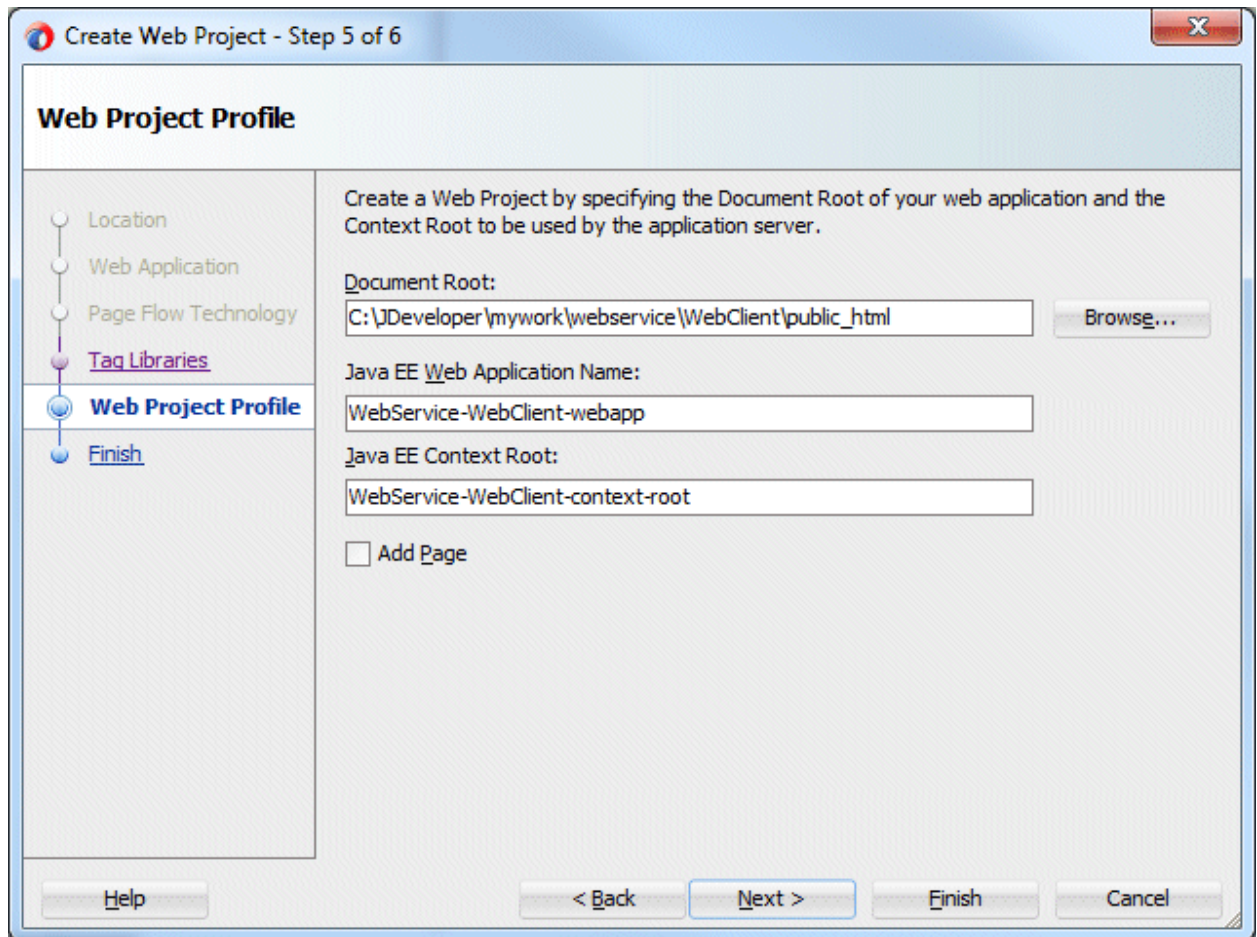
4. In the **Page Flow Technology** step of the wizard, select JavaServer Faces (JSF), and click **Next**.



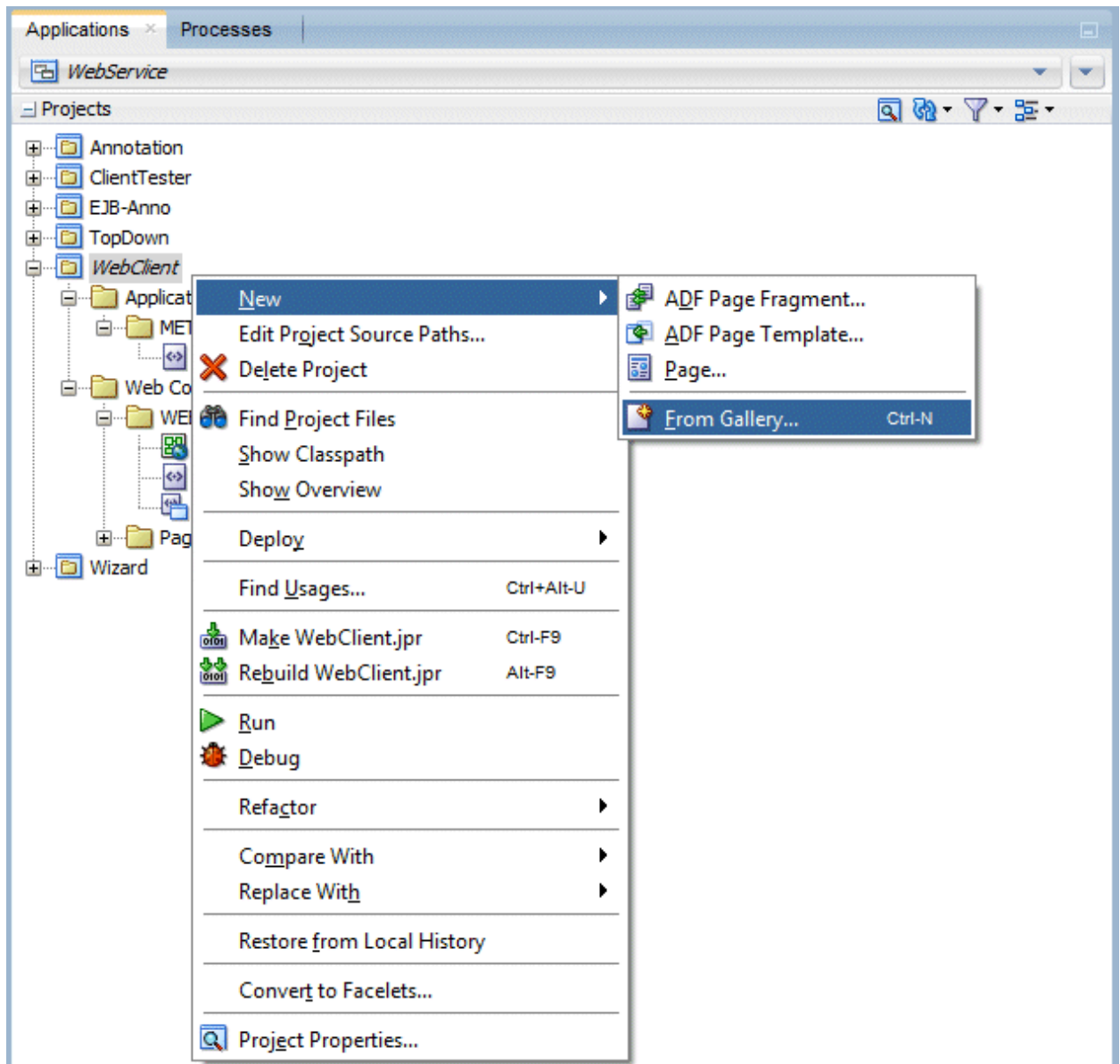
5. In the tag libraries, shuttle the ADF Faces Components 11 from **Available Libraries** to **Selected Libraries**. Notice that it automatically includes JSF Core 2.1 and JSF HTML 2.1.



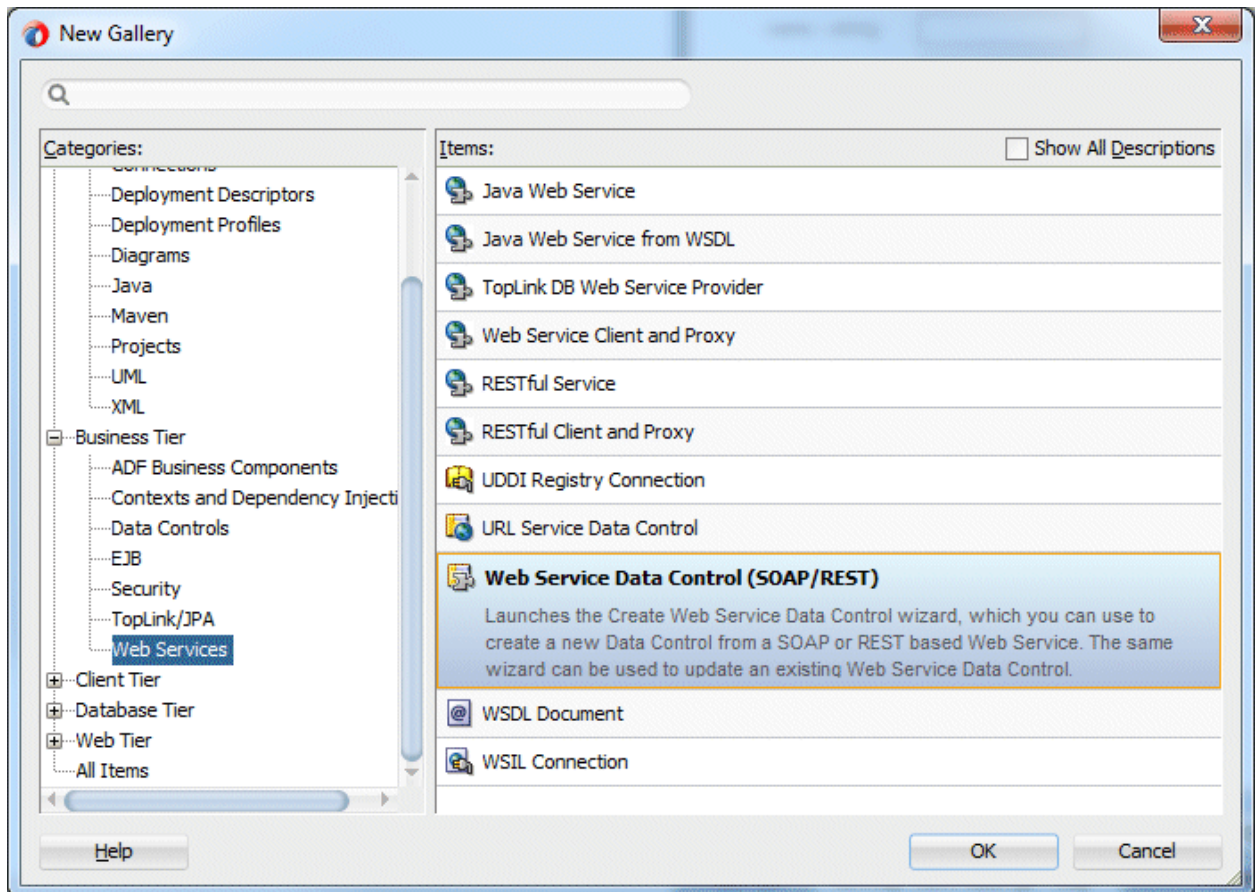
6. Click **Next**.



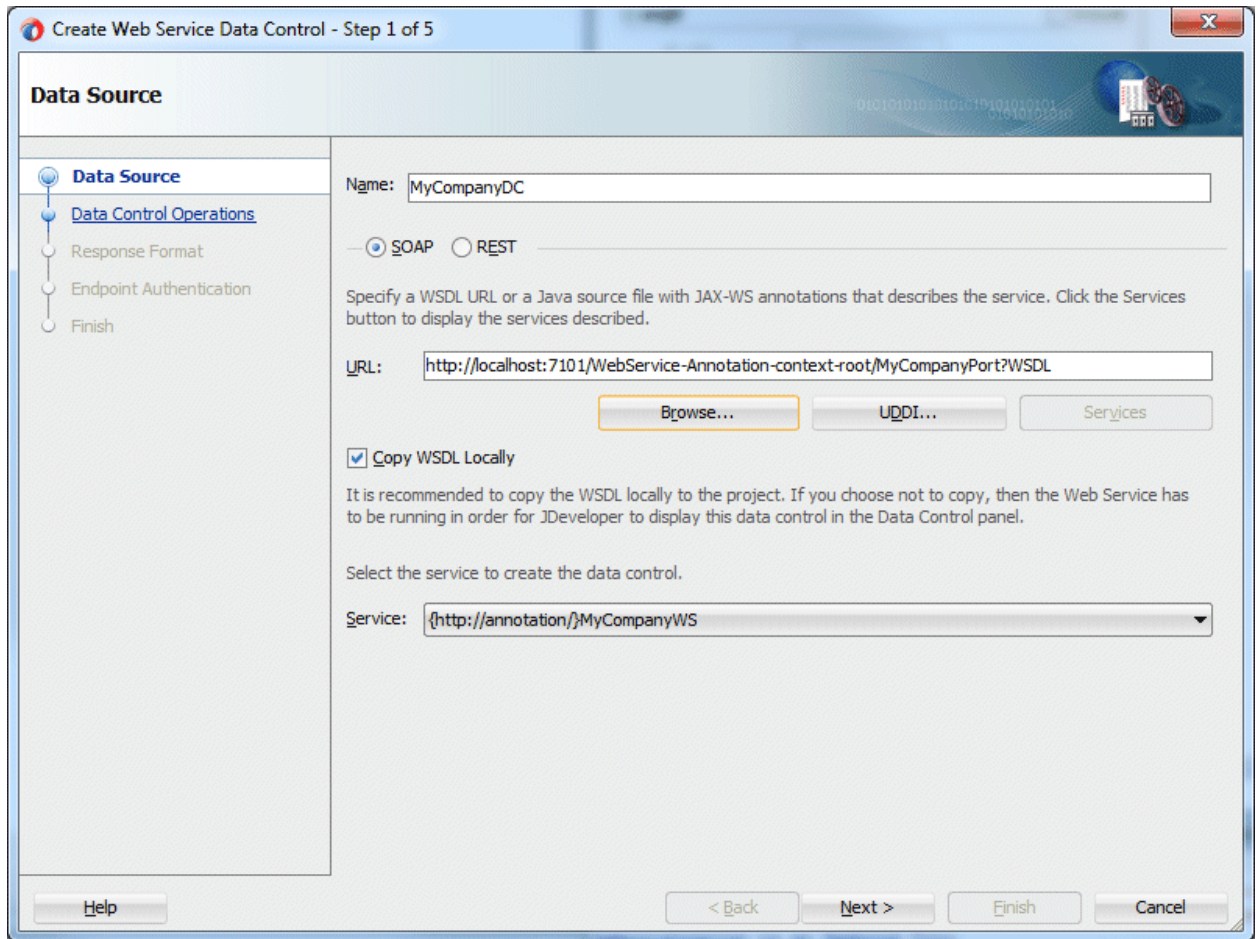
7. Click **Next** and **Finish**.
8. In the **Applications** window, right-click the **WebClient** node and select **New > From Gallery** from the context menu.



9. In the New Gallery **Business Tier** Category, select the **Web Services** node and choose the **Web Service Data Control (SOAP/REST)** in the Items list and click **OK**.

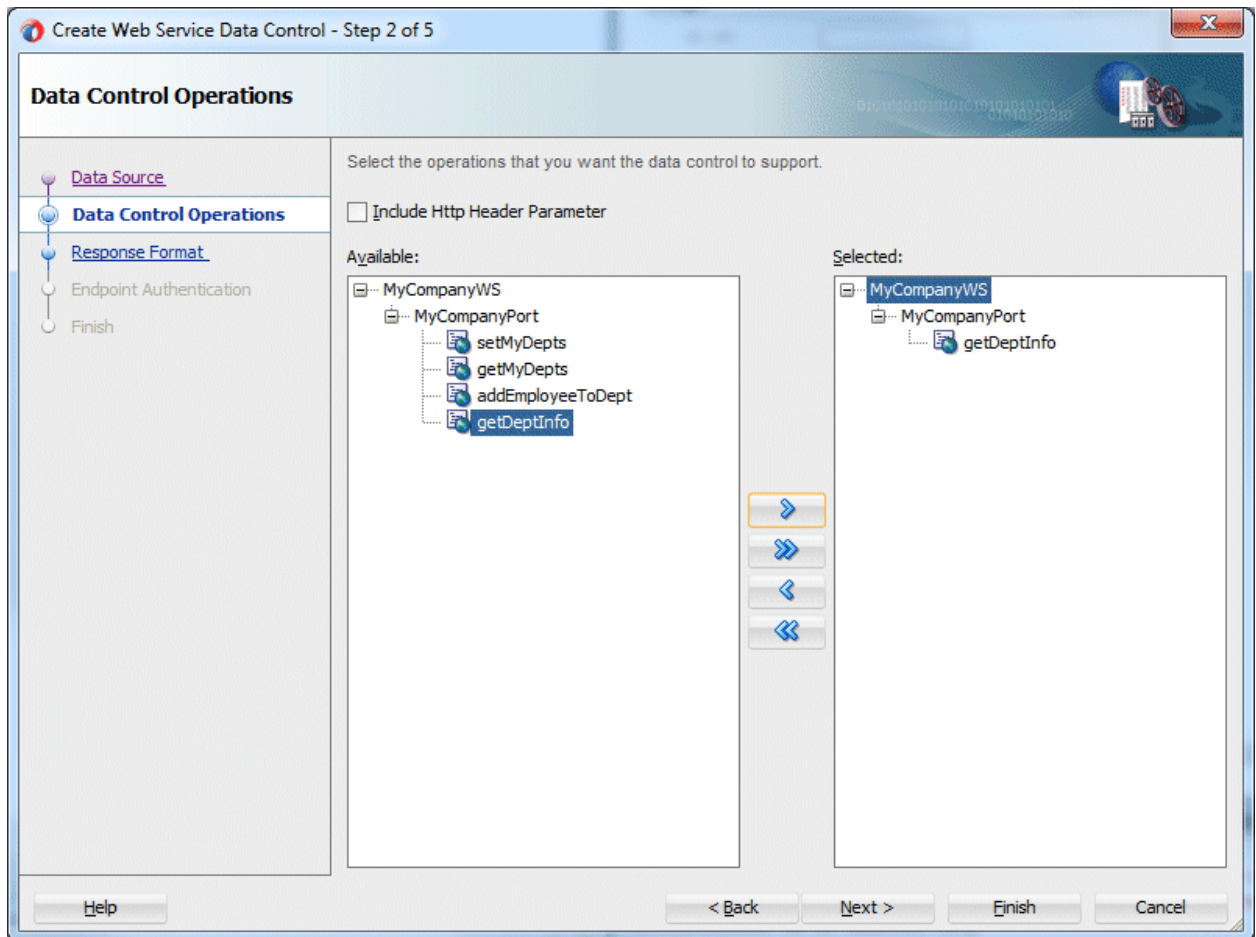


10. In the first step of the Create Web Service Data Control wizard, name the data control **MyCompanyDC**, and then paste the WSDL URL that you copied earlier: <http://localhost:7101/WebService-Annotation-context-root/MyCompanyPort?WSDL>
11. Press **[Tab]**, and the wizard will populate the Service field from the web service.



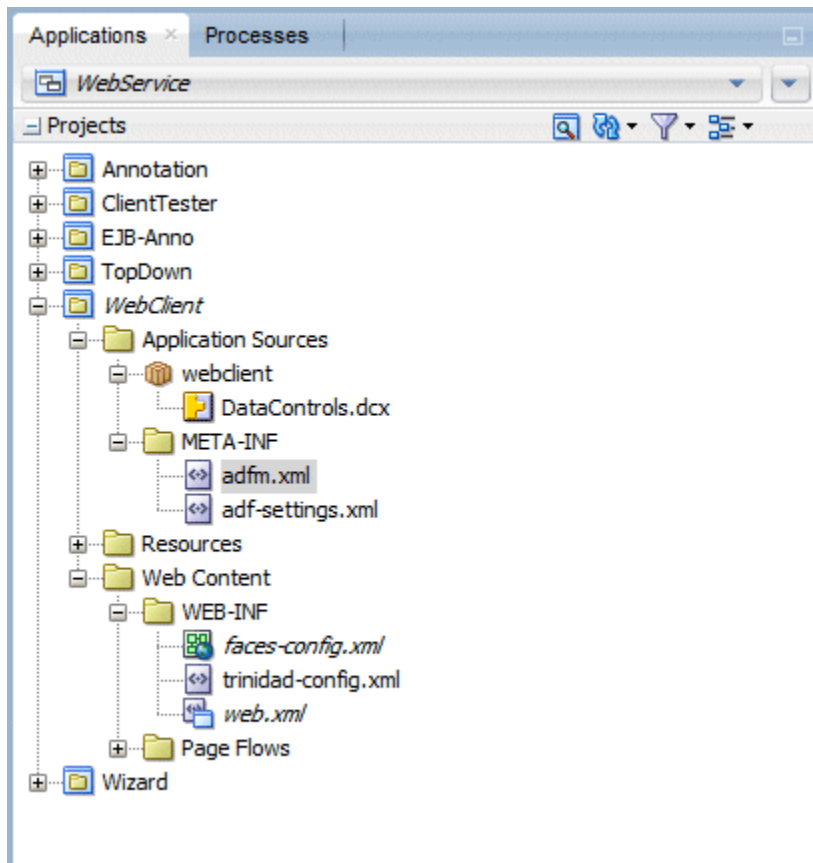
12. Click **Next**.


13. In the **Data Control Operations** page, shuttle the `getDeptInfo` method to the Selected side.

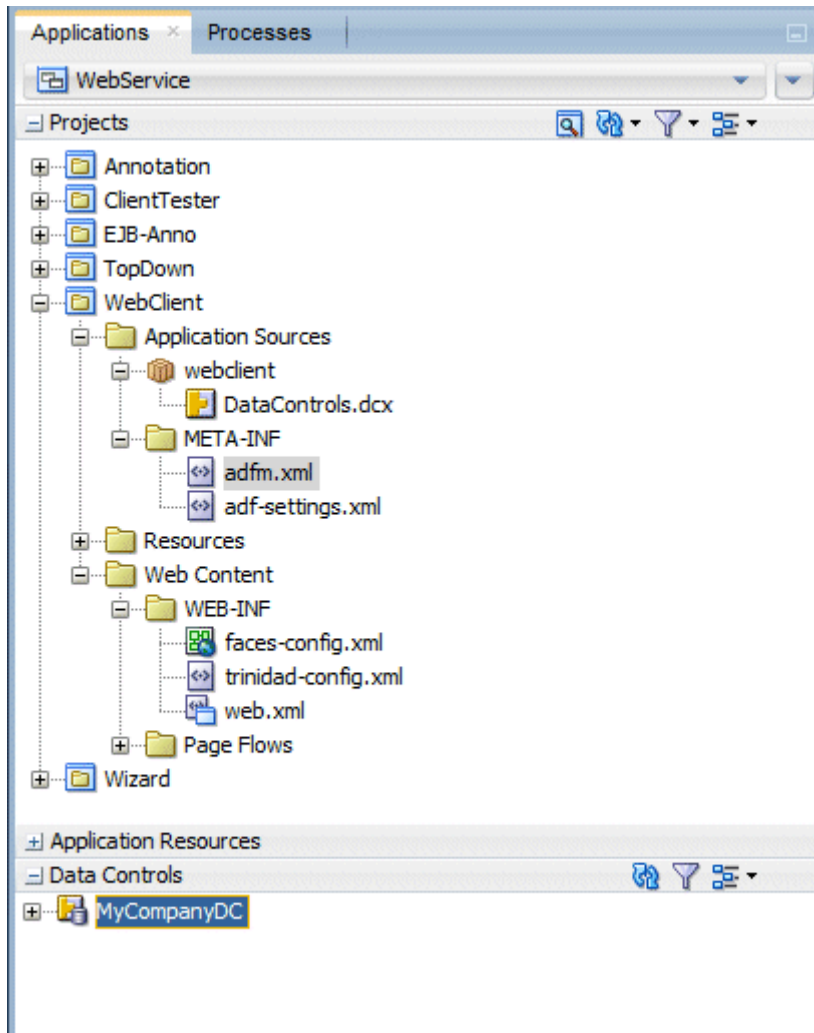


14. Click **Next**, then **Finish**.

The Applications window should look something like:



15. Click **Save All**  to save your work.
16. Expand the **Data Controls** accordion in the Applications window to expose the **MyCompanyDC** data control that has just been created. If it's not visible, then click the blue reload arrow to see the data control.

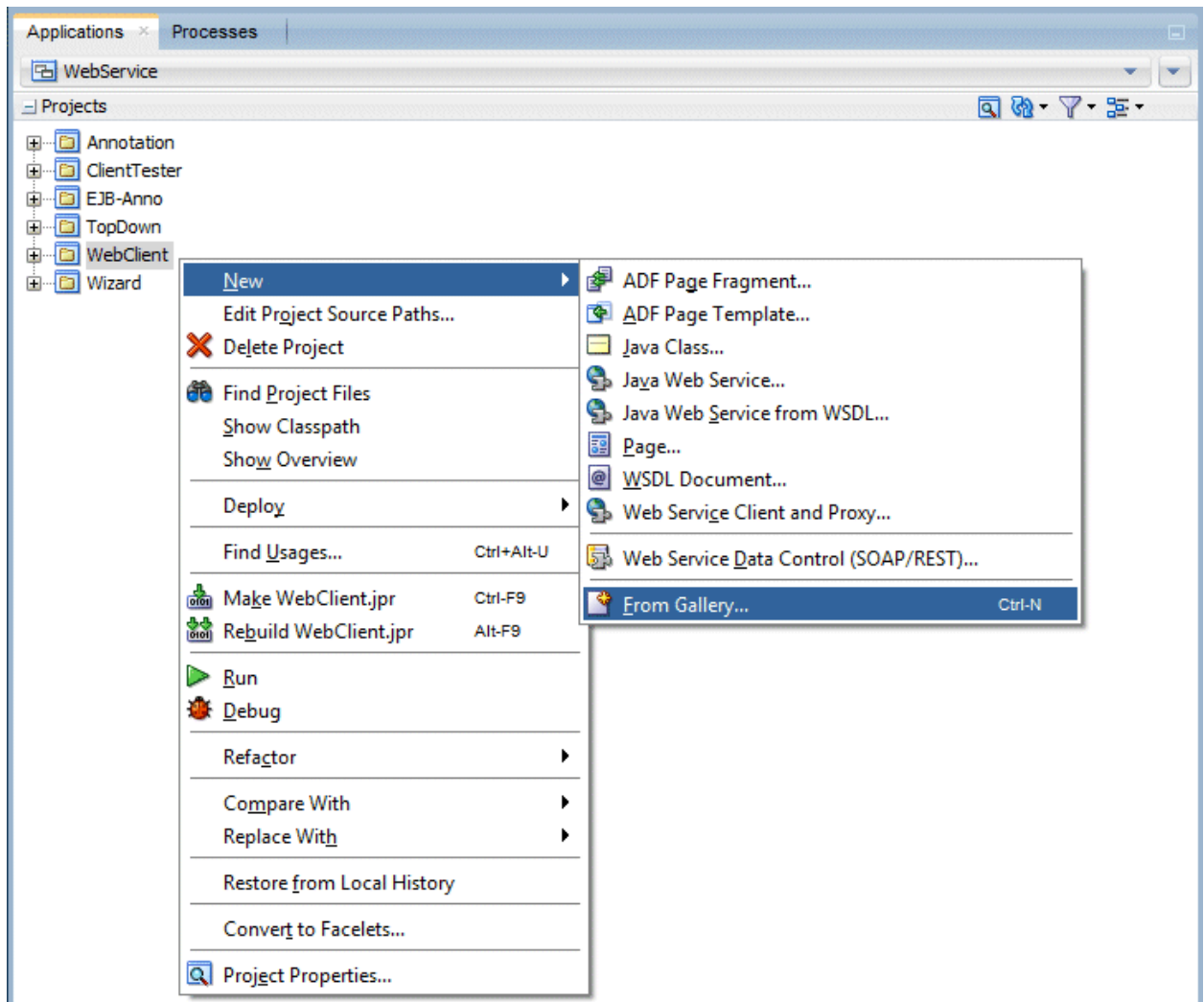


At this point you are ready to create the JSF page and use the data controls.

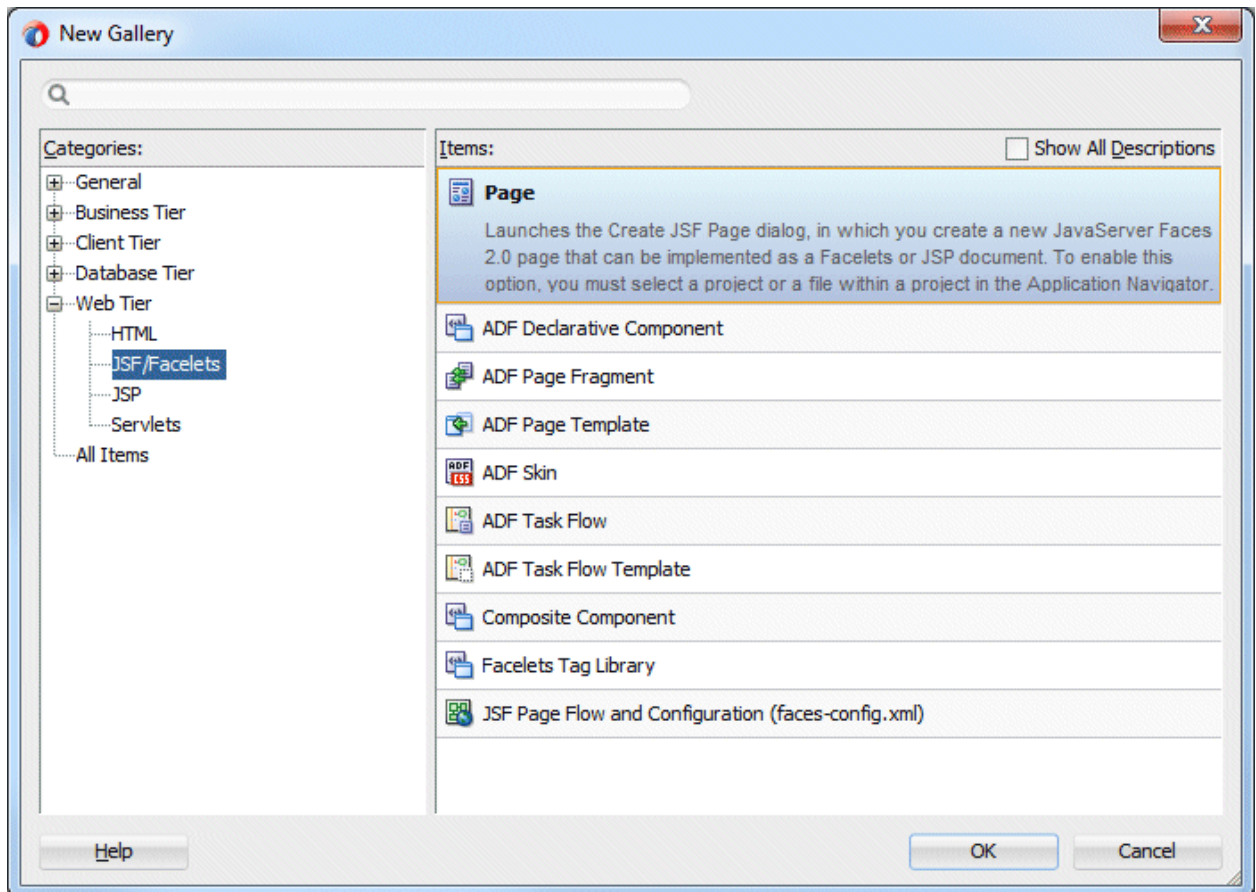
Step 2: Creating a JSF Page to Consume the Web Service Data Control

In this section you update the client class to invoke the web service proxy and return the result to the message window.

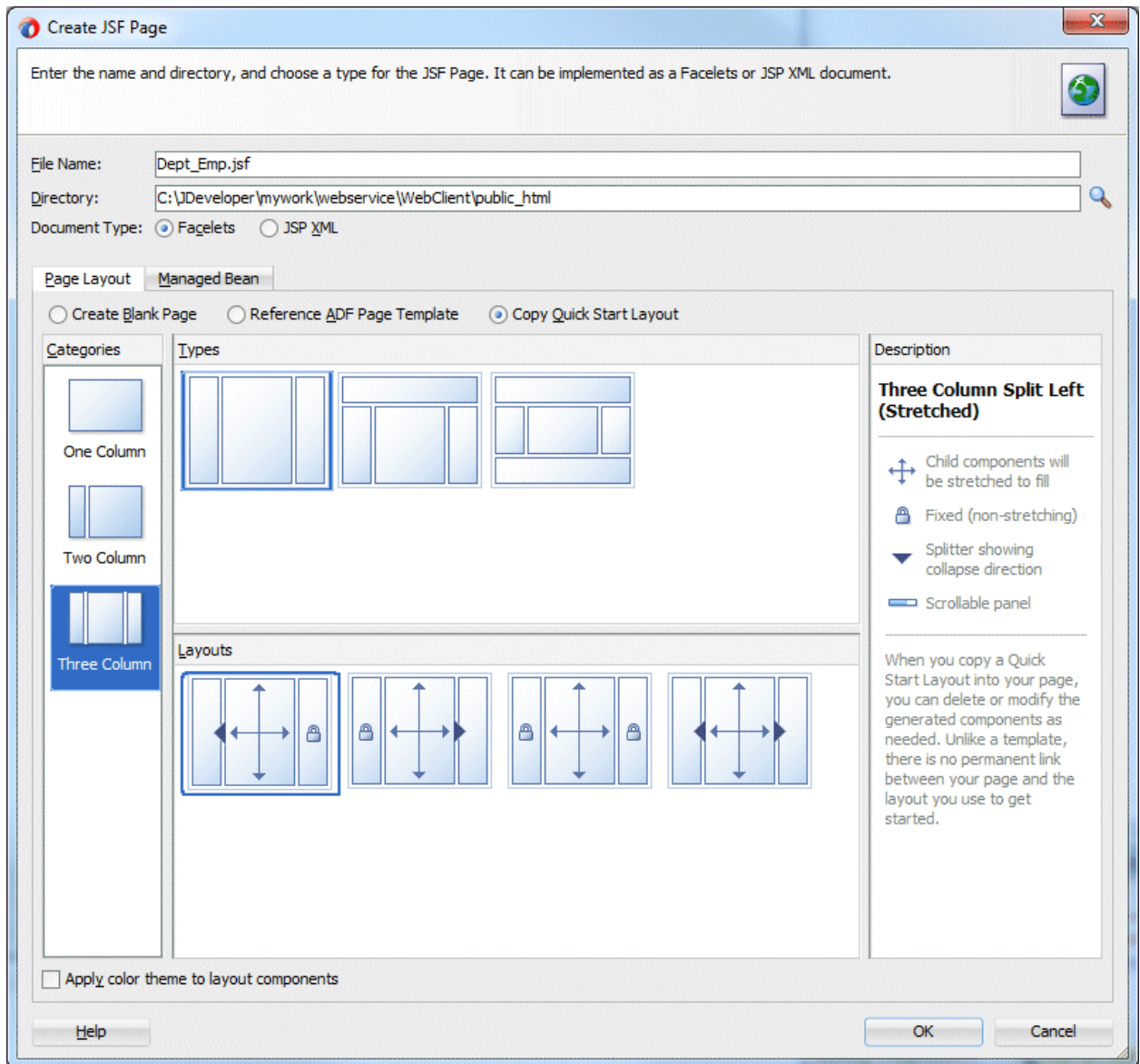
1. Invoke the **New Gallery > From Gallery** from the **WebClient** project.



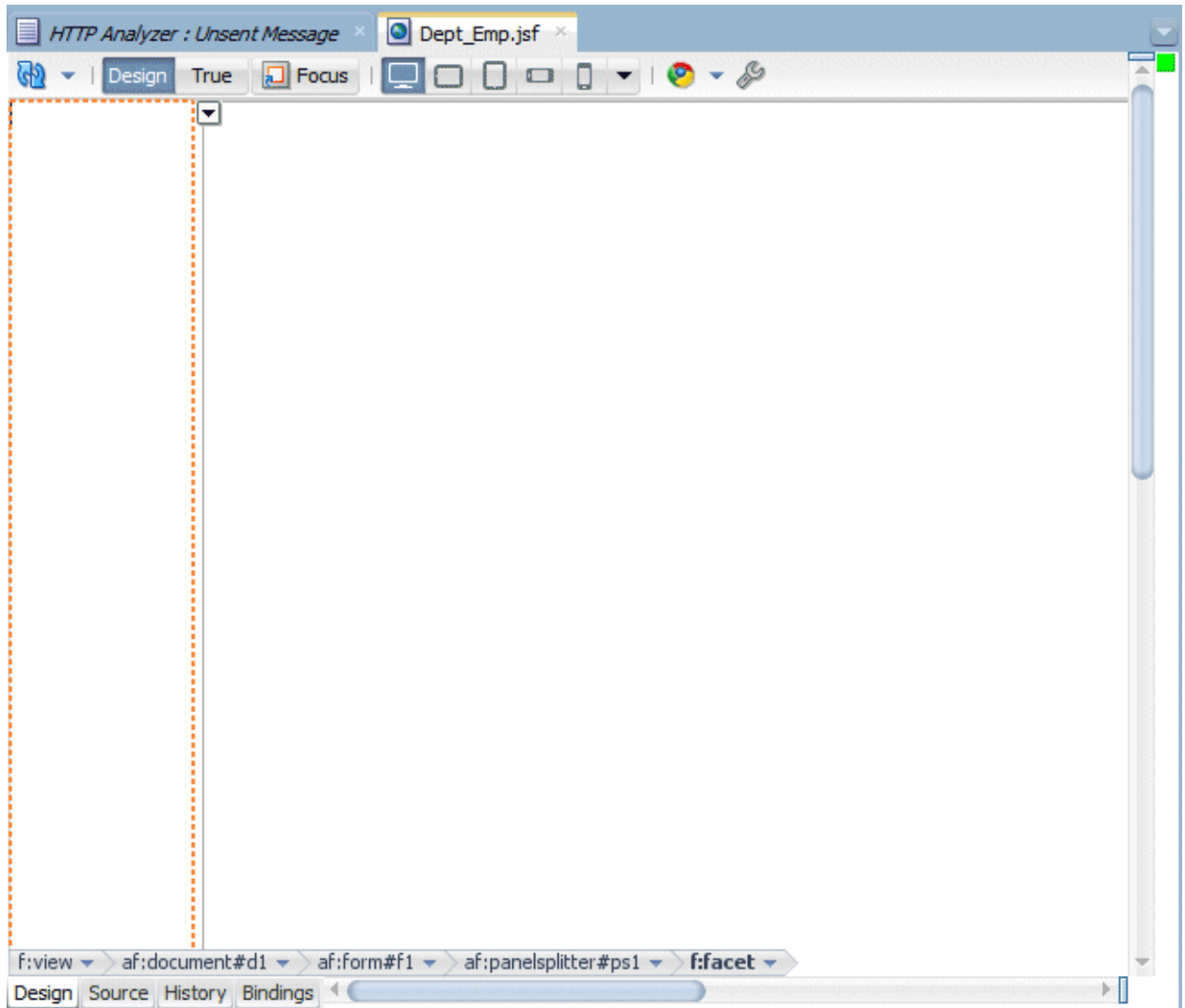
2. Expand the **Web Tier** node and select the **JSF / Facelets** sub node. In the Items column, select **Page** and click **OK**.



3. Name the page Dept_Emp.jsf and set the **Document Type** to Facelets. In the **Page Layout** tab, select Copy Quick Start Layout and in **Categories** select the Three Column Layout and click **OK**.

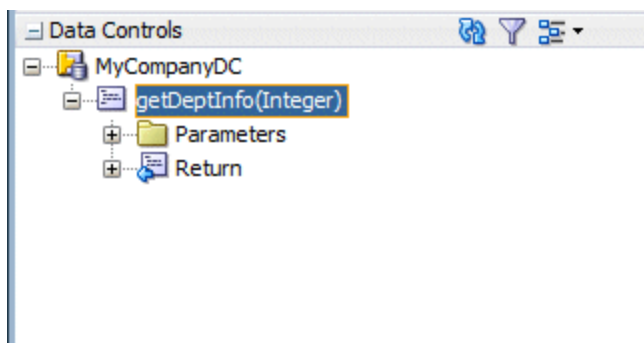


The page will open in the page editor

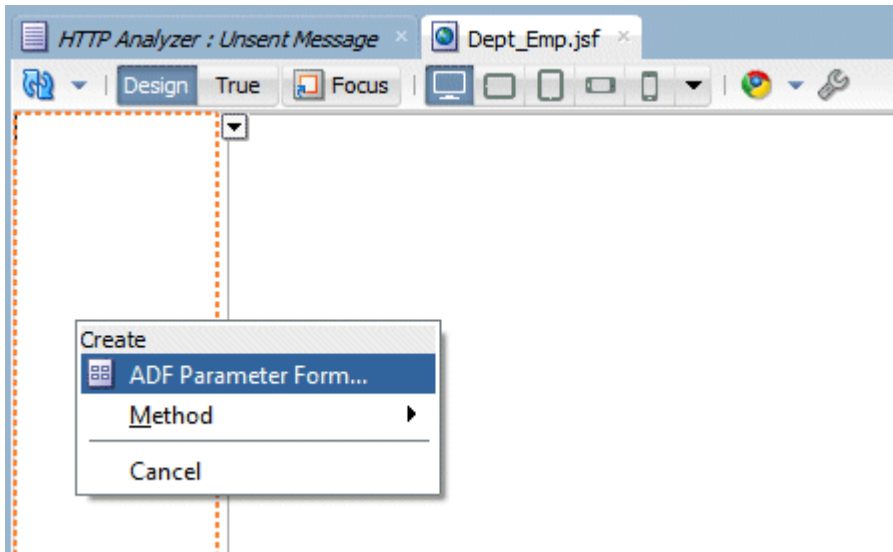


In the next few steps, you add the data controls for the method. The page will accept a parameter, execute the call to the web service, and return the results in a Master Detail layout.

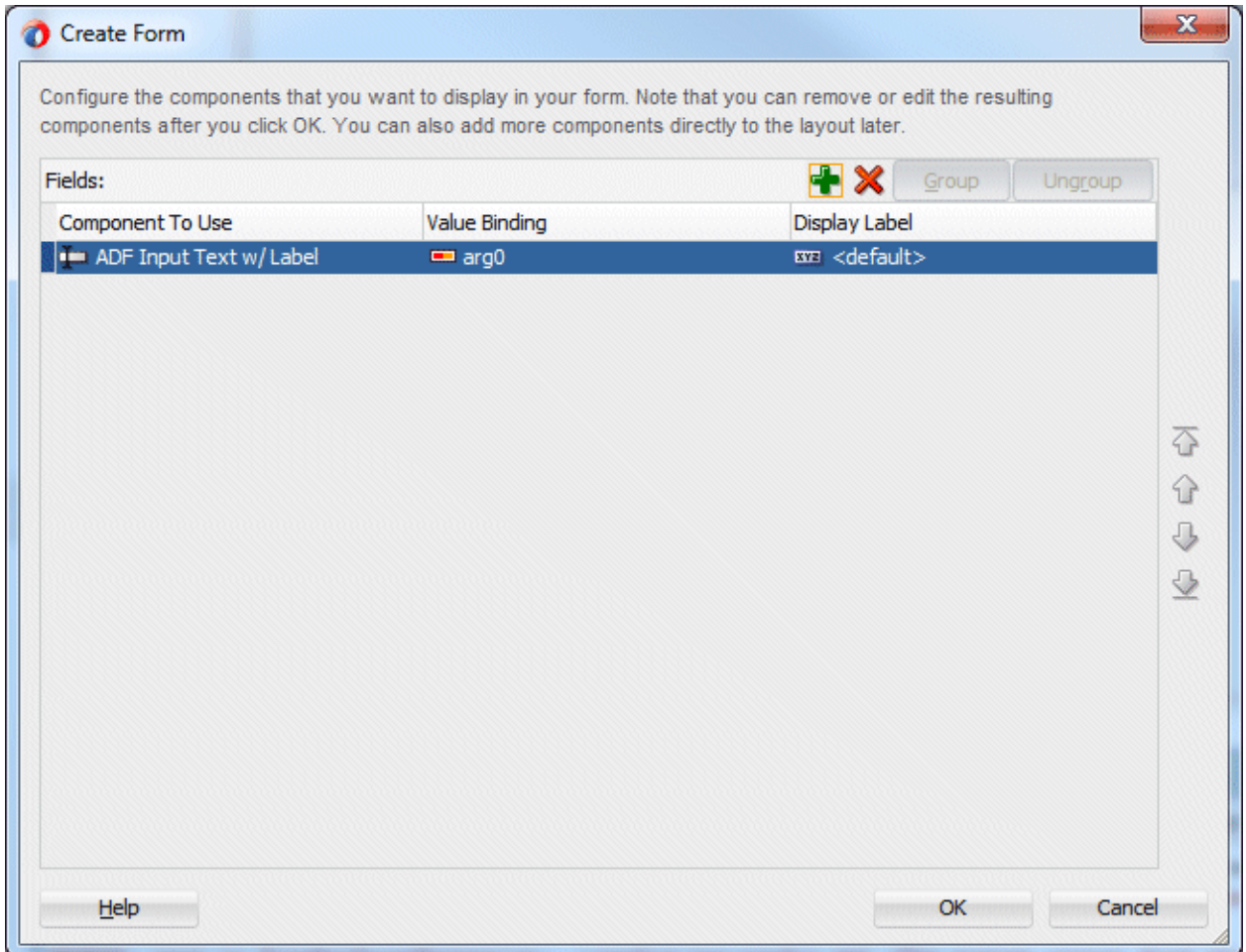
4. In the Data Control panel, expand the **MyCompanyDC** data control and select the **getDeptInfo(Integer)** method.



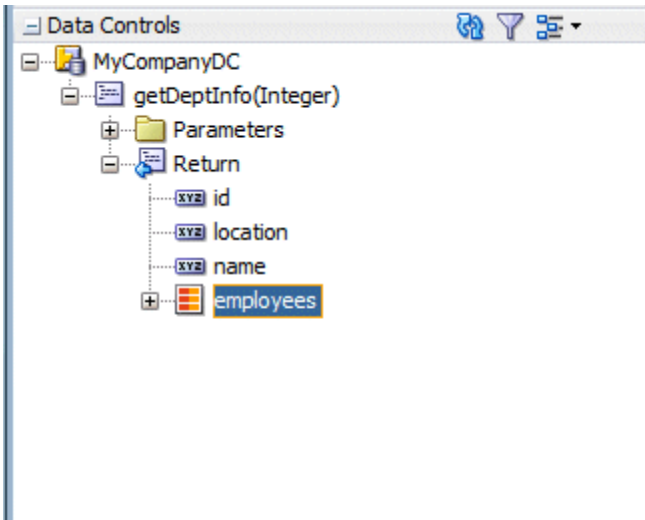
5. Drag the **getDeptInfo(Integer)** method onto the **first** facet of the page. When you drop it, a menu is displayed. Select **ADF Parameter Form**.



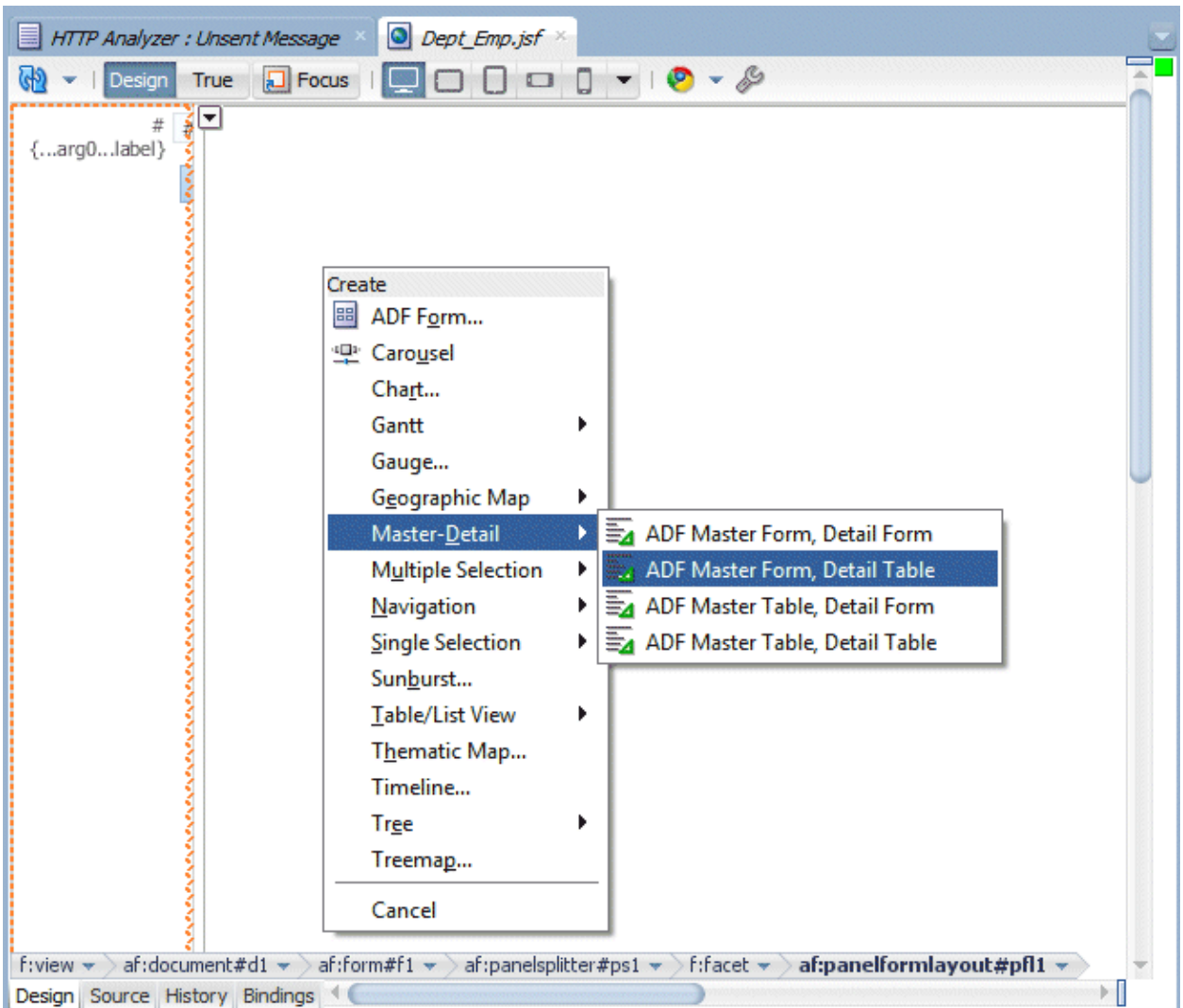
6. In the **Create Form** dialog, click **OK** to accept the default values.




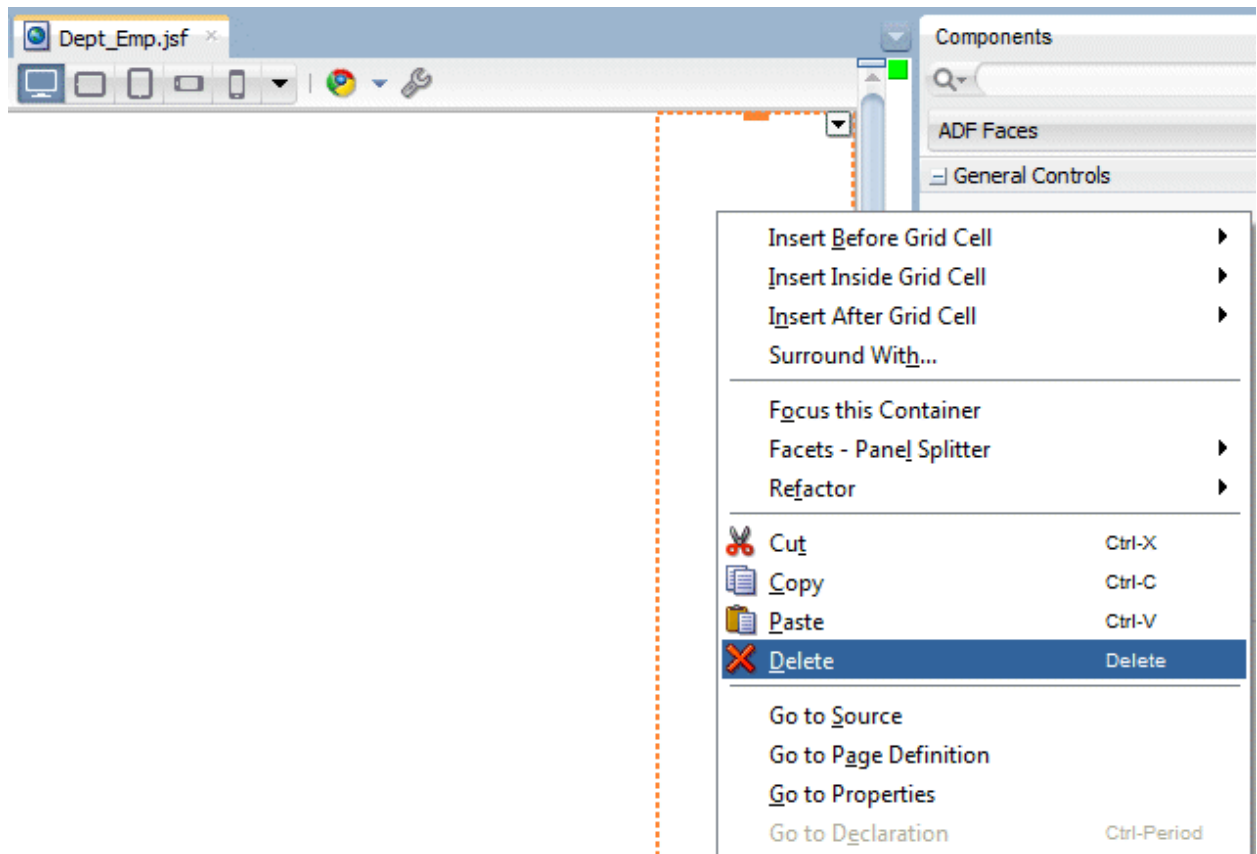
7. In the Data Controls panel, expand **MyCompanyDC > getDeptInfo(Integer) > Return** and select **employees**.



8. Drag **employees** onto the **second** facet (just to the right of the first facet). In the dynamic menu select **Master-Detail > ADF Master Form, Detail Table**.



9. Click **Save All**  to save your work.
10. Right-click within the **end** facet and select **Delete** from the context menu.




11. In the Structure window for the page, expand the **second** facet. Under the second facet, expand the **af:panelGridLayout > af:gridRow > af:gridCell > af:panelGroupLayout > af:panelHeader - employees**. Right-click the **af:table - t1** component and select **Surround With** from the context menu.





Dept_Emp.jsf - Structure

- Issues (2)
- html
- f:view
 - af:document - Dept_Emp.jsf
 - af:messages
 - af:form
 - af:panelSplitter - horizontal
 - Panel Splitter facets
 - f:facet - first
 - Left
 - af:panelFormLayout
 - af:inputText - #{...arg0...}
 - af:button - getDeptInfo
 - Panel Form Layout facets
 - f:facet - second
 - af:panelGridLayout - pgl1
 - af:gridRow - 100%
 - af:gridCell - 100%
 - Content
 - af:panelGroupLayout
 - af:panelHeader
 - af:panelHeader
 - af:table - t
 - Panel Head
 - Panel Group La
 - Document facets

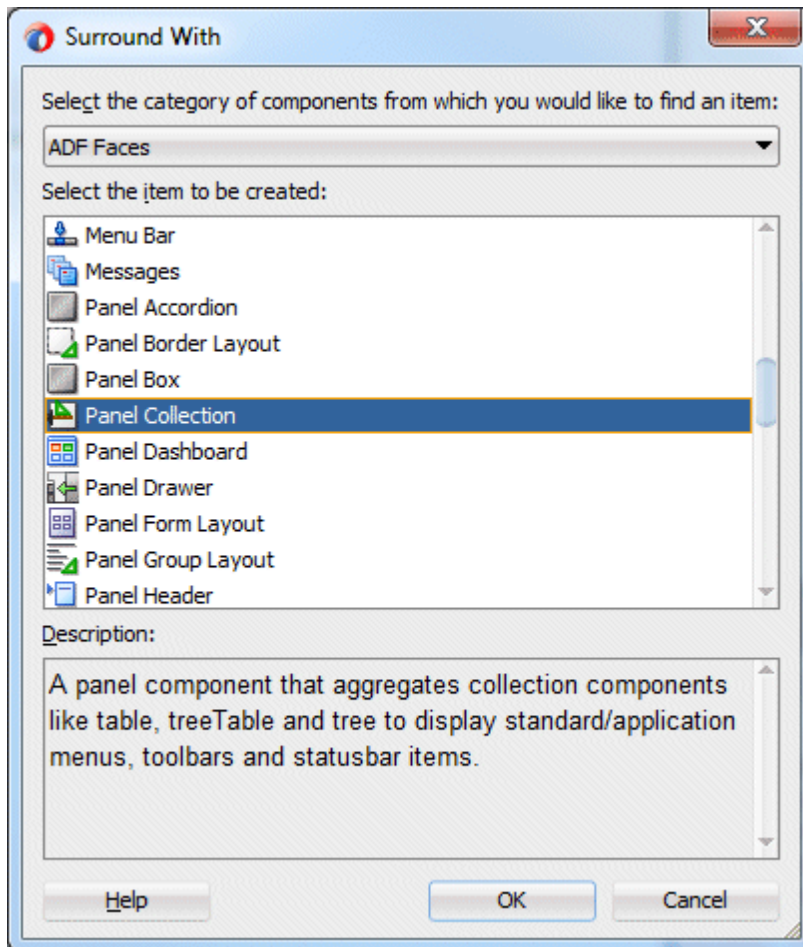
- Insert Before Table ▶
- Insert Inside Table ▶
- Insert After Table ▶
- Surround With...

- Focus this Container
- Table ▶
- Facets - Table ▶
- Hide Component in Design View
- Refactor ▶
-  Rebind to Another ADF Control Ctrl-B

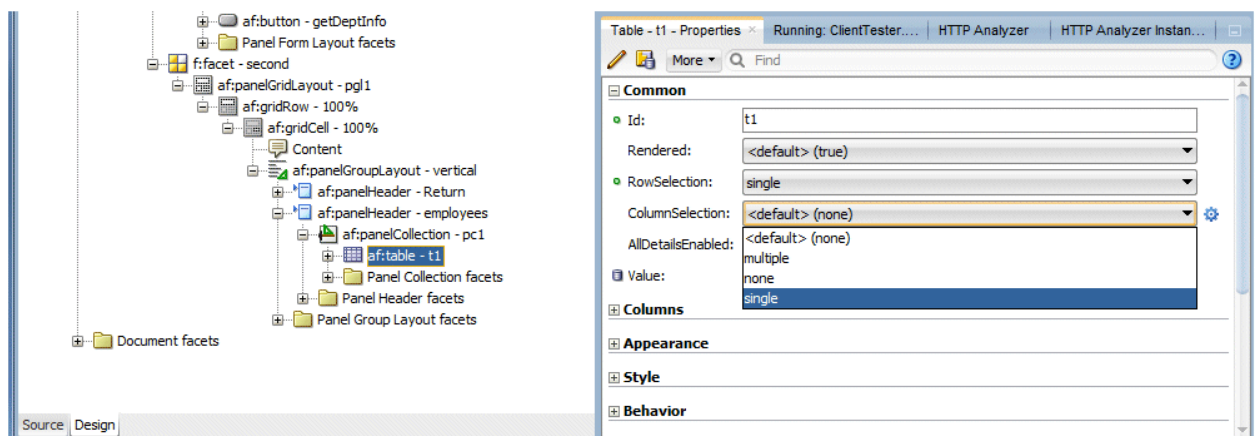
- Expand All Below
- Collapse All Below
- Show As Top
- Show Root

-  Cut Ctrl-X
-  Copy Ctrl-C
-  Paste Ctrl-V
-  Delete Delete

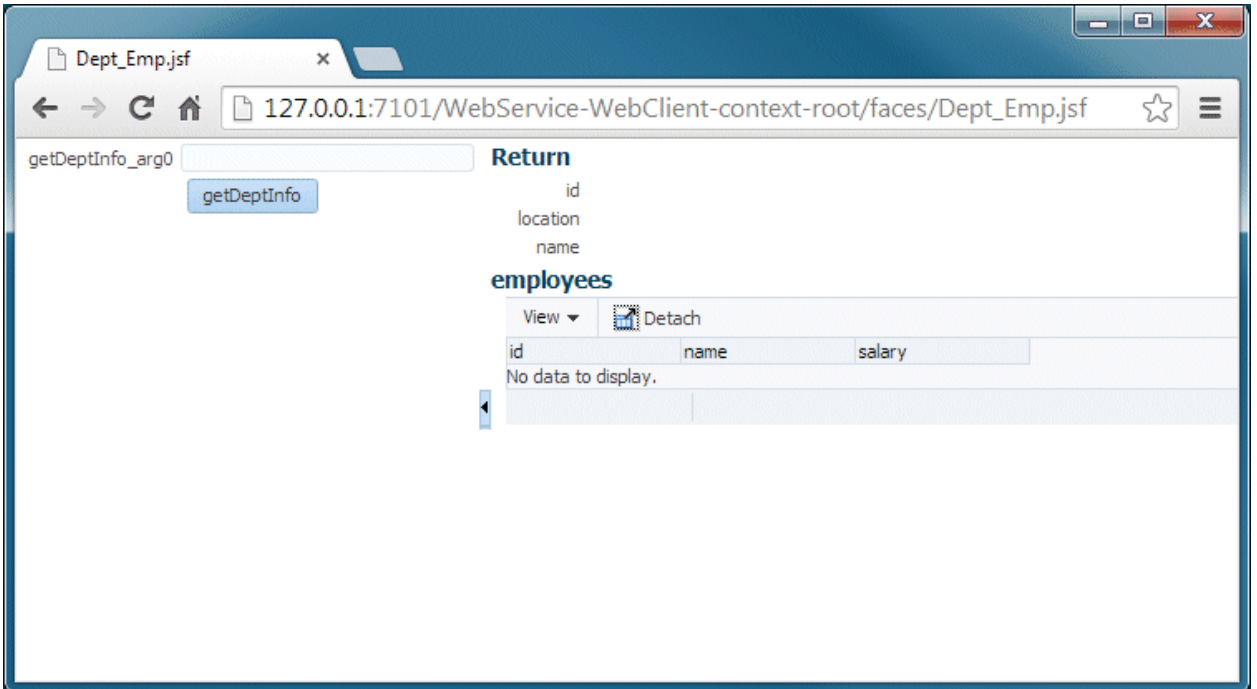
12. In the **Surround With** window select the Panel Collection and click **OK**.



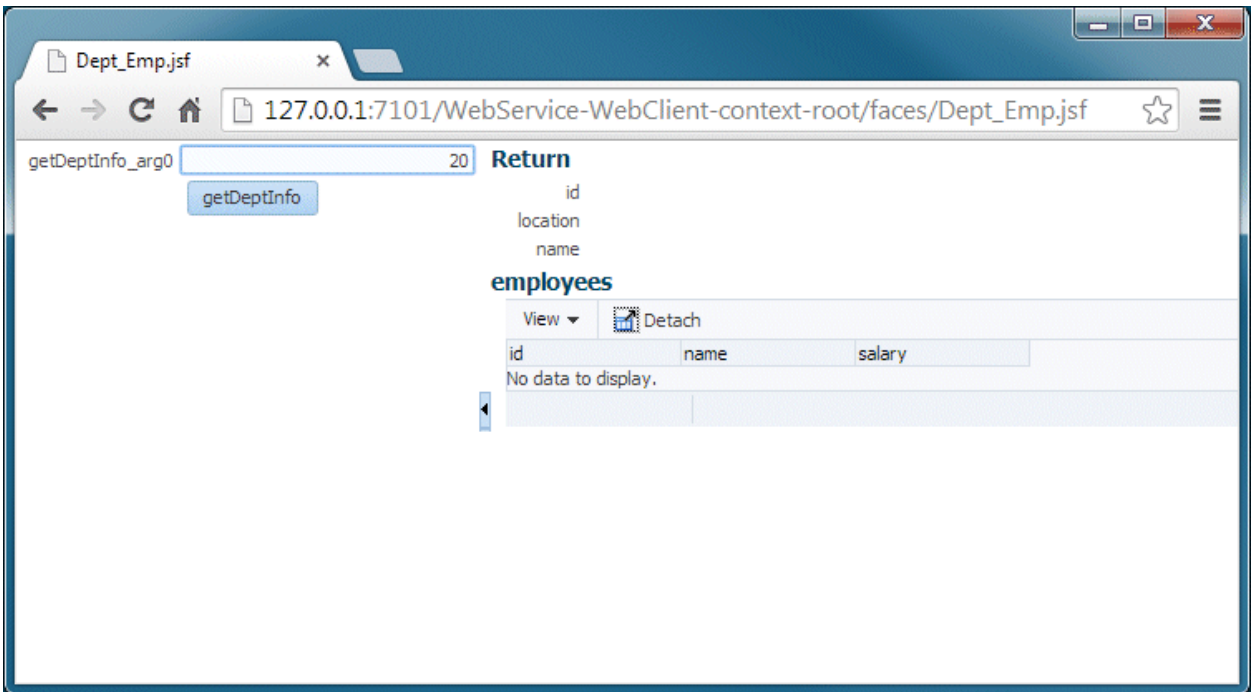
13. Expand the **panelCollection** and select the **af:table - t1** component. In the Properties window, set the **Column Selection** property to single.



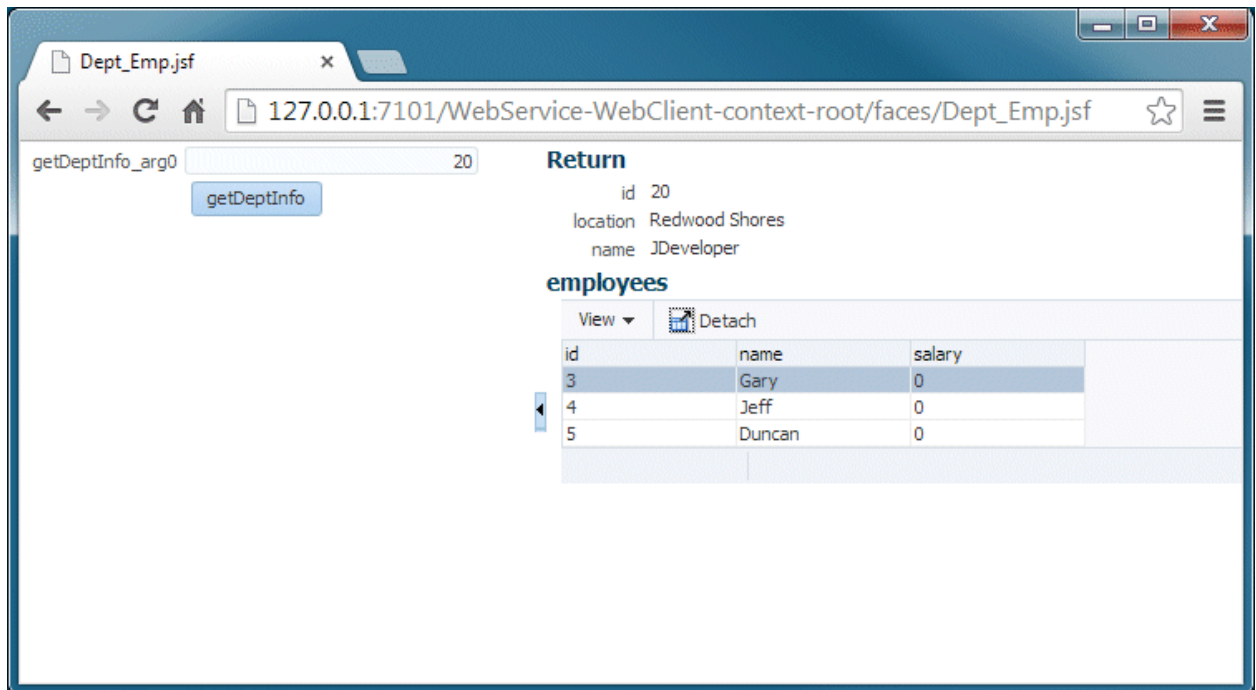
14. Save all your work.
15. Right-click within the page and select **Run** from the context menu.
16. The page loads into your default browser.



- C
17. In the parameter field enter 30 as the department value and click the **getDeptInfo** button.



18. The department information and the related employees are retrieved and displayed



Summary:

In this tutorial you have seen how to create Web Services from two perspectives; Top-down and Bottom-up. In the Top-down approach you created Web Services from scratch based on what you wanted to do. In the Bottom-up sections, you created Web Services based on existing services. You learned how to:

- Create an annotated POJO Web Service
- Create a POJO and use JDeveloper to declaratively annotate the Web Service
- Build a Web Service using an existing WSDL document
- Build a annotated Web Service based an Enterprise Java Bean
- Create a proxy class for an existing Web Service so you can consume it for a Java program
- Create an ADF Data Control based on an existing Web Service

Courtesy: https://docs.oracle.com/cd/E53569_01/tutorials/tut_web_services/tut_web_services_6.html

Modified: 2021.10.14.7.34.PM

Dököll Solutions, Inc.