



Tutorial: Building a Web user interface for a subfile

Version 5.1



Tutorial: Building a Web user interface for a subfile

Version 5.1

Contents

Introduction to this tutorial	v	Exercise 3.2: Editing an iSeries source member.	38
Learning objectives	v	Exercise 3.3: Compiling source changes	41
Prerequisite knowledge	vi	Recap	45
Chapter 1. Module 1: Reviewing the existing RPG e-business application . . .	1	Chapter 4. Module 4: Creating the update record output page	47
Exercise 1.1: Reviewing the Web customer inquiry application	1	Exercise 4.1: Generating the output page from the Web Interaction wizard	47
Exercise 1.2: Introducing the enhanced Web application	4	Exercise 4.2: Linking to the update record page	59
		Exercise 4.3: Creating the update error page	63
		Recap	66
Chapter 2. Module 2: Creating the customer list input page	9	Chapter 5. Module 5: Creating the update complete output page	67
Exercise 2.1: Starting the Development Studio Client workbench	10	Exercise 5.1: Generating another output page using the Web Interaction wizard	67
Exercise 2.2: Opening the Web perspective	11	Exercise 5.2: Adding the flow control error page	77
Exercise 2.3: Creating the customer list page	12	Exercise 5.3: Linking the update complete page to the inquiry page.	79
Exercise 2.4: Linking to the customer list page.	21	Recap	81
Exercise 2.5: Creating another Web interaction.	26		
Exercise 2.6: Adding an iSeries ILE program to the interaction.	30		
Creating a connection	30	Chapter 6. Module 6: Running the Web application	83
Selecting the service program	31	Exercise 6.1: Restarting the server	83
Exercise 2.7: Adding a parameter	33	Exercise 6.2: Testing the Web application	84
Recap	35	Recap	88
Chapter 3. Module 3: Adding the iSeries program to your Web application	37	Appendix. Notices	89
Exercise 3.1: Opening the Remote System Explorer perspective	37	Programming interface information	90
		Trademarks	91

Introduction to this tutorial

In the tutorial called Building an RPG e-business application, you learned how to create an e-business RPG customer inquiry application. When you ran your Web RPG application a 'new' Web customer inquiry page opened in the workbench Web browser. You then entered a customer number and clicked the submit button. The customer details from a DB2/400 database appeared in the workbench Web browser.

In this tutorial, you will learn how to display a list of customers, display the details for each selected customer and then update selected customer information. You will learn how to use Page Designer to create a new customer list input page from a subfile. You will then learn how to link this new page to the customer inquiry page that you built in the previous tutorial. Then you will learn how to create new output pages (update record and update complete) using the Web Interaction wizard. You will also learn how to link these pages using the Web interaction wizard. Finally you will learn how to test your new and updated input and output pages in your existing e-business RPG customer inquiry application.

Note: Because this is an advanced tutorial, the instructions do not take you through each step to build a Web interface for a subfile, but outline the development steps taken to create such an interface.

Tutorial purpose:

Before you begin, take a minute to read the learning objectives for the tutorial.

Length of time:

This tutorial takes approximately 1.0 hour to complete.

Modules:

Work on the modules in sequence. The pictures in the modules show similar tasks. Some of the names and icons may be different from the environment you will be working with when you complete the modules.

Prerequisites

You should also learn about the prerequisite knowledge for the tutorial before beginning. This section describes the type of knowledge you need to already have to fully benefit from this tutorial.

Learning objectives

In this tutorial, you will create a new input page to view a list of customers (custList.jsp) and use an existing output page (result.jsp) to create new output pages to update customer records and confirm updates were completed on the iSeries™ (custDispUpdateResults.jsp and custDoUpdateResults.jsp) for your existing e-business customer inquiry application. While creating these pages, you will learn how to use the iSeries wizards in Development Studio Client to link pages together to form an application with multiple Web interactions. You will then learn how to run the updated customer inquiry application in the WebSphere® Test Environment that is part of Development Studio Client.

The tutorial is broken into five modules, each with their own specific learning objectives. You can choose to complete all modules, or you can choose to complete

only a few, depending on your own learning goals. Each module is composed of several exercises. These exercises must be completed in order to successfully achieve the module's goals.

The first module, Reviewing the Web customer inquiry application, shows you the completed customer inquiry application that you will build in the tutorial called Building an e-business RPG application. You will then see how this application changes with the updates you will make working through this tutorial.

The second module, Creating the customer list input page, helps you learn how to use Page Designer to create a Web input page of a subfile. This new input page will pass the customer number back to the inquiry page. You also learn how to modify the inquiry page by adding a link (Find button) that will open the customer list page.

The third module, Adding the iSeries program to your Web application explains how to select the RPG service program to handle the Web interaction you created in the first module.

The fourth module, Creating the customer update output page, helps you learn how to create an output page using the Web Interaction wizard. This is the page that shows the customer information you can update.

The fifth module, Creating the update complete output page, helps you learn how to create another output page using the Web Interaction wizard. This is the page that shows the customer record was updated on the iSeries.

The sixth module, Running the Web application explains how to run the updated Web application in the WebSphere Test Environment.

Be sure to read about the Prerequisite knowledge for the tutorial before beginning. This section describes the type of knowledge you need to already have to fully benefit from this tutorial.

Prerequisite knowledge

In order to complete this tutorial end to end, you should already completed the tutorial called Building an RPG e-business application and have working knowledge of the following:

- Basic Microsoft® Windows® operations such as working with the desktop and basic mouse operations such as opening folders and performing drag-and-drop operations.
- How Web applications work.
- How to use a browser to navigate the Internet.

It is also useful, but not necessary, for you to have basic knowledge of the following:

- HTML
- Servlets and Java™ Server Pages (to understand the generated output of the Web Interaction wizard)

If you do not have current knowledge of these technologies and concepts, you might be interested in learning more about these after you complete this tutorial. You can find some additional information on these topics in the Help system that is part of WebSphere Development Studio Client for iSeries. There are also plenty

of resources on the Internet at our product Web page
<http://www.ibm.com/software/awdtools/iseries>.

When you are ready to begin, start with Module 1: Creating the customer list page.

Chapter 1. Module 1: Reviewing the existing RPG e-business application

In this module, you will review the existing RPG e-business application that you built in the tutorial called Building an e-business RPG application. You will then see what this application will look like when you have completed this tutorial.

In order to gain an introductory understanding of the customer inquiry application that you will add on to in this tutorial, there are several steps you will need to learn about and follow:

- Describe the Web customer inquiry application
- Describe the enhanced Web customer inquiry application

In order to accomplish these learning objectives, there are several steps that are involved, including:

Exercise 1.1: Reviewing the Web customer inquiry application

Exercise 1.2: Introducing the Web customer inquiry application enhancement

The exercises in this module must be completed in order. Start with Exercise 1.1 when you are ready to begin.

Length of time:

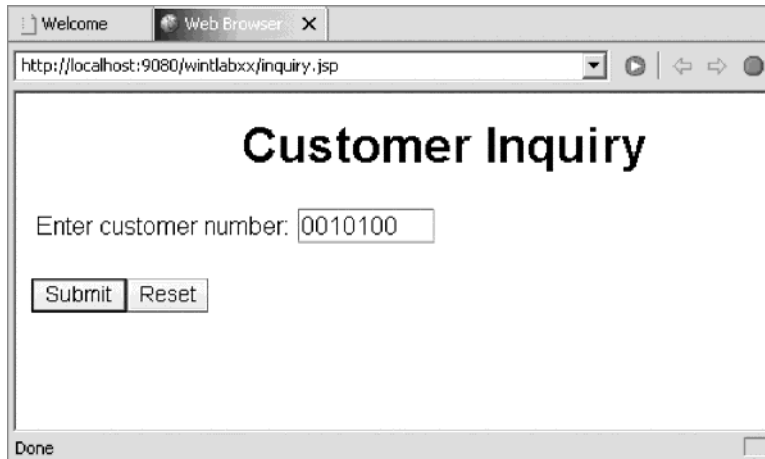
This module will take approximately 10 minutes to complete.

Exercise 1.1: Reviewing the Web customer inquiry application

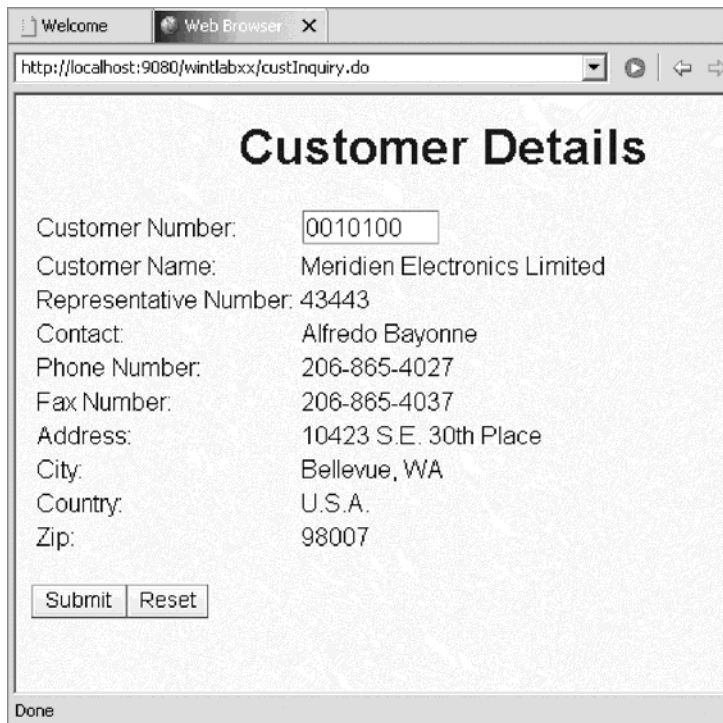
In the tutorial called Building an e-business RPG application you created a simple e-business customer inquiry application that uses a Web-based front end to communicate with RPG business logic residing on an iSeries server. You created a representation of a Web interaction in a Web diagram. You then invoked the Web Interaction wizard to create the input and output pages and to create the servlet to invoke the RPG program or service program to get data from the iSeries database. Next you added this RPG program or service program to the Web application. Finally you tested the application in the local test environment on your workstation. You also created an informative error page for when a customer entered an incorrect customer number.

Here is what the finished e-business customer inquiry application looked like.

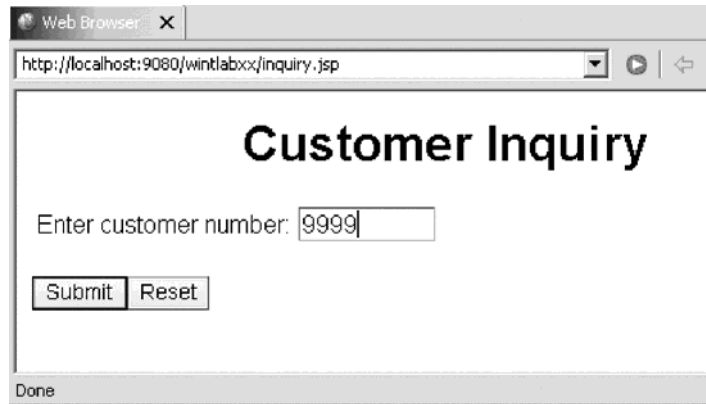
After the WebSphere server started, your Web application Customer Inquiry input page (inquiry.jsp) displayed in the workbench browser:



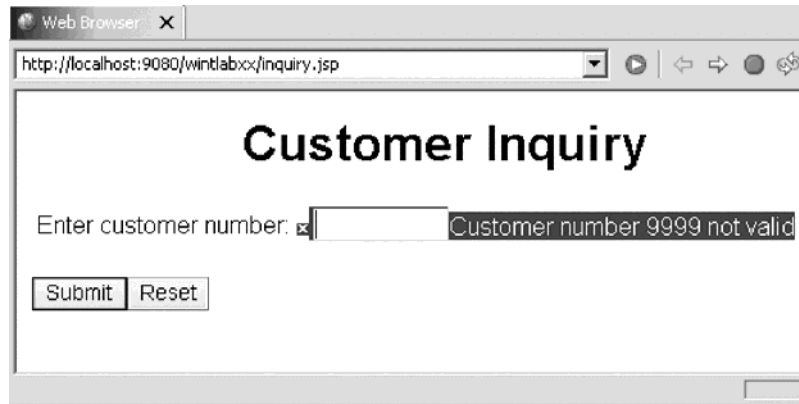
When you enter a customer number and click the **Submit** push button, the Customer Details output page (result.jsp) appears:



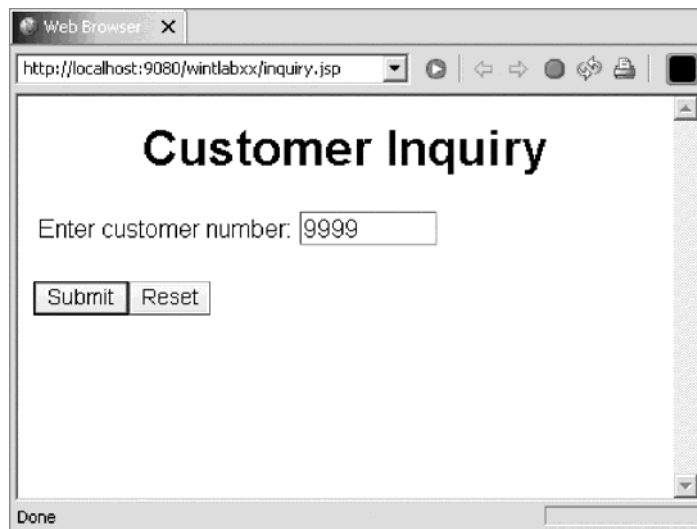
Next, you clicked the Back button in the browser to return to the Customer Inquiry input page. Here you entered an incorrect customer number and clicked **Submit**.



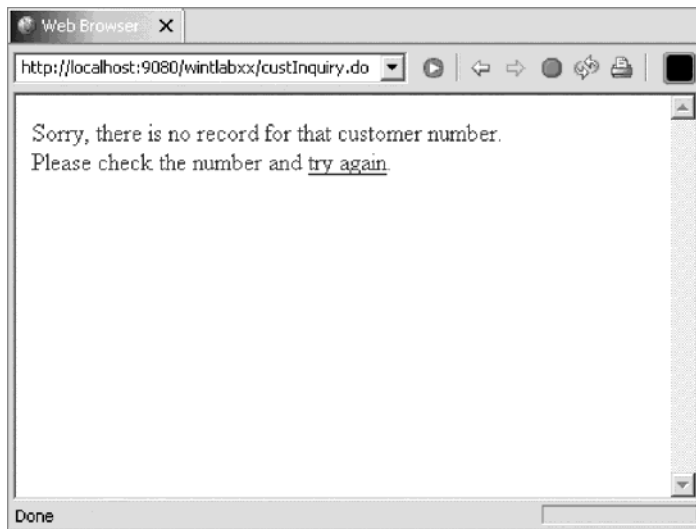
An error message appeared.



You then added a new output error page to improve the message information so that if a wrong customer number was entered,



an error page opened instead of a message.



Exercise 1.2: Introducing the enhanced Web application

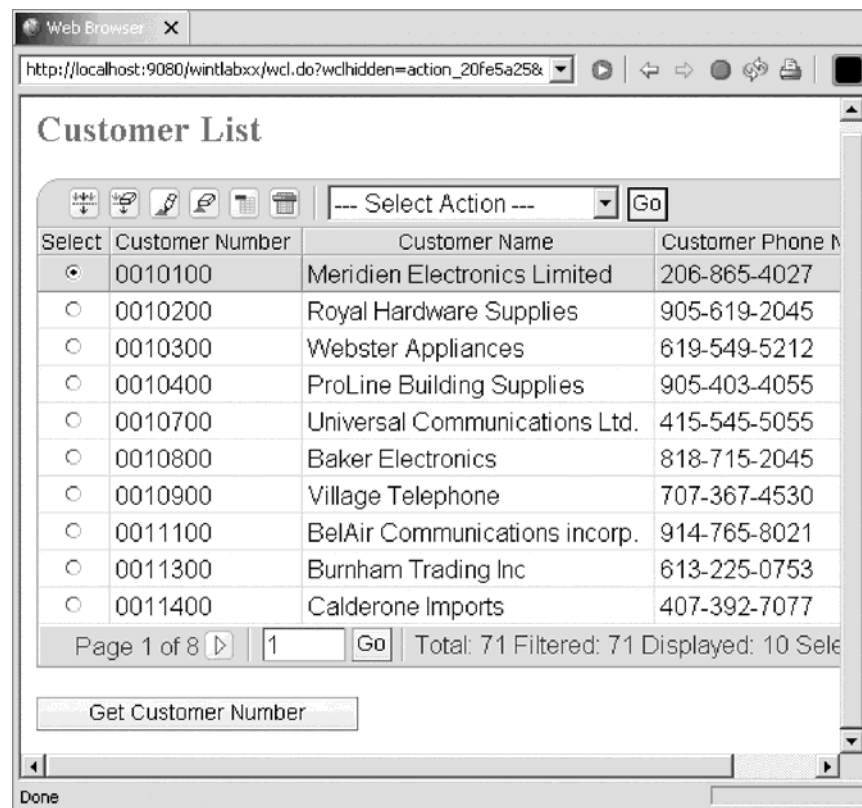
By the end of this tutorial you will have enhanced the Web customer inquiry application. You will invoke the Web Interaction wizard to first create a new input page to view a list of customers (custList.jsp). Next you will add an iSeries RPG service program to handle this Web interaction that you just created. Then you will invoke the Web Interaction wizard to use the existing output page (result.jsp) to create new output pages to update customer records and confirm updates were completed on the iSeries (custDispUpdateResults.jsp and custDoUpdateResults.jsp). You will then run the application in the local test environment on your workstation. You will also create an informative error page if there is an error during the update of a customer record.

Here is the completed e-business customer inquiry application.

This is the Customer Inquiry input page.

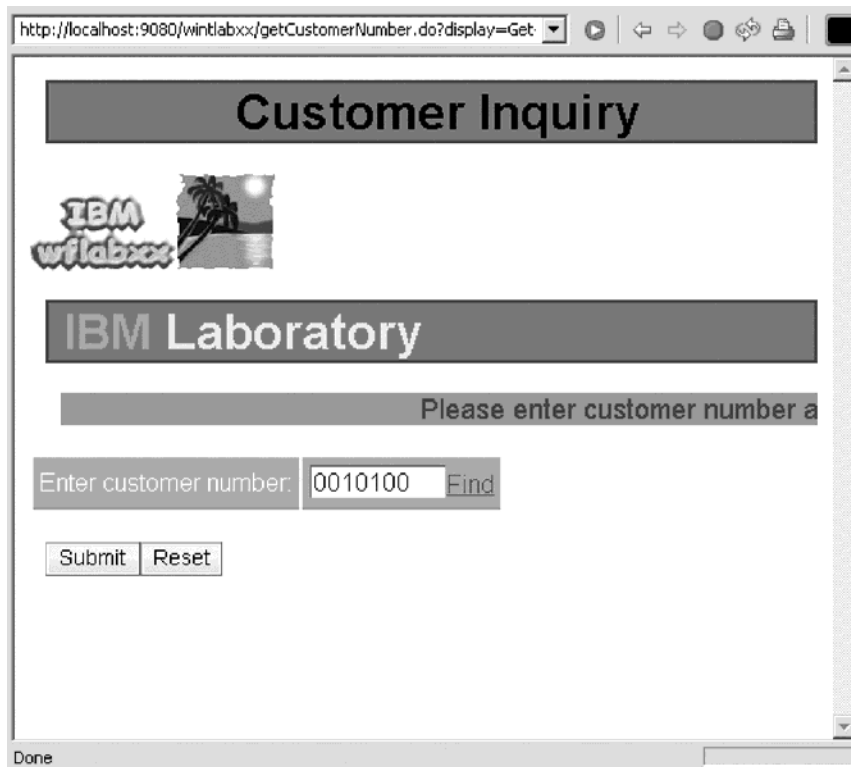


You click the **Find** link. The Customer List page opens:



You select the first entry on the list and then click the **Get Customer Number** button. You return to the Customer Inquiry page with the customer number

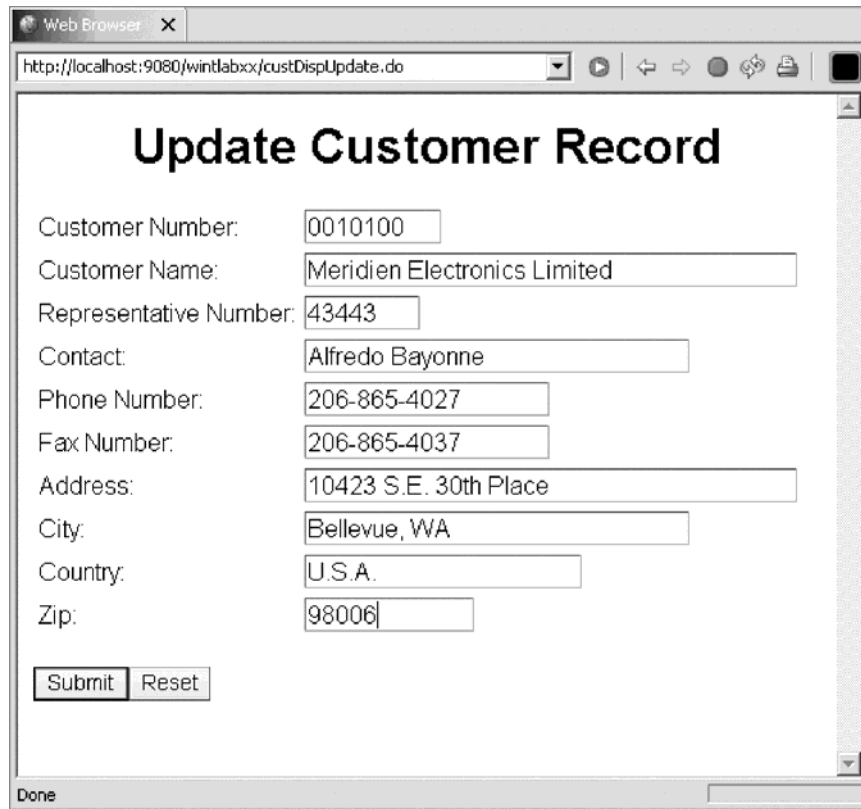
entered into the customer number field.



Next you click the **Submit** button. The Customer details page opens showing the details of the selected customer.



When the information is displayed, you click the **Update Customer Record** button.

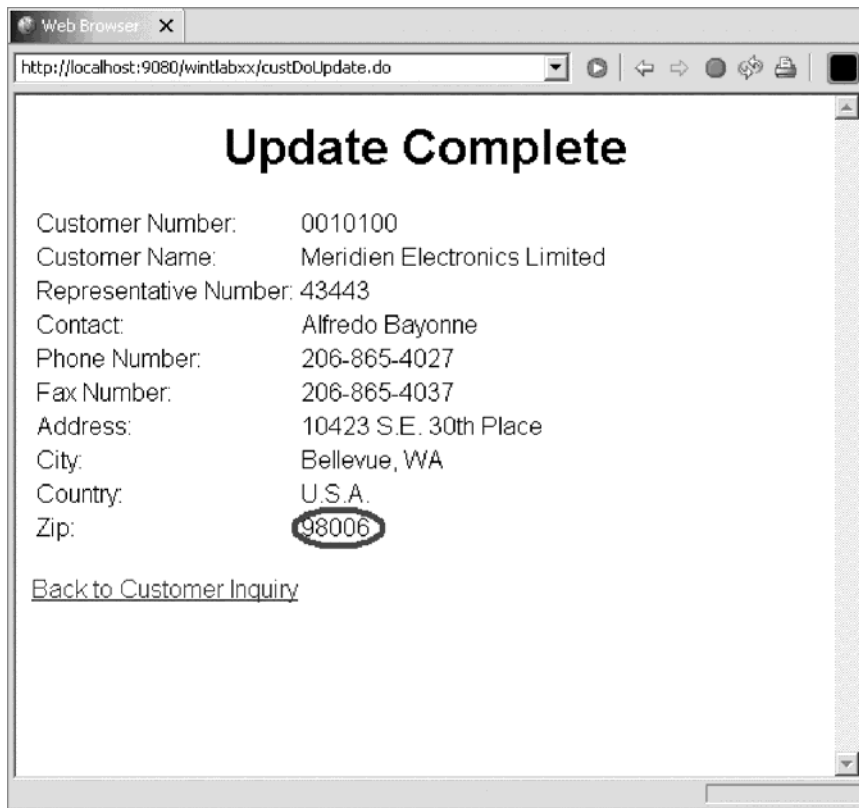


The screenshot shows a web browser window with the address bar containing `http://localhost:9080/wintlabxx/custDispUpdate.do`. The page title is "Update Customer Record". The form contains the following fields and values:

Customer Number:	0010100
Customer Name:	Meridien Electronics Limited
Representative Number:	43443
Contact:	Alfredo Bayonne
Phone Number:	206-865-4027
Fax Number:	206-865-4037
Address:	10423 S.E. 30th Place
City:	Bellevue, WA
Country:	U.S.A.
Zip:	98006

At the bottom of the form, there are two buttons: "Submit" and "Reset". The browser status bar at the bottom left shows "Done".

You change some of the fields and click the **Submit** button. You see the confirmation page with the new values.



You then click the **Back to Customer Inquiry** link and you return to the Customer Inquiry page.

Chapter 2. Module 2: Creating the customer list input page

In this module, you will learn how to create a new input page from a subfile and link that new input page to the customer inquiry input page. You will also learn how to add the iSeries business logic (service program) and additional parameter to the existing Web application in order to process the request to get a customer number. Remember: Before beginning this module, you should learn about the Prerequisite knowledge for the tutorial.

In order to create the customer list input page, there are several steps you will need to learn about and follow:

- Starting Development Studio Client
- Using Page Designer to create the customer list input page from a subfile
- Linking the inquiry page to the customer list page
- Creating another Web interaction to use the new input page
- Adding the iSeries service program (getCustNumber) to the Web interaction
- Adding a new parameter (cnumber) for the iSeries service program to pass the customer number back to your new input page

In order to accomplish these learning objectives, there are several steps that are involved, including:

Exercise 2.1: Starting the Development Studio Client workbench

Exercise 2.2: Opening the Web perspective

Exercise 2.3: Creating the customer list page

Exercise 2.4: Linking to the customer list page

Exercise 2.5: Creating another Web interaction

Exercise 2.6: Adding an iSeries ILE program to the interaction

Exercise 2.7: Adding a parameter

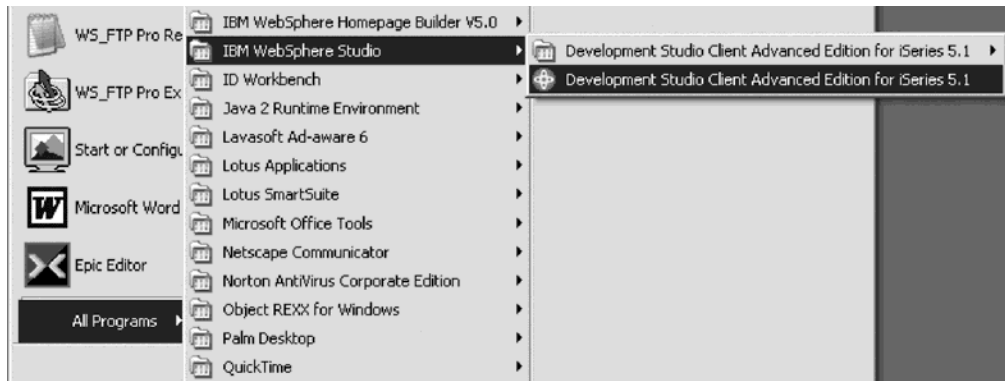
The exercises in this module must be completed in order. Start with Exercise 1.1 when you are ready to begin.

Length of time:

This module will take approximately 15 minutes to complete.

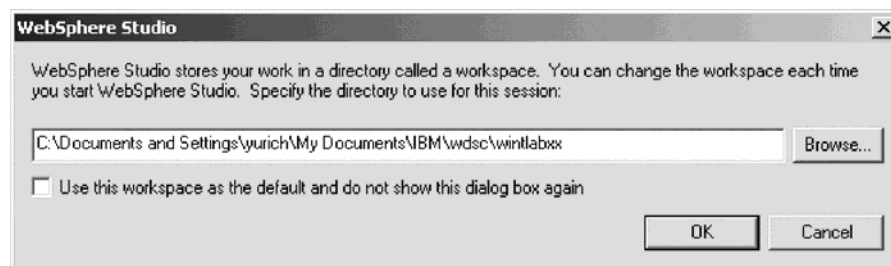
Exercise 2.1: Starting the Development Studio Client workbench

Now go ahead and start Development Studio Client.



To start Development Studio Client:

1. On the desktop task bar, click: **Start > All Programs > IBM® WebSphere Studio > Development Studio Client Advanced Edition for iSeries 5.1.**



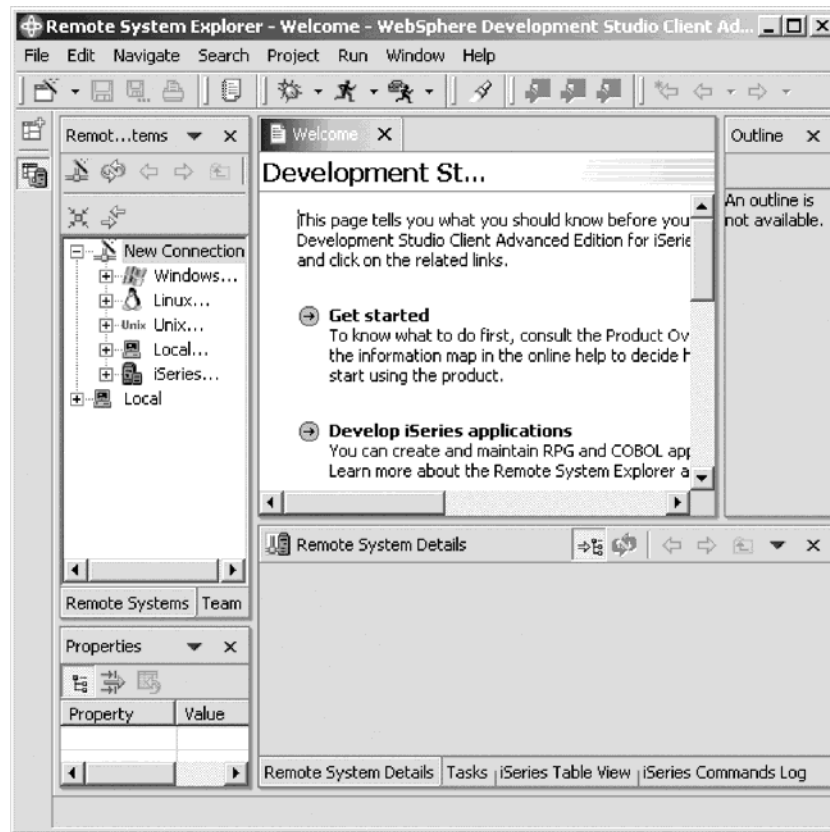
A dialog for workspace selection will appear asking you for the workspace location, unless you used the Development Studio Client before and selected not to show this dialog again.

2. Name the directory of the workspace as shown above. Use the directory name `wintlabxx`. This is the same directory that you used for the tutorial called Building an e-business RPG application.

Note: You must use the same workspace used for the previous tutorial as you need to access an existing Web project and Web application that you created in the previous tutorial called Building an e-business RPG application.

3. Click **OK**.

After a few moments of loading, the workbench opens, and the initial window of Development Studio Client appears:

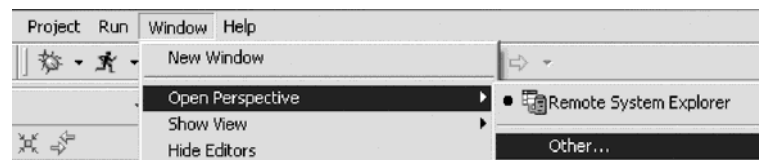


Exercise 2.2: Opening the Web perspective

In this exercise you will use the Web perspective since it gives you access to unique views and tools targeted towards Web tasks.

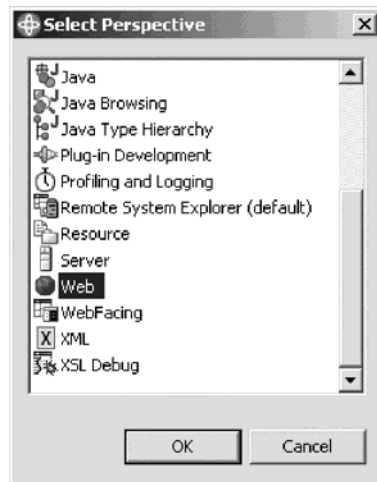
You first need to open the Web perspective to create a new customer list page. The perspective should already be open, but if not, check the workbench task bar on the left for the Web perspective icon, otherwise follow the steps below.

To open the Web perspective:



1. Click **Window > Open Perspective > Other** from the workbench menu.

The Select Perspective dialog opens:



2. Select **Web** from the list.
3. Click **OK**.

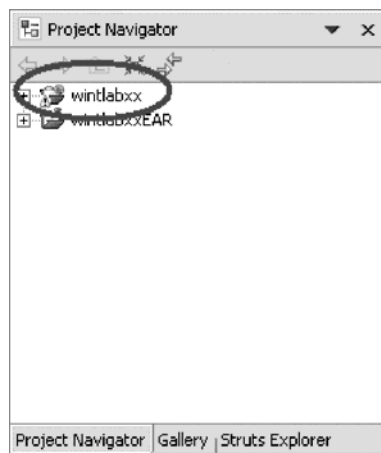
The workbench shows the Web perspective with the Web Projects open in the Project Navigator.

You are now ready to create the customer list page (custList.jsp) which will pass the customer number to the inquiry page (inquiry.jsp). You also need to modify the inquiry page by adding a link (Find button) that will open the customer list page.

Exercise 2.3: Creating the customer list page

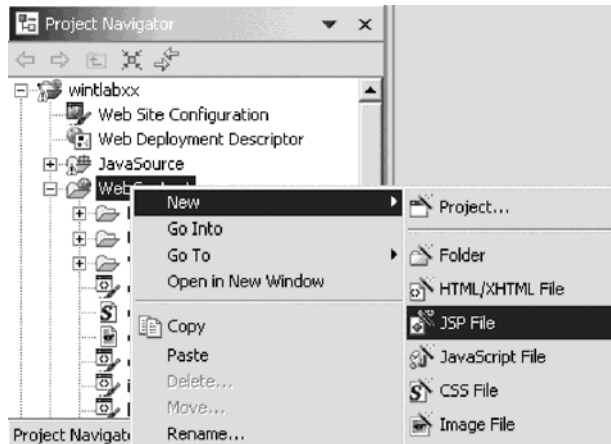
You will create the customer list page using several Page Designer iSeries Web components.

To create a new customer list file:

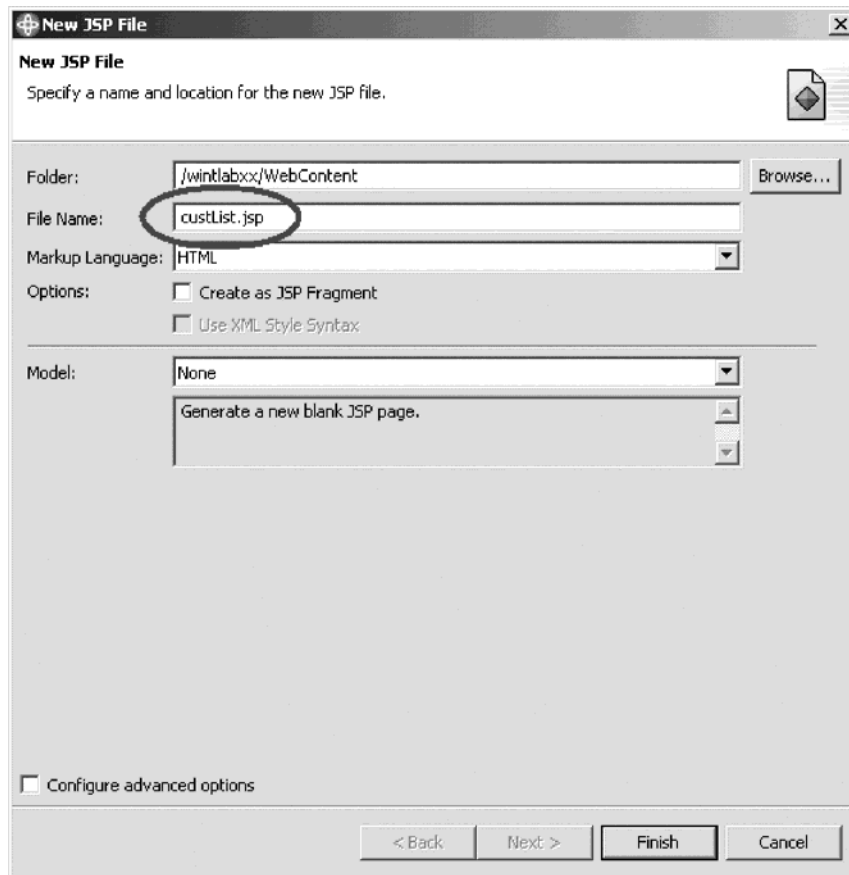


1. Locate the project wintlabxx that you created in the previous tutorial.
2. Expand the wintlabxx folder, if not already expanded.
3. Navigate to the WebContent folder in your project.

- Right-click the WebContent folder and click **New > JSP File** on the pop-up menu.

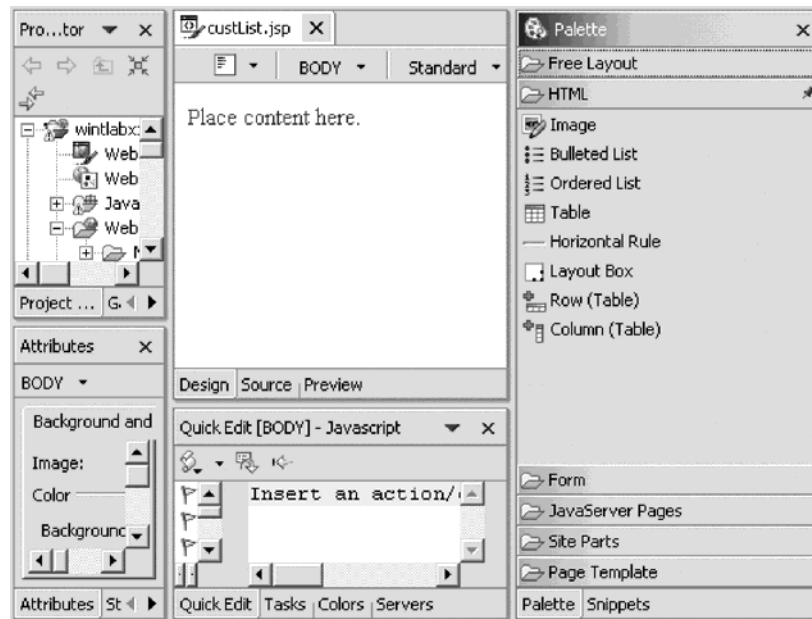


The New JSP File page opens:



- In the **File Name** field, type `custList.jsp`.
- Click **Finish** to create a new JSP file.

The Page Designer opens in the workbench:



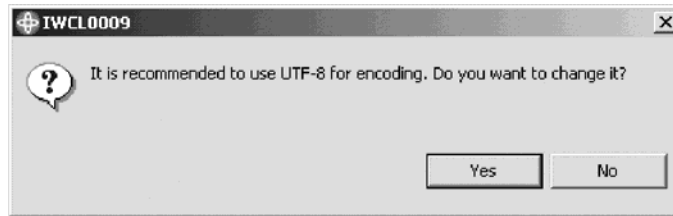
To add the iSeries table component:

1. Select the **Snippets** tab in the bottom right of the workbench.

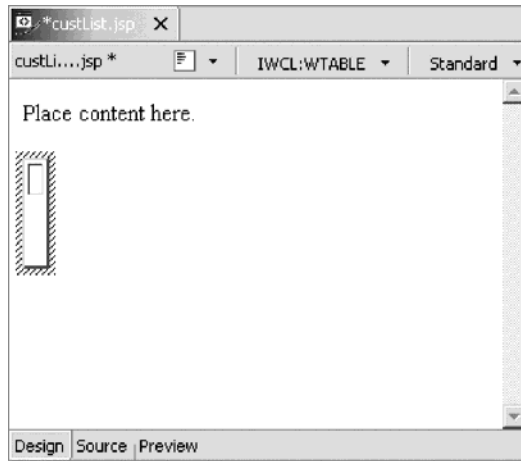


2. Expand **iSeries Web Components**.
3. Select **Table** and drag it to your JSP under the text "Place content here".
Make sure the Design page is selected to place the table component on the new custList.jsp page.
You have selected the **iSeries Table Component**.

4. If you see this dialog, click **Yes**.



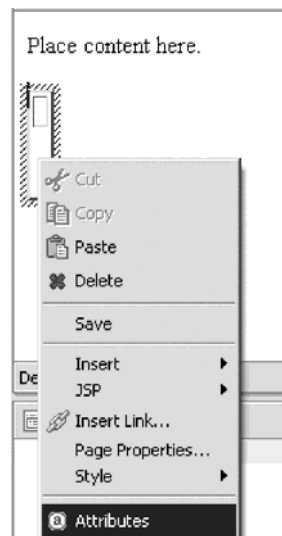
The Page Designer should look like this where the iSeries Table is on the Designer page.



Now you select the attributes for the table.

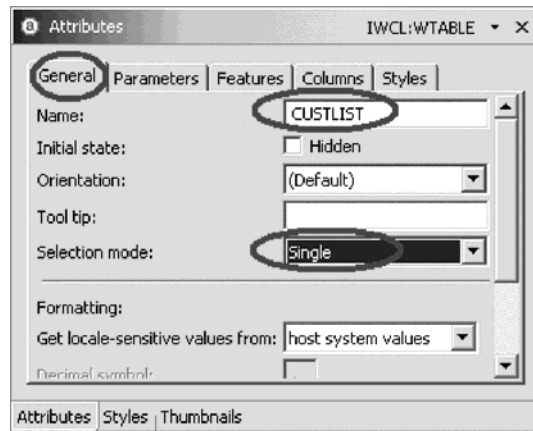
To select attributes for the table:

1. The Attributes pane in the bottom left of the workbench should open automatically. If not right-click on your table object and click **Attributes** on the pop-up menu.

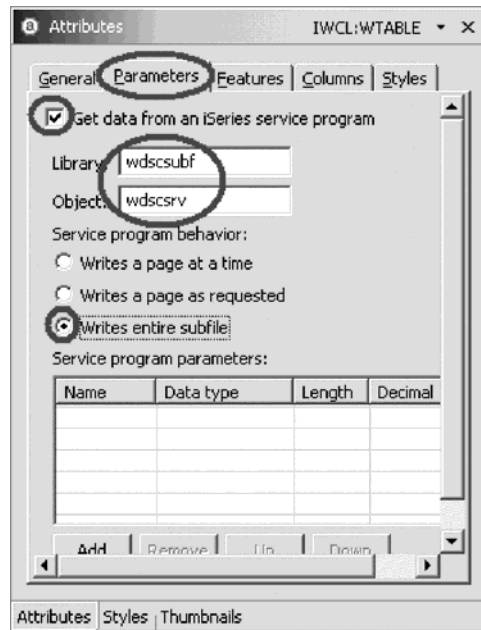


Next you fill in the general attributes for the iSeries Table.

- In the Attributes pane, select the **General** tab.

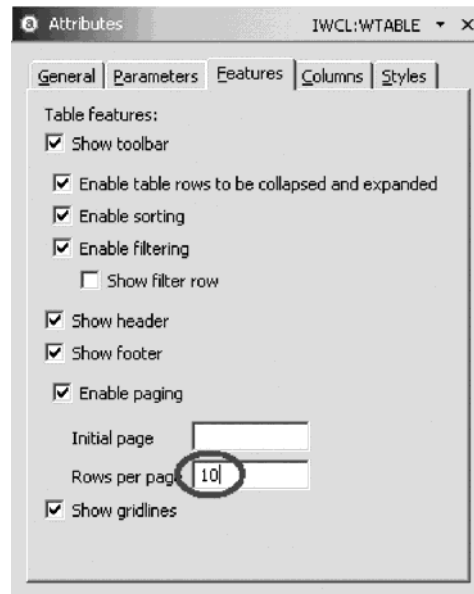


- In the **Name** field, type CUSTLIST.
- Select **Single** from the **Selection mode** list.
- Click the **Parameters** tab to set the parameters for the iSeries Table.



- Select the **Get table data from an iSeries service program** check box.
 - In the **Library** field, type wdscsubf.
 - In the **Object** field, type wdscsrv.
 - Click the **Writes entire subfile** radio button.
- Next you set the features for the iSeries Table.

10. Click the **Features** tab.

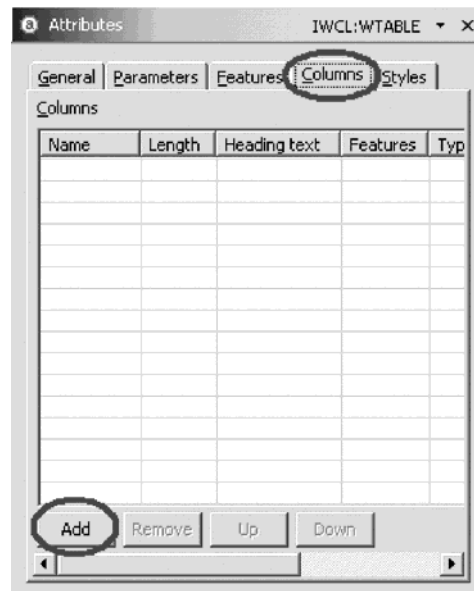


This will switch to the Features Attributes pane.

11. In the **Rows per page** field, type 10.

12. Click the **Columns** tab.

This will switch to the Column Attributes pane where you set the columns for the iSeries Table.

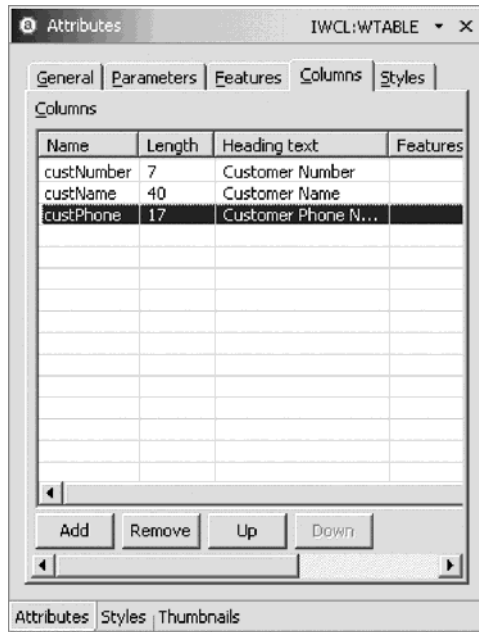


13. Click **Add**.

14. Fill in the following, and click **Add** after each row.

NAME	LENGTH	HEADING TEXT
custNumber	7	Customer Number
custName	40	Customer Name
custPhone	17	Customer Phone Number

Your completed Attributes pane for your table columns should look like this:



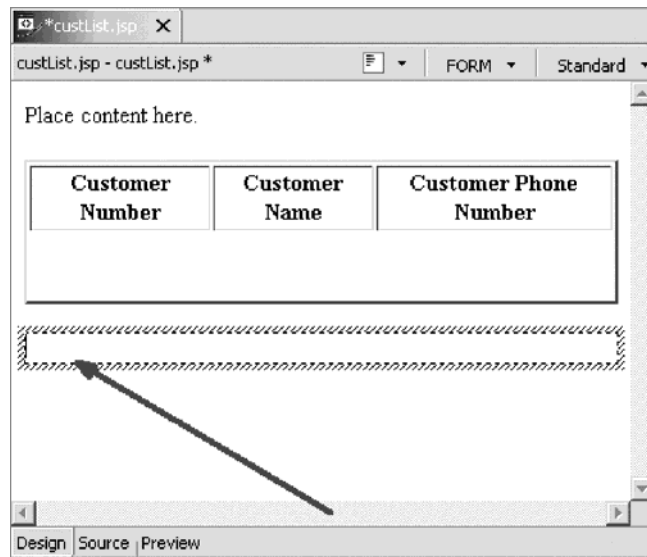
Now insert a form on the page.

To insert a form on the page:

1. Place your cursor below the table in the Page Designer (if not already), right-click and then click **Insert > Form and Input Fields > Form** on the pop-up menu.



2. Place your cursor inside the dotted rectangle, which represents the form that you just created.



Next you want to access the Snippets view again to select an **iSeries Button Component**.

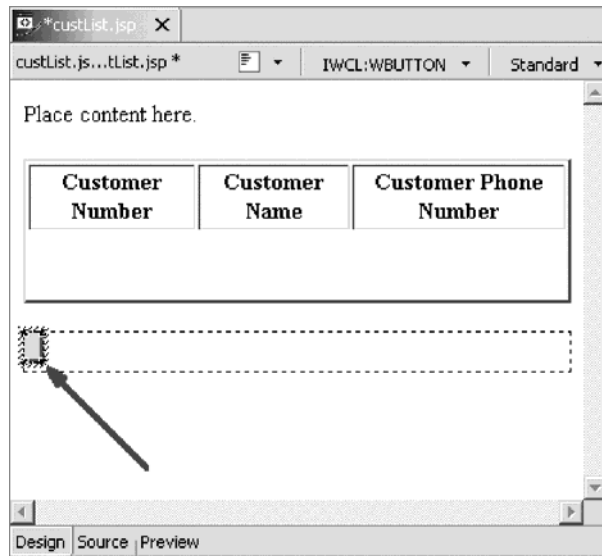
To add the iSeries Button Component:

1. Navigate back to the Snippets view. Select the **Snippets** tab at the bottom right of the workbench:



2. Expand **iSeries Web Components** if not already.
3. Select **Button** then drag it to the form and drop it.

You have placed the iSeries Button inside the form on the page.



Next you complete the iSeries Button attributes.

To add iSeries Button attributes:

1. The Attributes pane for the button should automatically show up; if not, right-click the button and click **Attributes** on the pop-up menu..



2. In the **Name** field, type display.
3. In the **Label** field, type Get Customer Number.

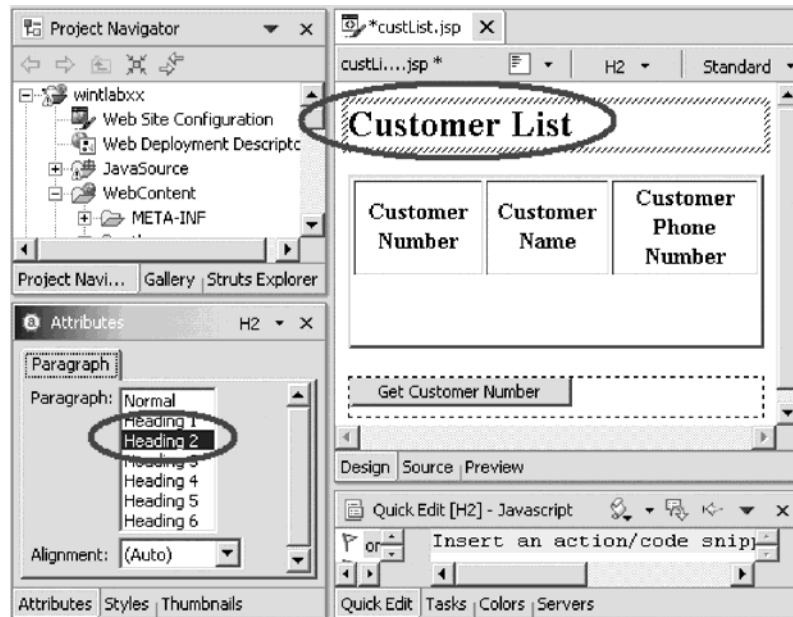
You have finished creating and modifying the iSeries Web Components.

Let's add a heading to this page.

To add a heading to this page:

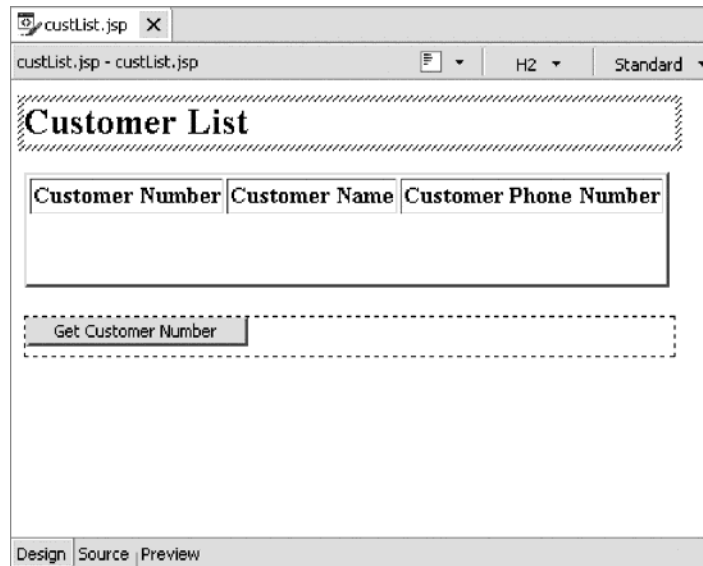
1. Select the text Place content here.
2. Replace it with the text Customer List.

3. In the Attributes pane on the bottom left of the workbench, select **Heading 2**.



You are finished creating the JSP.

Your finished page should look like this:



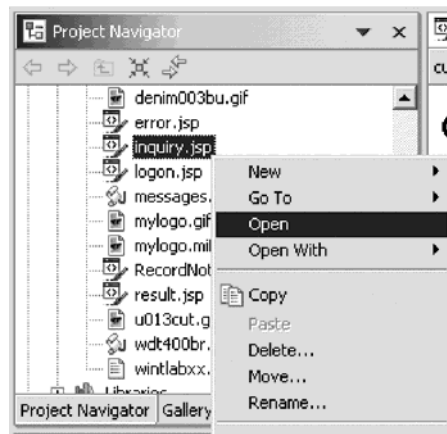
4. If you haven't done so already, save the JSP file (**Ctrl + S**).

Exercise 2.4: Linking to the customer list page

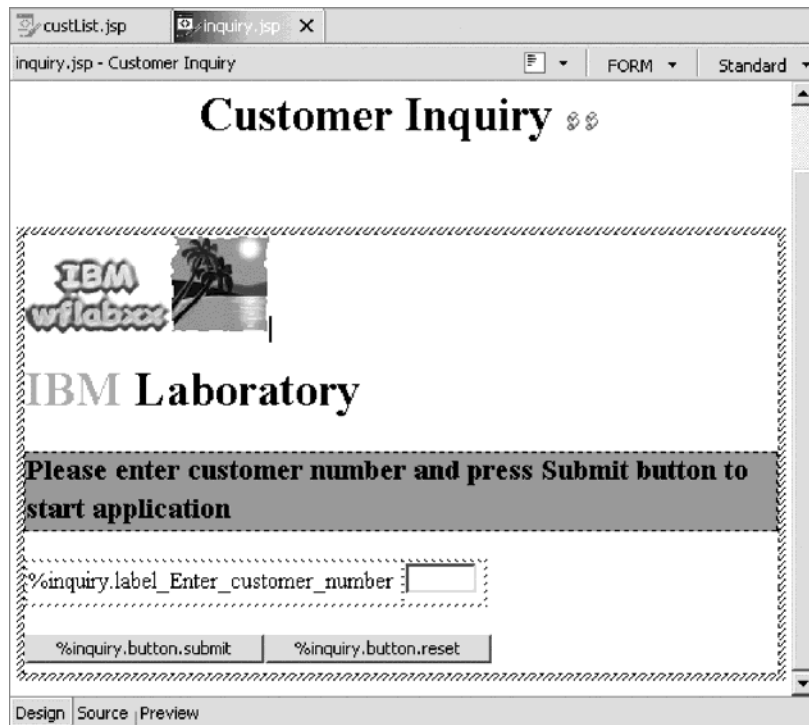
Now, let's modify the inquiry page so that you can link (Find button) to the customer list page.

1. In the Project Navigator, right-click inquiry.jsp.

2. Click **Open** on the pop-up menu to edit the inquiry page named inquiry.jsp.

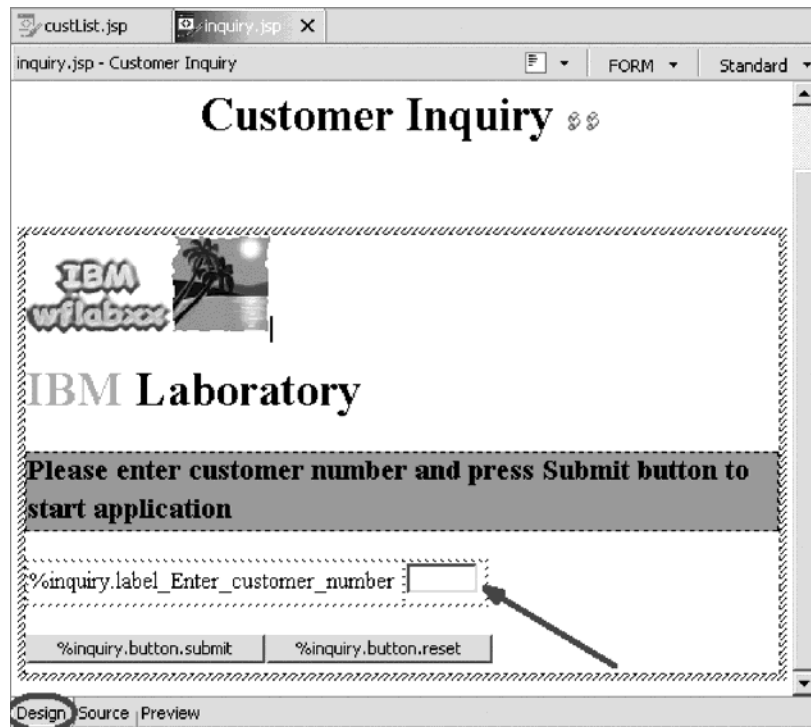


The Page Designer opens in the upper right pane of the workbench and shows the inquiry.jsp as the Web Interaction wizard created it and with the changes you made in Page Designer.

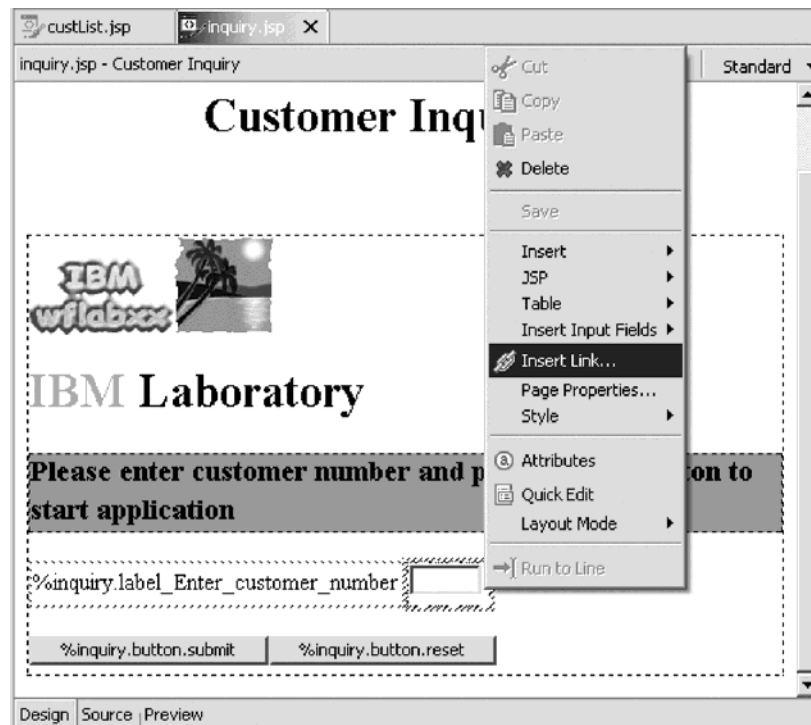


3. Make sure that you are on the Design page in Page Designer. Select the **Design** tab.

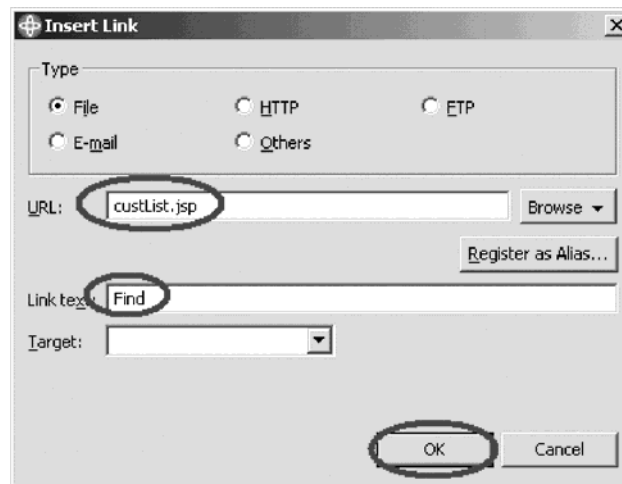
4. Now put the cursor after the text field as shown:



5. Right-click the page and click **Insert Link** on the pop-up menu.

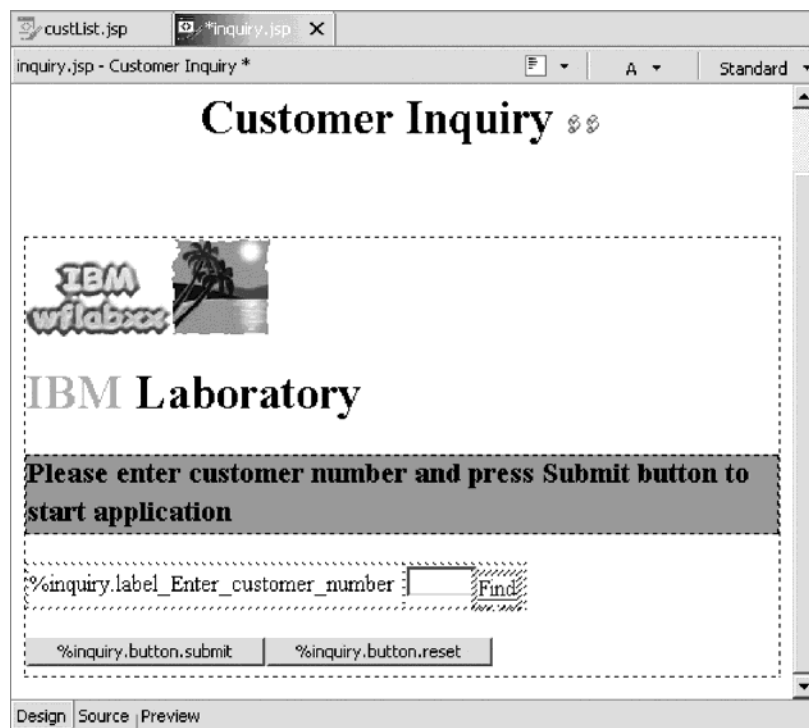


The Insert Link dialog opens:



6. In the **URL** field, type `custList.jsp`.
7. In the **Link text** field, type `Find`.
8. Click **OK**.

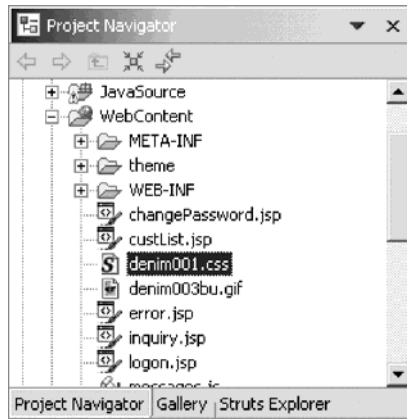
The Customer Inquiry page after adding a link to customer list page (`custList.jsp`) should look like this:



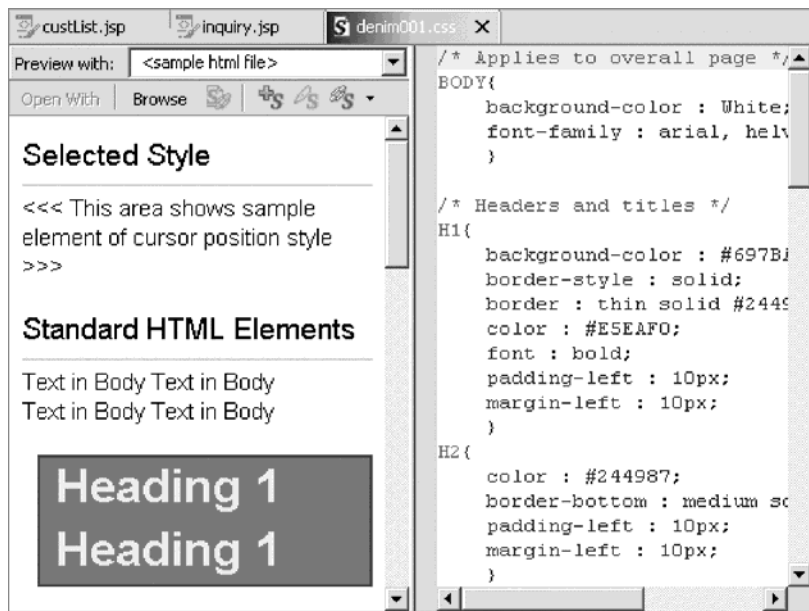
9. If you haven't done so already, save the JSP file (**Ctrl + S**).
Now that you've added the Find link, let's modify the style sheet so that the link does not blend with the background.

To modify the style sheet:

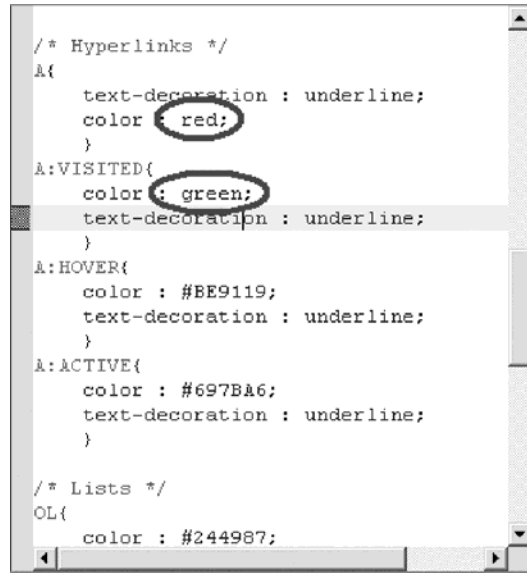
1. Locate the denim001.css file under wintlabxx > WebContent > WEB-INF.



2. Double-click denim001.css and you should see the following window:



3. In the right pane of this window, scroll down until you find the `/*hyperlinks*/` section.



```
/* Hyperlinks */
A{
  text-decoration : underline;
  color : red;
}
A:VISITED{
  color : green;
  text-decoration : underline;
}
A:HOVER{
  color : #BE9119;
  text-decoration : underline;
}
A:ACTIVE{
  color : #697BA6;
  text-decoration : underline;
}

/* Lists */
OL{
  color : #244987;
}
```

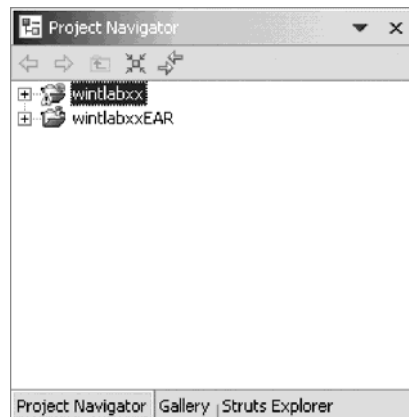
4. Type red beside color on the **A** section.
5. Type green beside color on the **A:VISITED** section.
6. Save the file (**Ctrl+S**).

Exercise 2.5: Creating another Web interaction

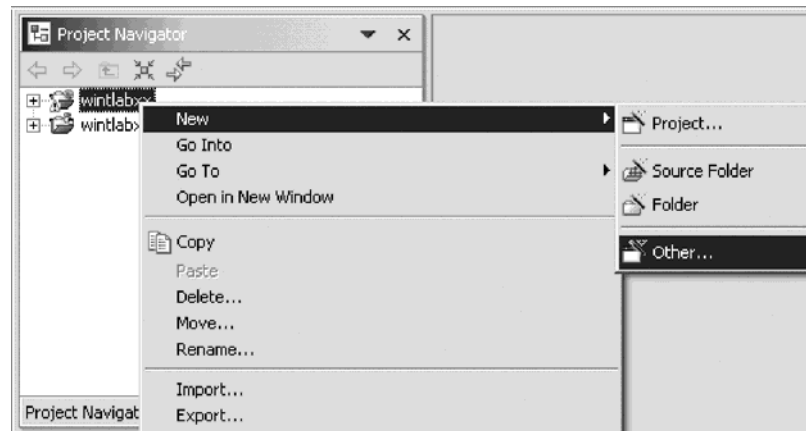
Next you create another Web interaction to use the new input page and output pages that you just created.

To create another Web interaction:

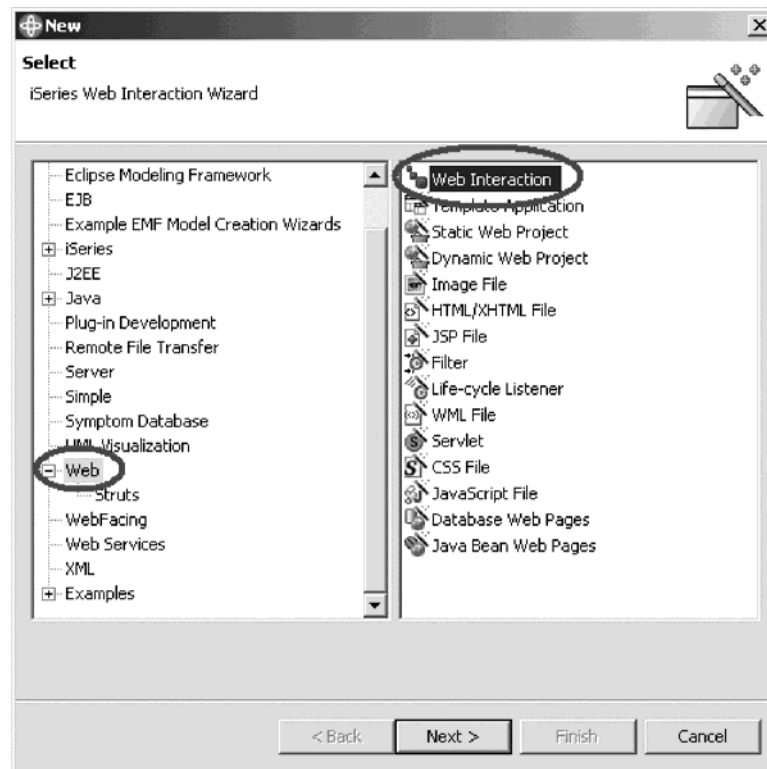
1. Locate your project from the previous tutorial in the Project Navigator.



2. Right-click on your project and click **New > Other** on the pop-up menu.



The Select page opens:



3. In the left pane of the Select page, select **Web**.
4. In the right pane of the Select page, select **Web Interaction**.
5. Click **Next**.

The Web Interaction wizard opens:

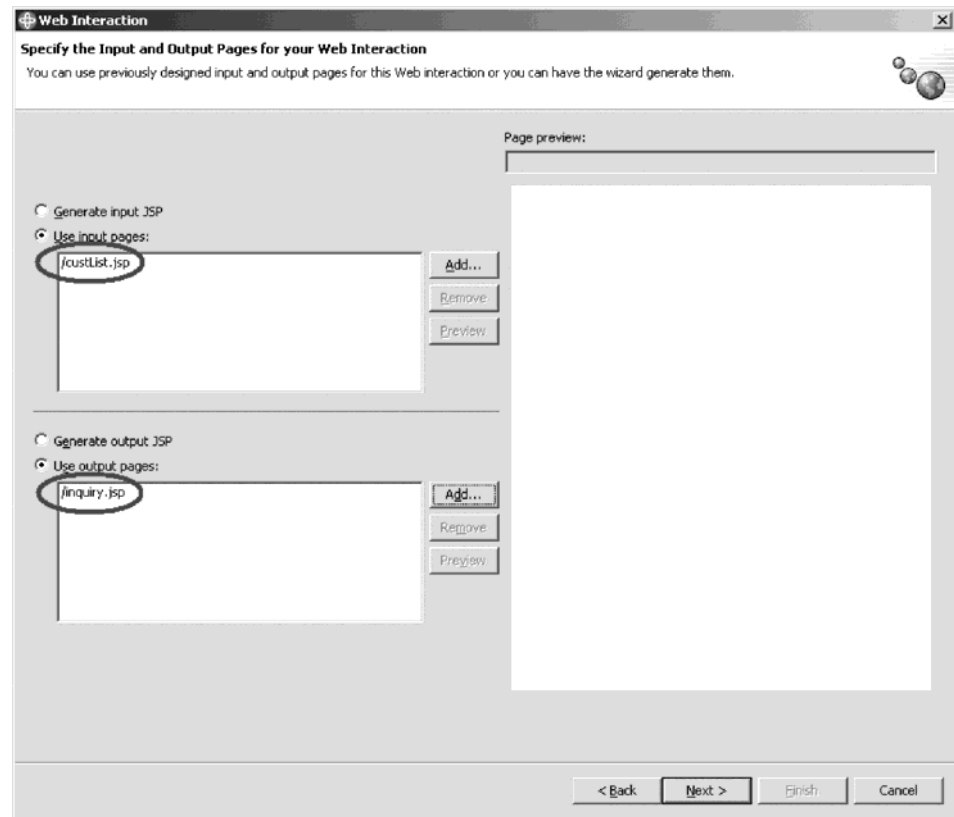
The screenshot shows a dialog box titled "Web Interaction" with the subtitle "Specify a Name and Location for your Web Interaction". Below the subtitle is a brief instruction: "Define the name and location for the communication between Web pages and the chosen business process. The location must be the Web Content directory inside a project." The dialog contains several input fields and checkboxes:

- Destination folder:** A text box containing "/winlaboc/WebContent" with a "Browse..." button to its right.
- Web Interaction name:** A text box containing "getCustomerNumber", which is circled in red.
- Java package:** An empty text box with a "Browse..." button to its right.
- Use error page:** A checked checkbox with a label "Use error page" circled in red. To its right is a text box containing "/error.jsp" with a "Browse..." button.
- Invalidate session after the interaction occurs:** An unchecked checkbox.

At the bottom of the dialog, there are four buttons: "< Back", "Next >", "Finish", and "Cancel".

6. In the **Web Interaction name** field, type getCustNumber.
7. Select the **Use error page** check box and select **error.jsp** on the Error page dialog. Click **OK**.
8. Click **Next**.

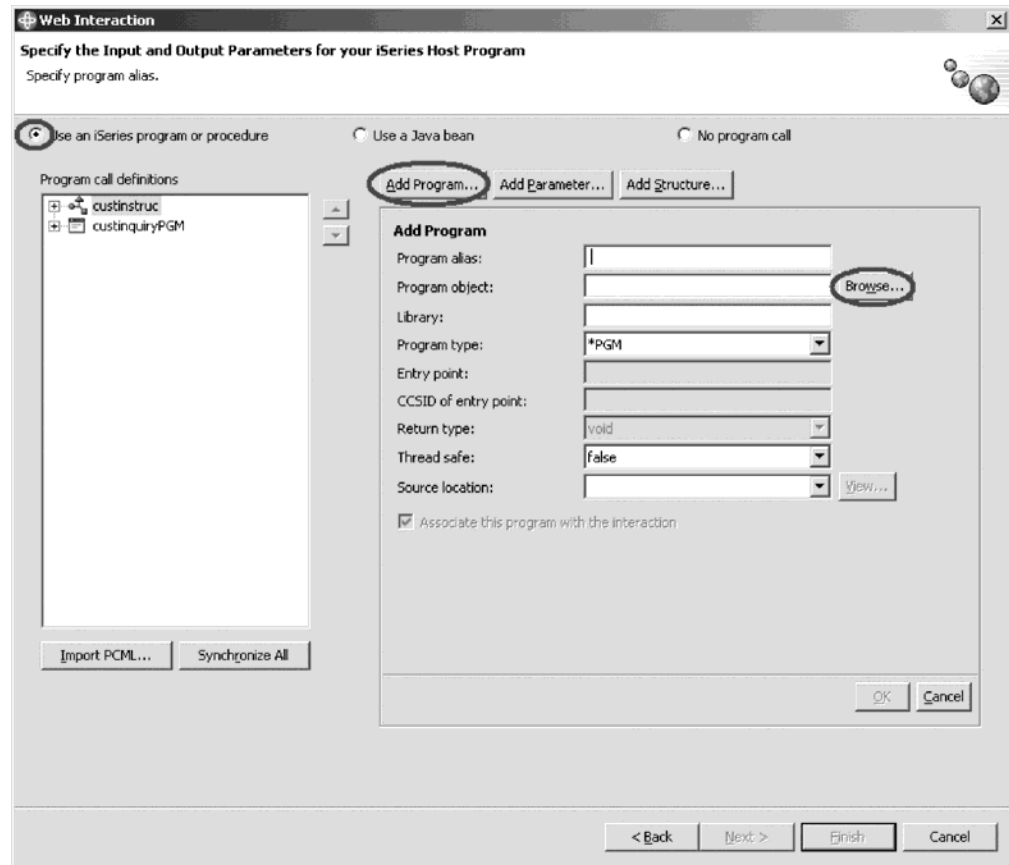
As you probably remember from the previous tutorial, this next page is where you specify the input and output pages.



9. Leave the **Use input pages** radio button selected and click **Add**.
10. Select **custList.jsp** on the Input page dialog. Click **OK**.
This is the subfile that contains the list of customers.
11. Leave the **Use output pages** radio button selected and click **Add**.
12. Select **inquiry.jsp** on the Output JSP dialog. Click **OK**.
13. Click **Next**.
You have now defined your input and output pages.

Exercise 2.6: Adding an iSeries ILE program to the interaction

Next you will select the RPG service program you will use to handle your Web interaction.



To add a program to the Web interaction:

1. Leave the **Use an iSeries ILE program** radio button selected.
2. Click **Add Program**.
3. Click **Browse** to browse for the service program you will use.

If you don't see your connection then you will need to create a connection.

Follow the steps under the section **Creating a connection**.

If you do see your connection then skip the following section and proceed to the section **Selecting the service program**.

Creating a connection

To create a connection:

1. In the list in the left pane, expand the **New Connection** list item by clicking on the plus sign beside its icon.
The **Name personal profile** page opens.
2. Accept the default profile. Click **Next**.

The Remote iSeries System Connection page opens:

New

Remote iSeries System Connection

Define connection information

Parent profile: YURICH

Connection name: s400a

Host name: s400a

Description:

Verify host name

< Back Next > Finish Cancel

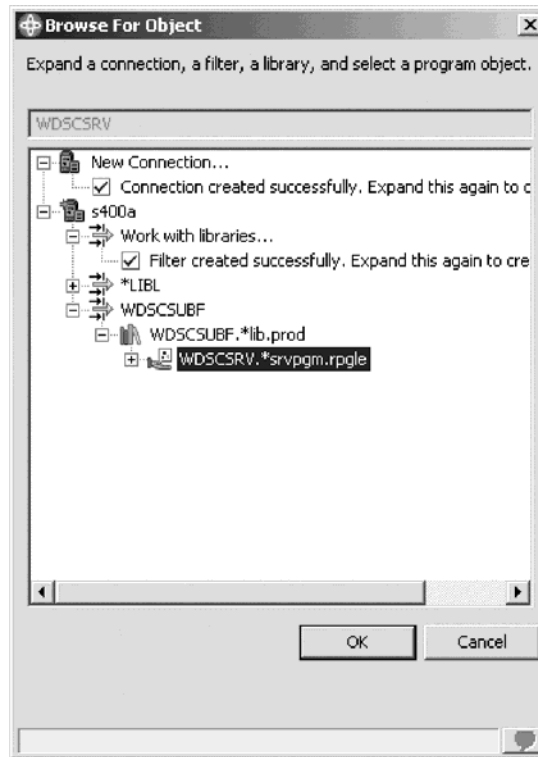
3. Leave the default value for the Parent Profile.
4. In the **Host Name** field, type the name of the iSeries server (Same server name as you specified for the runtime).
5. Click **Finish** to create a connection.

Selecting the service program

Now you select the RPG service program.

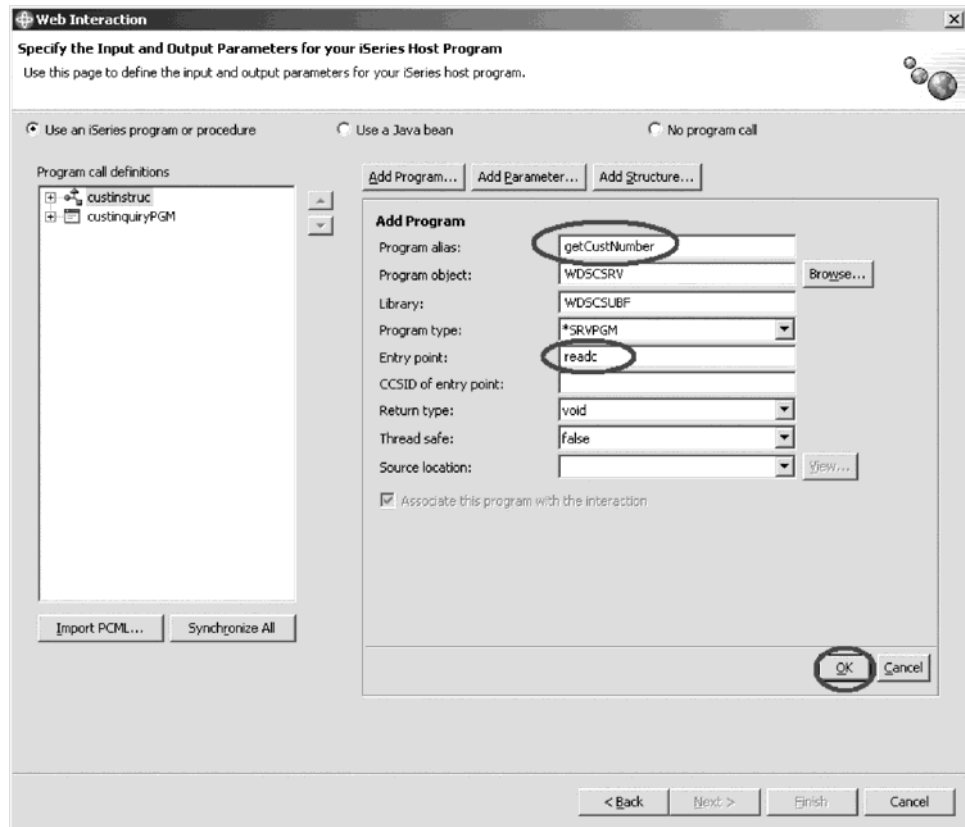
1. Expand your connection.
2. If you don't see the WDSCSUBF library, click **Work with libraries** and type WDSCSUBF in the dialog, then click **OK**.

- Expand the WDSCSUBF library and select the WDSCSRV service program.



- Click OK.

Most of the fields have now been filled in for you. However you must fill in the values for the program alias and entry point fields:



5. In the **Program alias** field, type getCustNumber.
6. In the **Entry point** field, type readc (case sensitive).
7. Click **OK**.

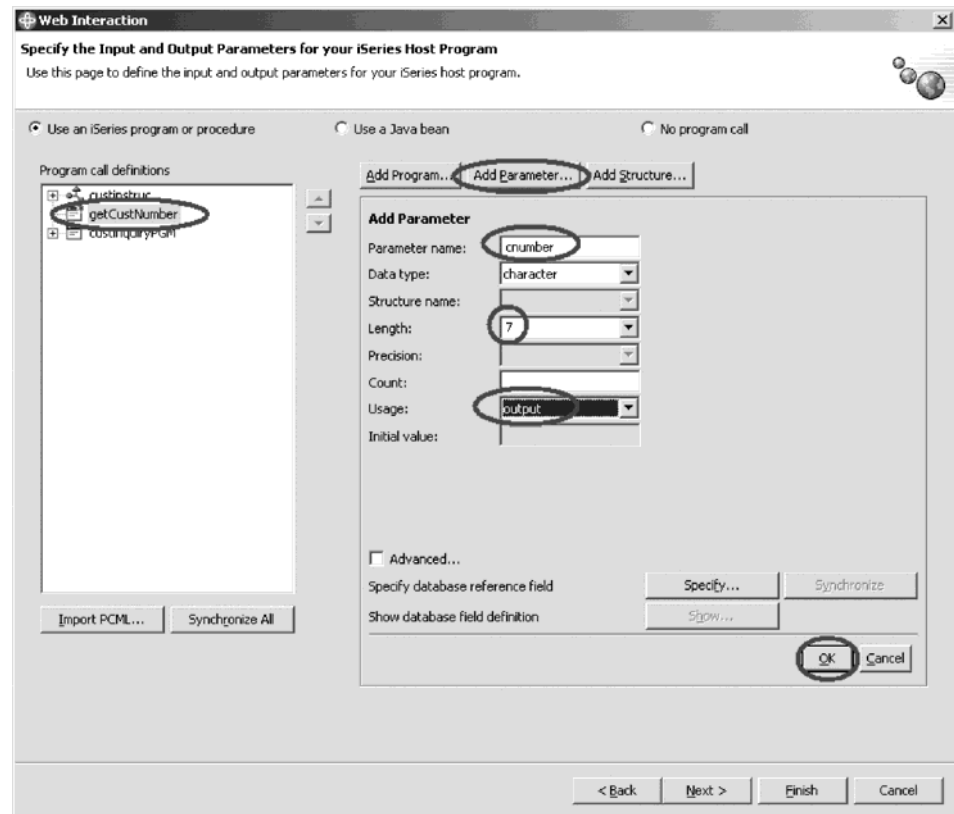
Exercise 2.7: Adding a parameter

Let's add this structure as a parameter to the service program that you just added in the previous exercise.

To add a parameter:

1. Under **Program call definitions**, select the getCustNumber program.
2. Click **Add Parameter**.
3. Leave **character** in the **Data type** list selected
4. In the **Parameter name** field, type cnumber.
5. In the **Length** field, type 7.
6. Select **Output** from the **Usage** list.

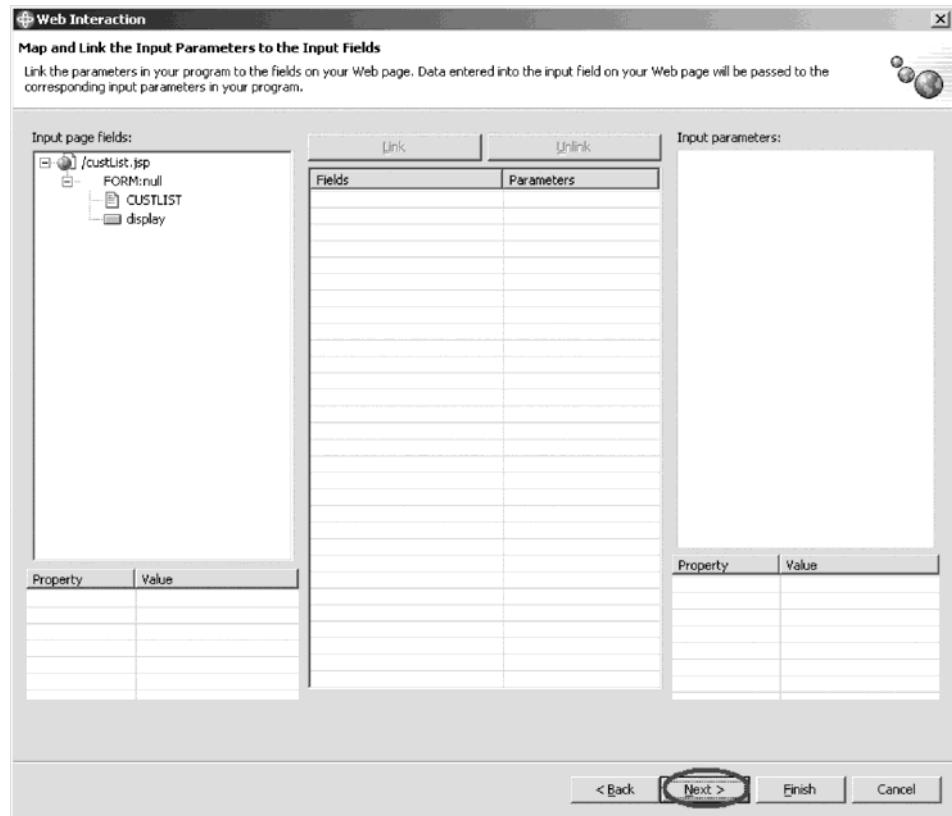
Your wizard should now look like this:



7. Click **OK**.

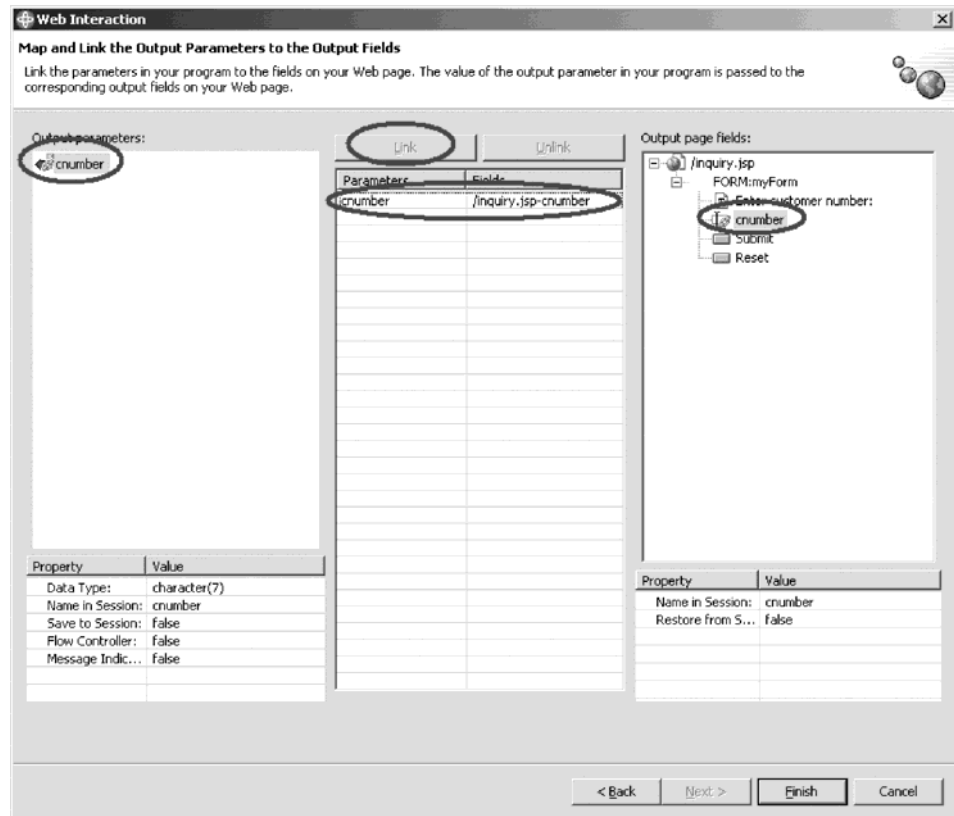
You have finished adding program parameters. Now you need to setup the JSP's.

8. Click Next.



9. You have no input parameters, so you don't need to modify this page.
10. Click Next again.

11. Under **Output parameters**, select **cnumber**.



12. Under **Output page fields**, expand **/inquiry.jsp > FORM:myForm** and select **cnumber**.
13. Click **Link** in the middle of the page.
14. Click **Finish** to link the output field to the output parameter.

Recap

Congratulations! You have completed Module 2: Creating the customer list page. You should now understand how to:

- Start Development Studio Client
- Use Page Designer to create the customer list input page from a subfile
- Link the inquiry page to the customer list page
- Create another Web interaction to use the new customer list input page
- Add the iSeries service program (getCustNumber) to the Web interaction
- Add a new parameter (cnumber) for the iSeries service program to pass the customer number back to your new customer list input page

Now that you have created a new input page from a subfile, linked that new input page to the customer inquiry input page and added the iSeries business logic (service program) and additional parameter to the existing Web application in order to process the request to get a customer number, you can continue to Module 3: Adding the iSeries program to your Web application.

Chapter 3. Module 3: Adding the iSeries program to your Web application

In this module, you will learn how to use the Remote System Explorer perspective to edit and compile an iSeries program.

In order to add the iSeries program to your Web application, there are several steps you will need to learn about and follow:

- Opening the Remote System Explorer perspective
- Editing an iSeries source member
- Compiling source changes

Note: To access iSeries data and resources you will need to have a job description setup and associated with your user profile on your iSeries system.

In order to accomplish these learning objectives, there are several steps that are involved, including:

Exercise 3.1: Opening the Remote System Explorer perspective

Exercise 3.1: Editing an iSeries source member

Exercise 3.3: Compiling source changes

The exercises in this module must be completed in order. Start with Exercise 2.1 when you are ready to begin.

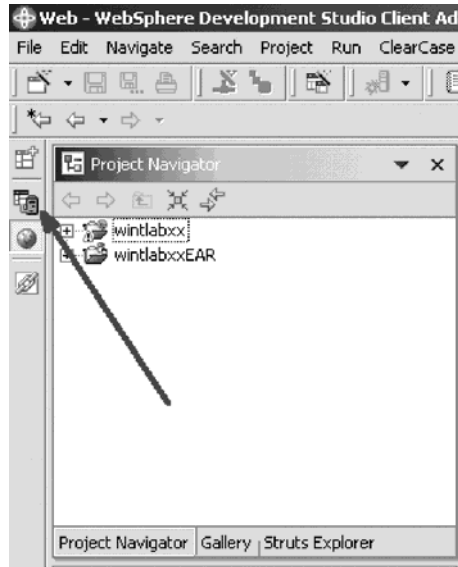
Length of time:

This module will take approximately 5 minutes to complete.

Exercise 3.1: Opening the Remote System Explorer perspective

To open the Remote System Explorer perspective:

1. In the workbench check whether you have the Remote System Explorer perspective already open. Look for the icon in the left taskbar of the workbench.



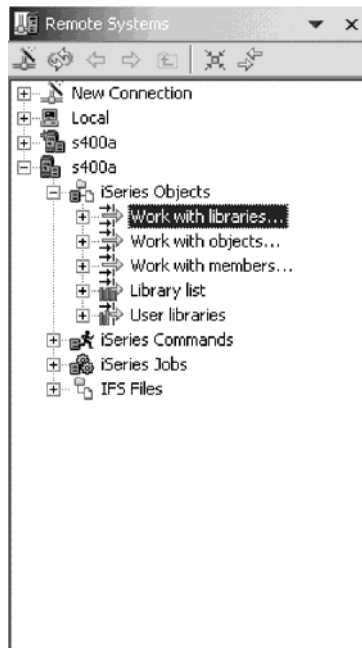
2. If you see it, click the Remote System Explorer icon.
3. Otherwise, click **Window > Open Perspective> Remote System Explorer** on the workbench menu.
The Remote System Explorer perspective opens.

Exercise 3.2: Editing an iSeries source member

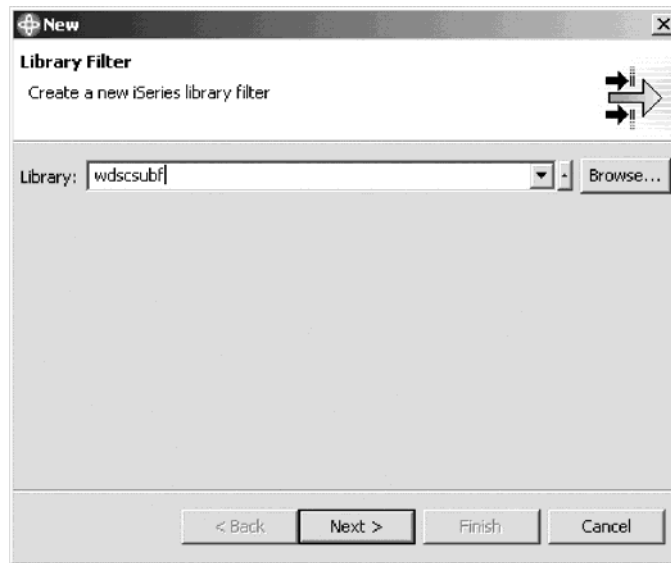
You will see a list of servers in the Remote Systems view.

To edit a member:

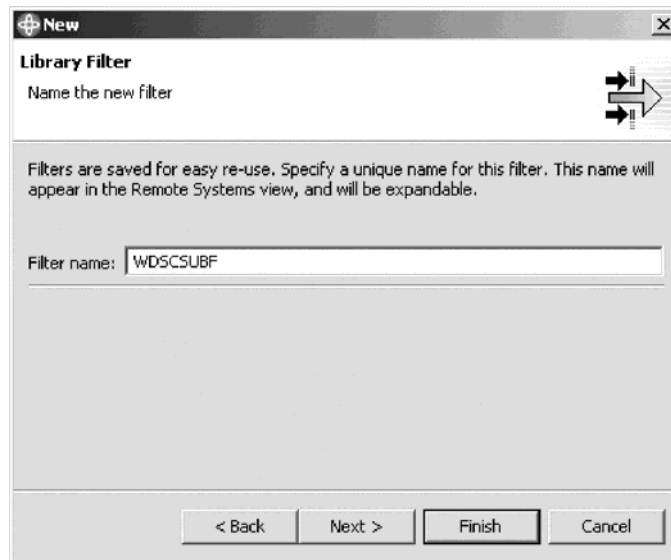
1. Double-click **Work with libraries**.



The Library Filter page opens.

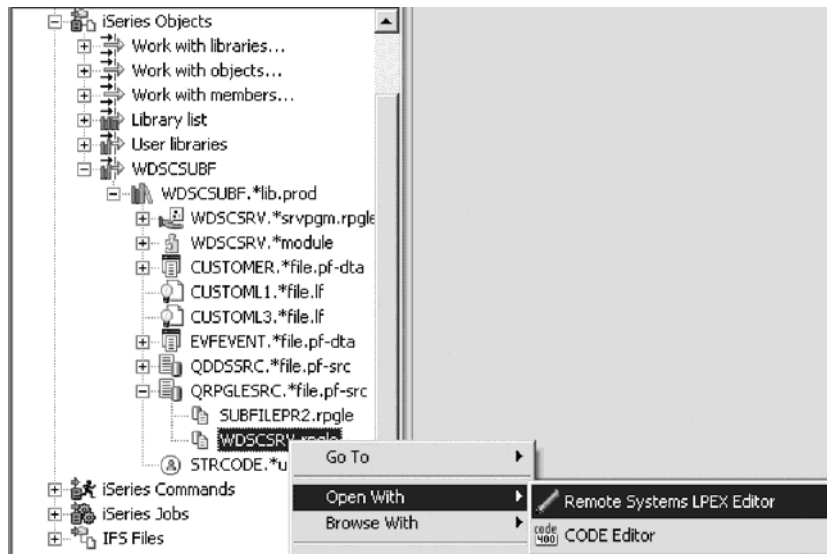


2. In the **Library** field, type WDSCSUBF.
3. Click **Next**.



4. In the **Filter name** field, type WDSCSUBF.
5. Click **Finish** to create the library filter WDSCSUBF.
Next you edit the source file.

- Expand WDSCSUBF.*lib.prod.



- Expand QRPGLESRC.*file.pf-src.
- Select WDSCSRV.rpgle.
- Right-click WDSCSRV.rpgle.
- Click **Open with > Remote Systems LPEX Editor** on the pop-up menu.

Tip: You can also double-click on a member to open the Remote Systems LPEX editor.

The editor window opens. You are ready to write your program.
- Add the following source changes:

```

14900      * Update the customer record
15000      DdoUpdate          PR              ExtProc('doUpdate')
15100      D info              LikeDS(CustInfo)
15200      D forward          20A
15300      *
15301      * Return customer number
15302      Dreadc              PR              ExtProc('readc')
15303      D info              Like(Custno)
15305      *
15306      *
15307      *
15400      /Copy WDSCSUBF/QRPGLESRC,SUBFILEPR2
15500      *-----
15600      * BldSfl

```

- Now, add these source changes at the bottom of the source file.

```

*
*-----
* readc
*
* Return customer number
*-----
*
*
*readc          B          export
d readc        pi
d info          like(CUSTNO)
*
C              Eval      rc=readcSF(CustList:
C                                  %Addr(CustRec):
C                                  %Size(CustRec):
C                                  1)
C              If        rc > 0
C              Eval      info = SflID
C              Endif
C
*
*readc          E
*
**** End of Source

```

- Save the file (Ctrl + S).

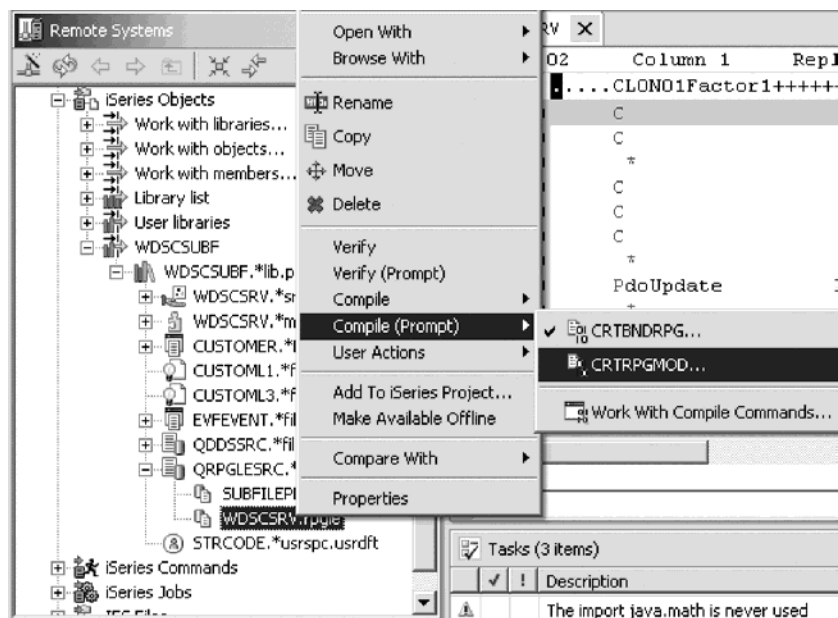
Exercise 3.3: Compiling source changes

You need to compile your source file in order for the changes to take effect.

There are two steps to compile: create a module and then create the service program.

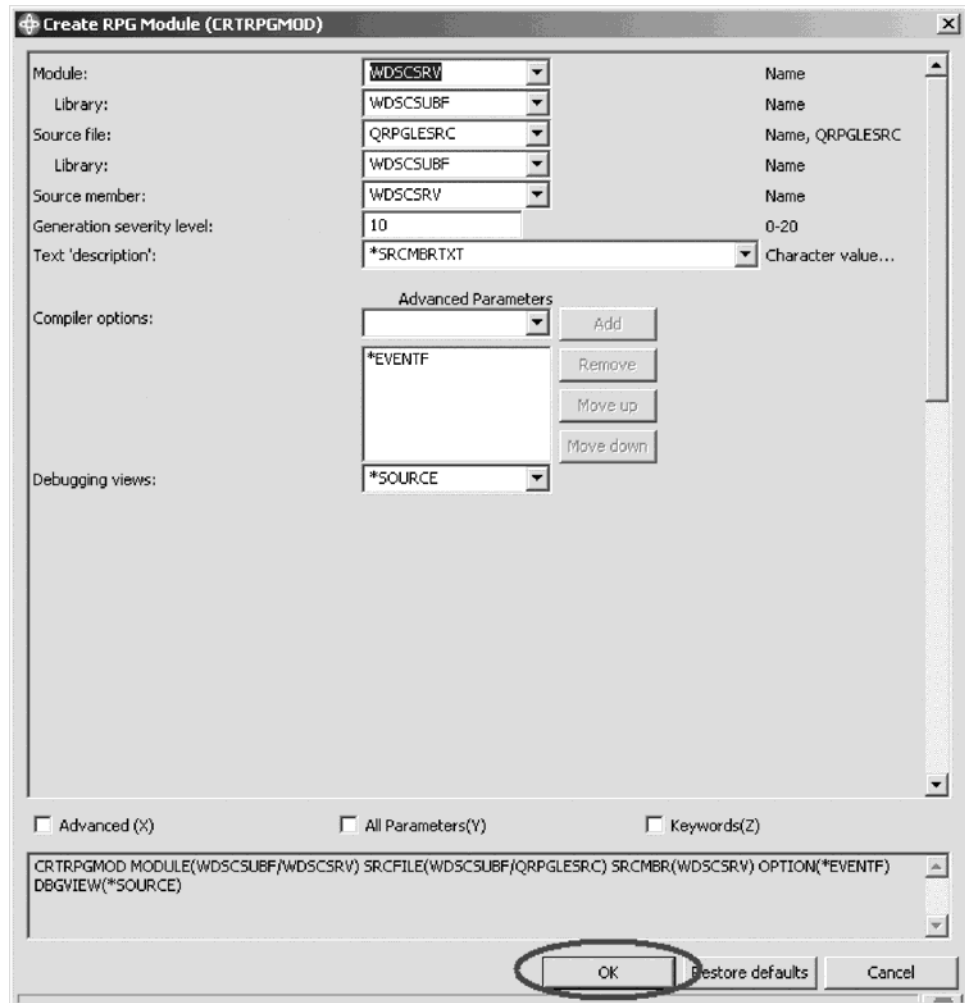
To create the RPG module:

- In the Remote Systems view, select **WDSCSRV.rpgle**.

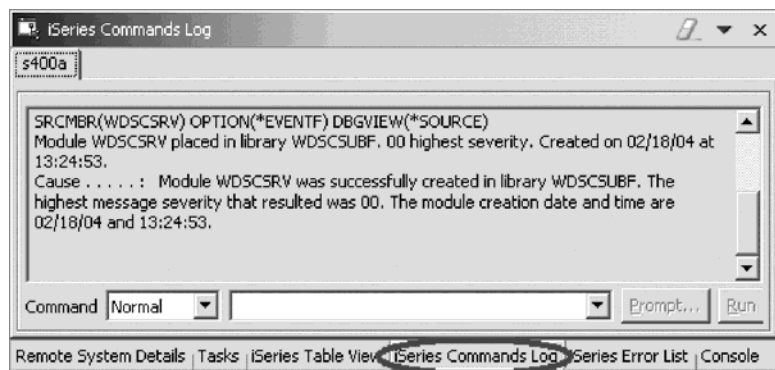


- Right-click and click **Compile (Prompt) > CRTRPGMOD** in the pop-up menu.

The Create RPG Module (CRTRPGMOD) dialog opens:

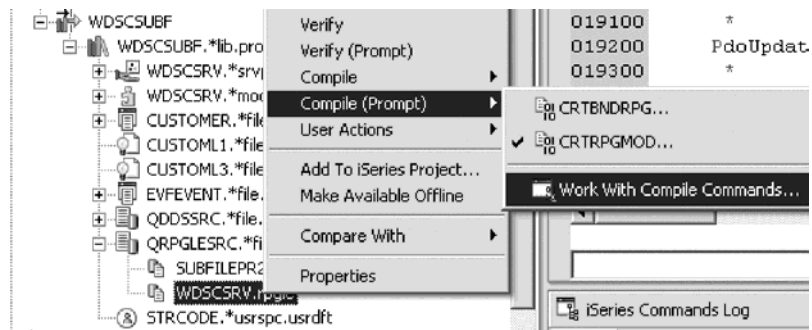


3. Click OK.



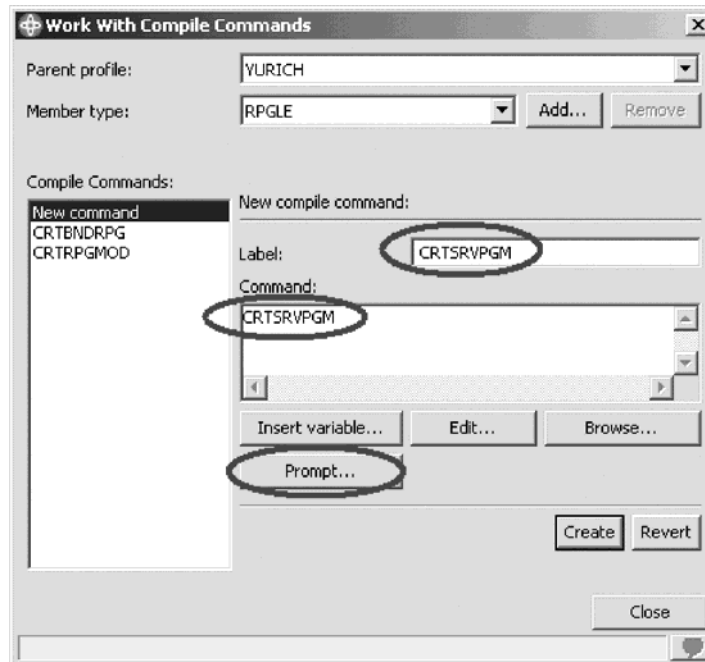
4. Check the iSeries Command Log and see if the module was created. You have now completed the first step of the compilation process. Now you complete the second step in compiling the source file.

To complete the second step:



1. Right-click WDCSRV.rpgle.
2. Click **Compile (Prompt) > Work with Compile Commands** in the pop-up menu.

The Work with Compile Commands dialog opens:



3. In the **Label** field, type CRTSRVPGM.
4. In the **Command** field, type CRTSRVPGM.
5. Click **Prompt**.

The Create Service Program (CRTSRVPGM) dialog opens:

Service program: &n
Library: &l
Module: *SRVPGM
Library: *LIBL

Export: *ALL
Export source file: QSRVSRC
Library: *LIBL
Export source member: *SRVPGM
Text 'description': *BLANK

Bind service program: qdtssf1
Library: qsys

Advanced Parameters

Binding directory: *NONE
Library: *LIBL

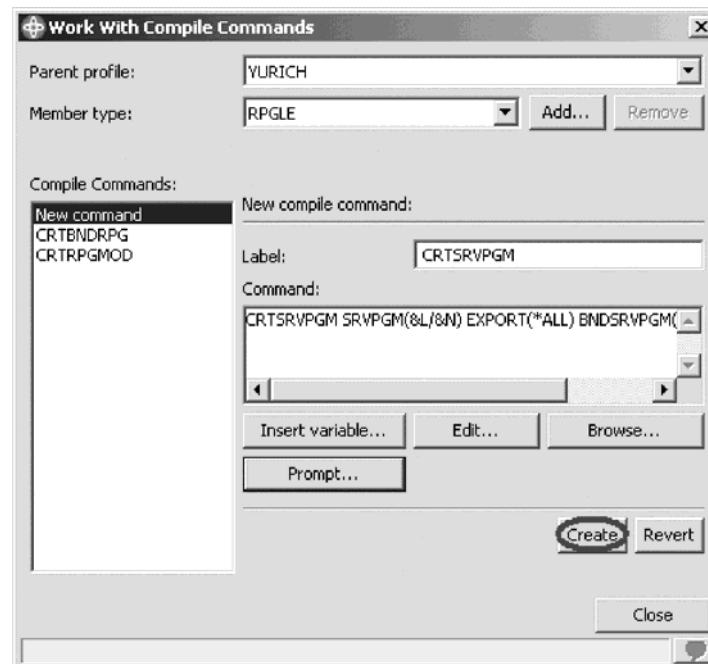
Advanced (L) All Parameters Keywords

CRTSRVPGM SRVPGM(&l/&n) EXPORT(*ALL) BND SRVPGM(QSYS/QDTSSF1)

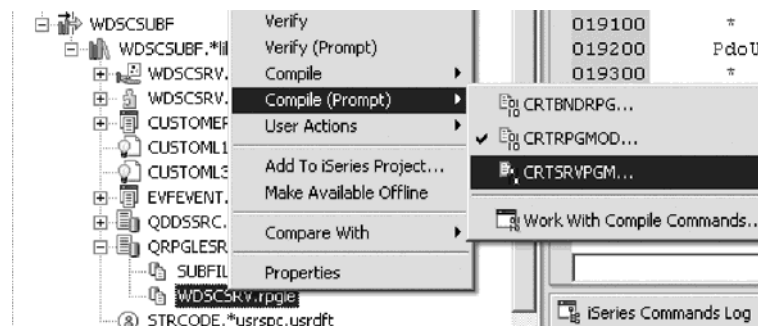
OK Restore defaults Cancel

6. In the **Service Program** field, type &n.
7. In the first (top) **Library** field, type &l.
8. In the Export list, select ***ALL**.
9. If you cannot see the Bind service program, select the **Advanced** check box.
10. In the **Bind service program** field, type qdtssf1.
11. In the **Library** field, type qsys.
12. Click **Add**.
13. Click **OK**.

The Work with Compile Commands displays:



14. Click **Create** to add the new compile command.
15. Click **Close**.
16. Right-click `WDSCSRV.rpgle`.
17. Click **Compile (Prompt)** > **CRTSRVPGM** on the pop-up menu, then click **OK**.



This completes the compilation step. The service program is created. Now you are ready to test the modifications.

Recap

Congratulations! You have completed Module 3: Adding an iSeries program to your Web application. You should now understand how to:

- Open the Remote System Explorer perspective
- Edit an iSeries source member
- Compile source changes

Now that you have used the Remote System Explorer perspective to edit and compile an iSeries program, you can continue to Module 4: Creating the update record output page.

Chapter 4. Module 4: Creating the update record output page

In this module, you will learn how to create a new output page for your existing Web application. This page will update customer information and will be generated from the Web interaction wizard.

In order to create the customer update output page, there are several steps you will need to learn about and follow:

- Generating a new update record output page using the Web interaction wizard
- Creating an error page to handle errors during the update record process
- Linking the customer details page to the update record page

In order to accomplish these learning objectives, there are several steps that are involved, including:

Exercise 4.1: Generating the output page from the Web Interaction wizard

Exercise 4.2: Linking to the update record page

Exercise 4.3: Creating the update error page

The exercises in this module must be completed in order. Start with Exercise 4.1 when you are ready to begin.

Length of time:

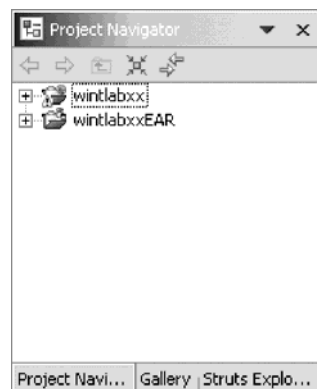
This module will take approximately 15 minutes to complete.

Exercise 4.1: Generating the output page from the Web Interaction wizard

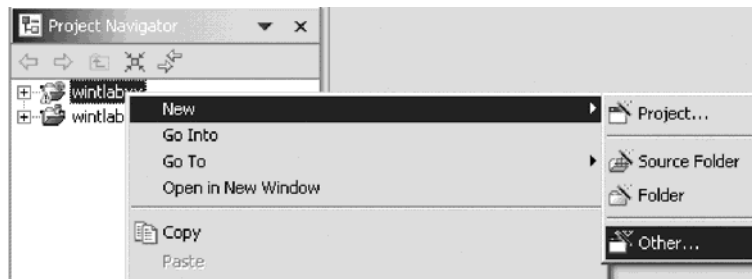
You will create the update record page using the Web interaction wizard.

To create the update record page:

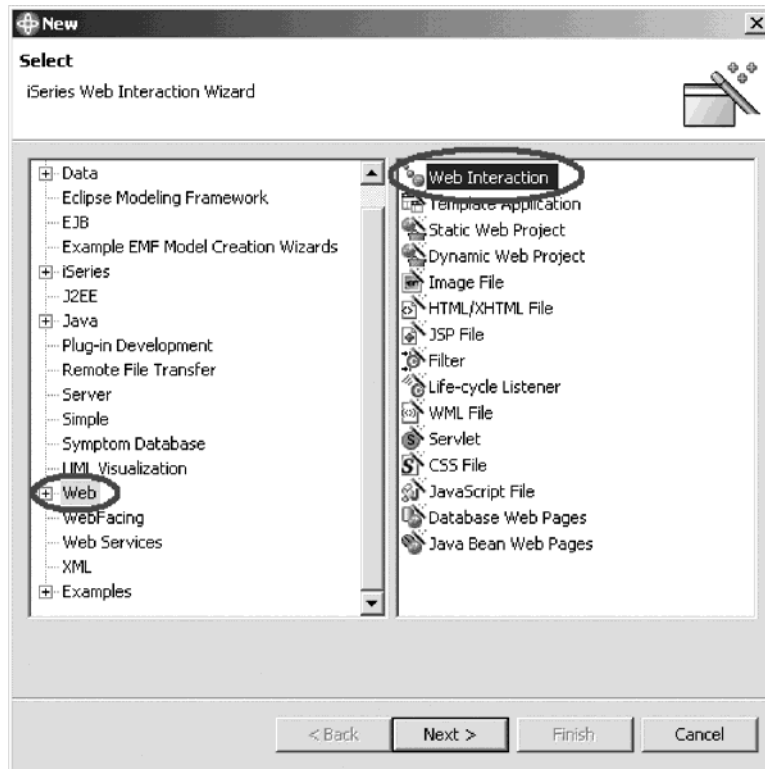
1. Open the Web perspective.
2. Locate your Web project in the Project Navigator.



3. Right-click your project and click **New > Other** on the pop-up menu.



The Select page opens:



4. In the left pane of the Select page select **Web**.
5. In the right pane of the Select page, select **Web Interaction**.
6. Click **Next**.

The Web Interaction wizard opens:

Web Interaction

Specify a Name and Location for your Web Interaction

Define the name and location for the communication between Web pages and the chosen business process. The location must be the Web Content directory inside a project.

Destination folder: /winlaboc/WebContent [Browse...]

Web Interaction name: custDispUpdate

Java package: [Browse...]

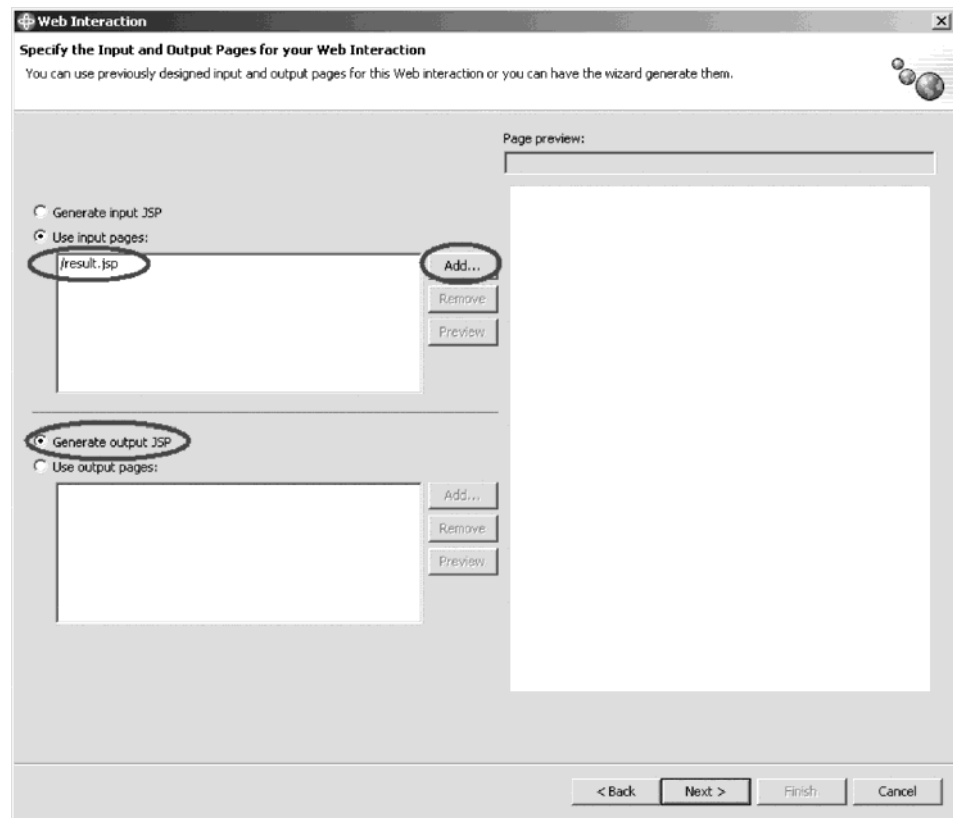
Use error page: /error.jsp [Browse...]

Invalidate session after the interaction occurs

< Back Next > Finish Cancel

7. In the **Web Interaction name**, type custDispUpdate.
8. Select the **Use error page** check box and select **error.jsp** on the Error page dialog. Click **OK**.
9. Click **Next**.

This next page is where you specify the input and output pages.



10. Leave the **Use input pages** radio button selected and click **Add**.
11. Select **result.jsp** on the Input dialog and click **OK**.

This is the page our original Web Interaction generated, we are now linking it with another Web Interaction.
12. Click the **Generate output JSP** radio button to tell the Web Interaction wizard to make an output page.
13. Click **Next**.

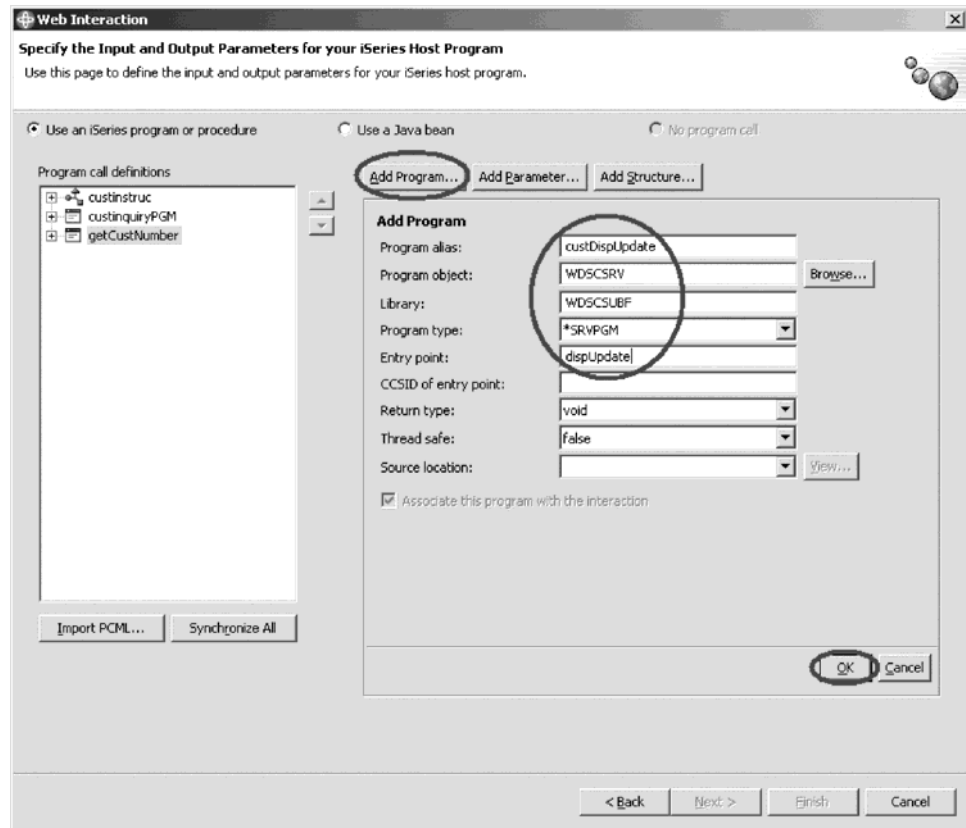
On the next page, you will need to add another program instance and define the input and output parameters for this Web Interaction.

You will notice that the structure and the program from your previous Web Interaction are visible, and available for your use.

To add a program:

1. Click **Add Program**.

2. Fill in the program details as shown:



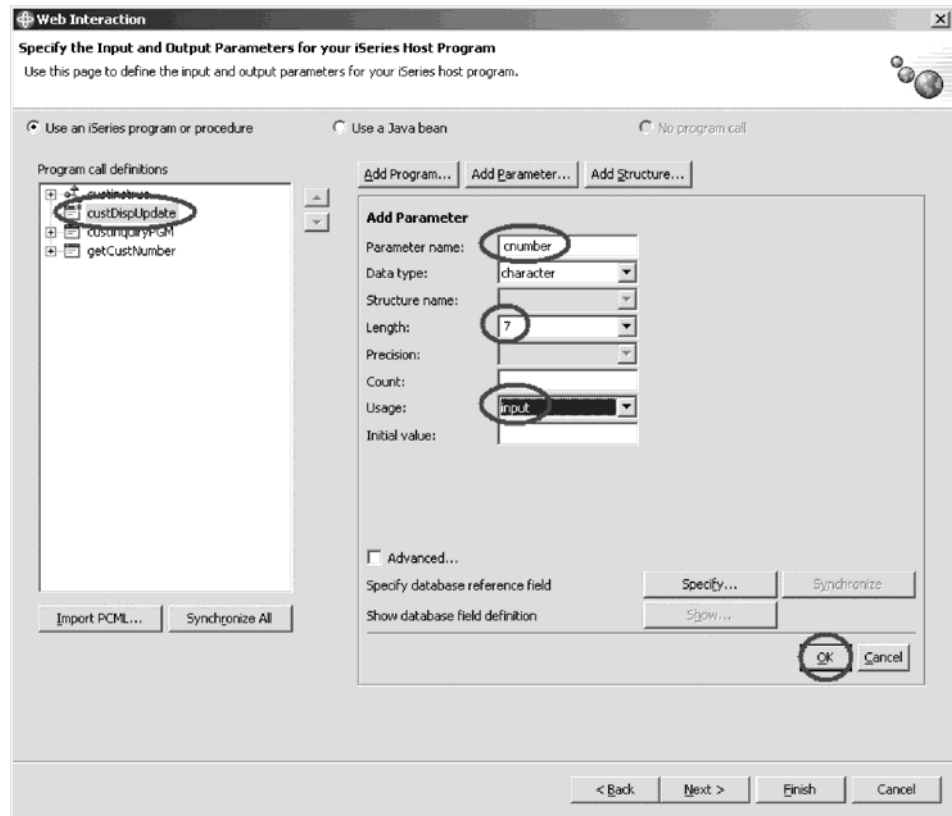
3. Click **OK** when finished.

You will see that the program has been added under the **Program call definitions**.

To add parameters:

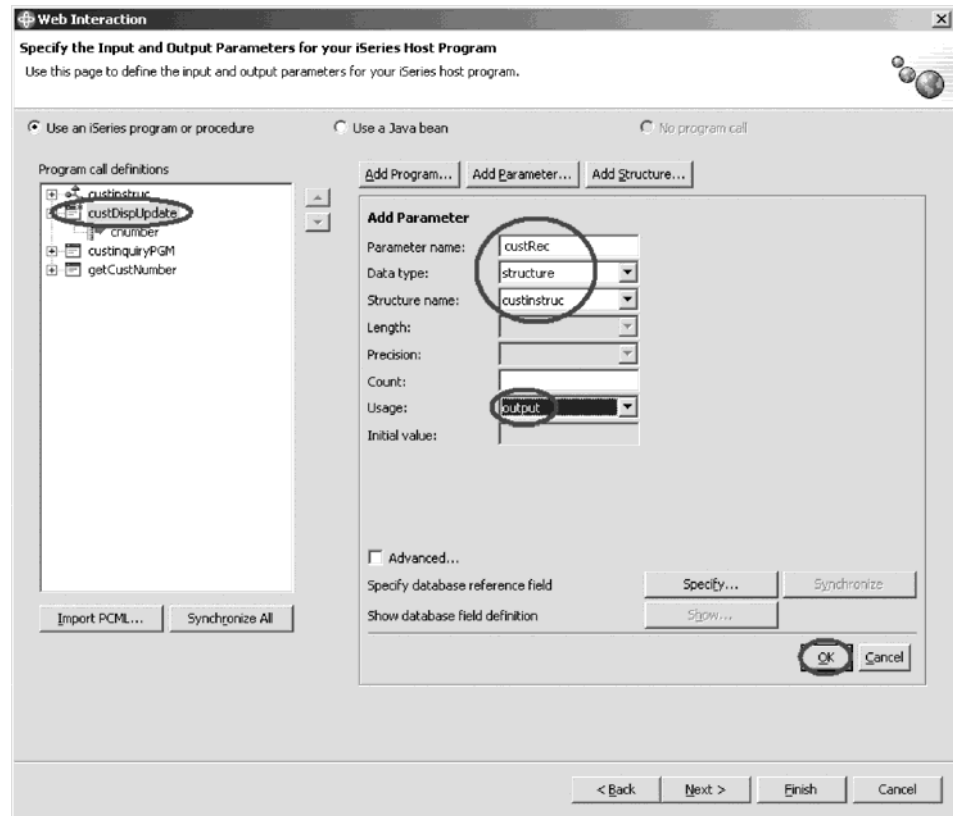
1. Click **Add Parameter**.

2. Fill in the parameter details as shown:



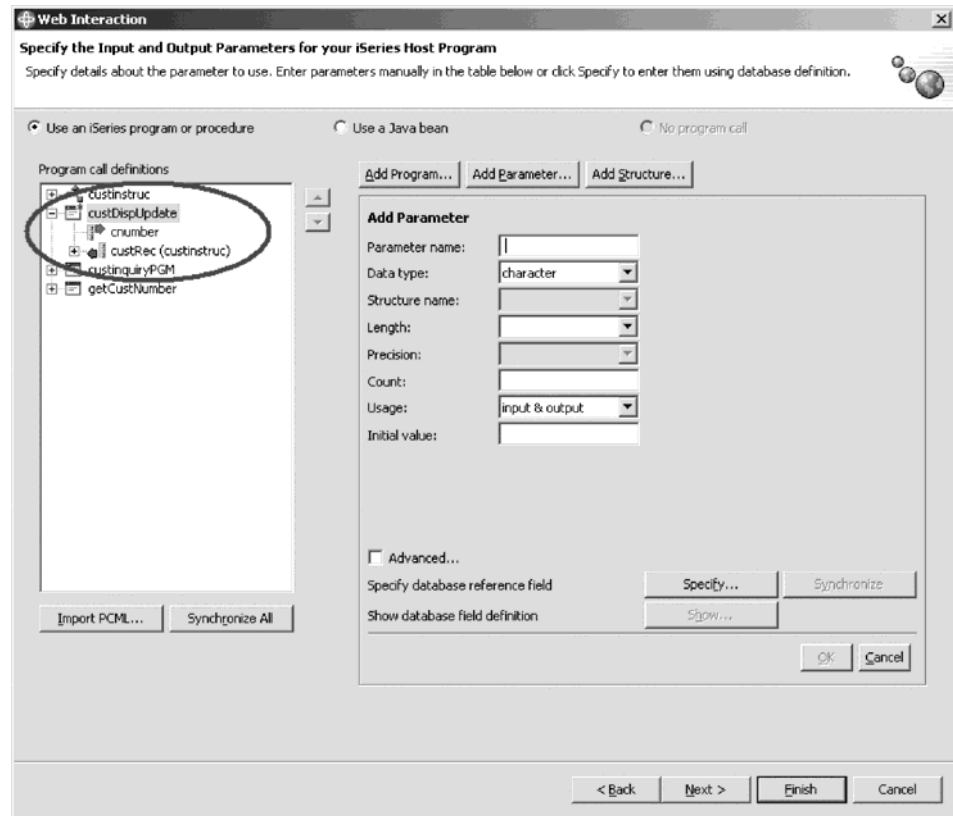
3. Click **OK** when finished.
Now we need to add the structure you defined in Exercise 2.6.
4. Click **Add Parameter**.

5. Fill in the details as shown:



6. Click OK.

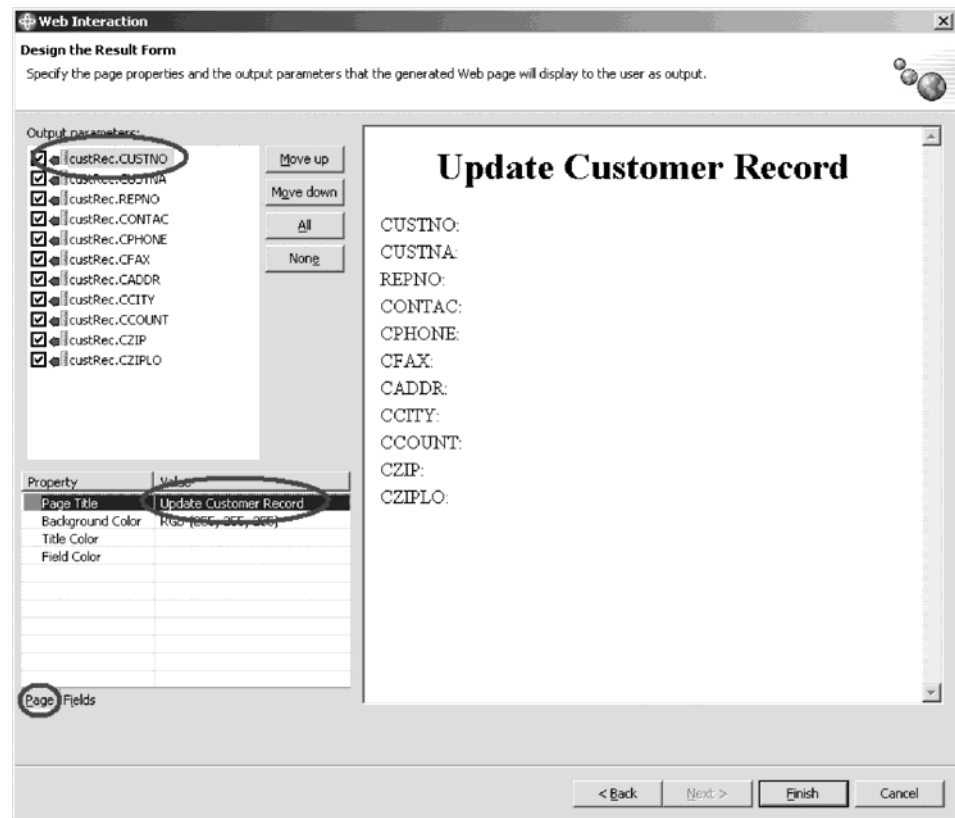
Your wizard page should now look like this:



You can see the new parameters you have defined for this interaction.

7. Click Next.

Now its time to design the output page:

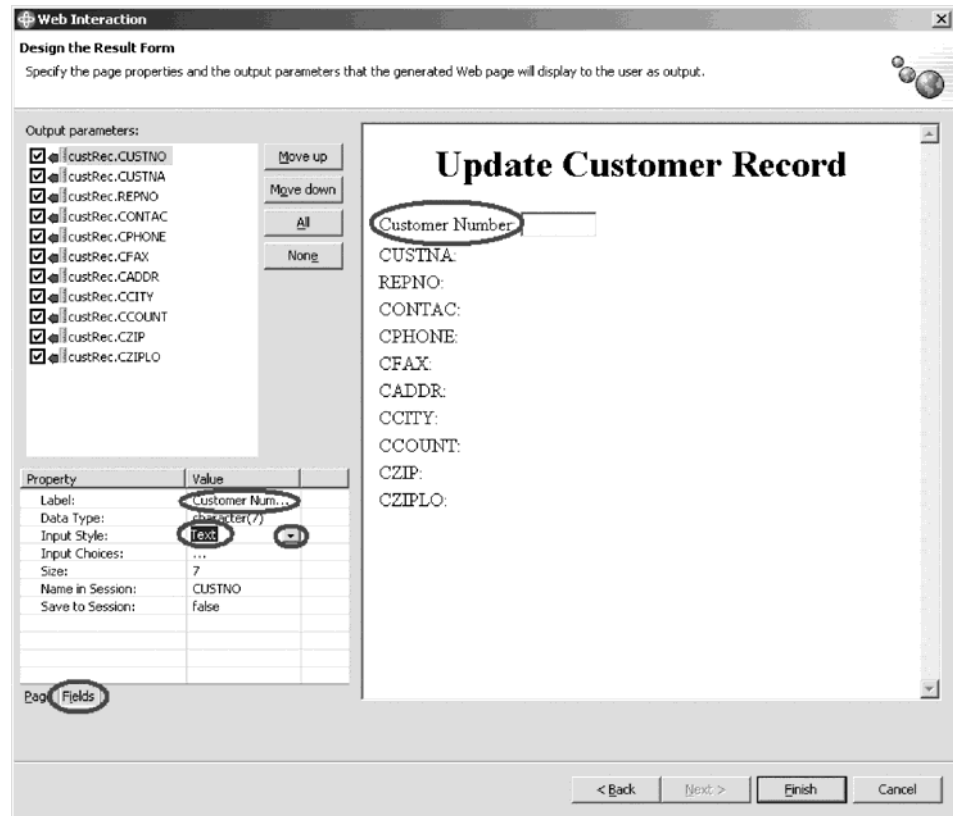


First you will change the output page title.

To design the output page:

1. In the Properties table, select the **Page** tab.
2. Change the value of the Page Title property to Update Customer Record.
Now you need to change the field labels to make sense and to make them editable.

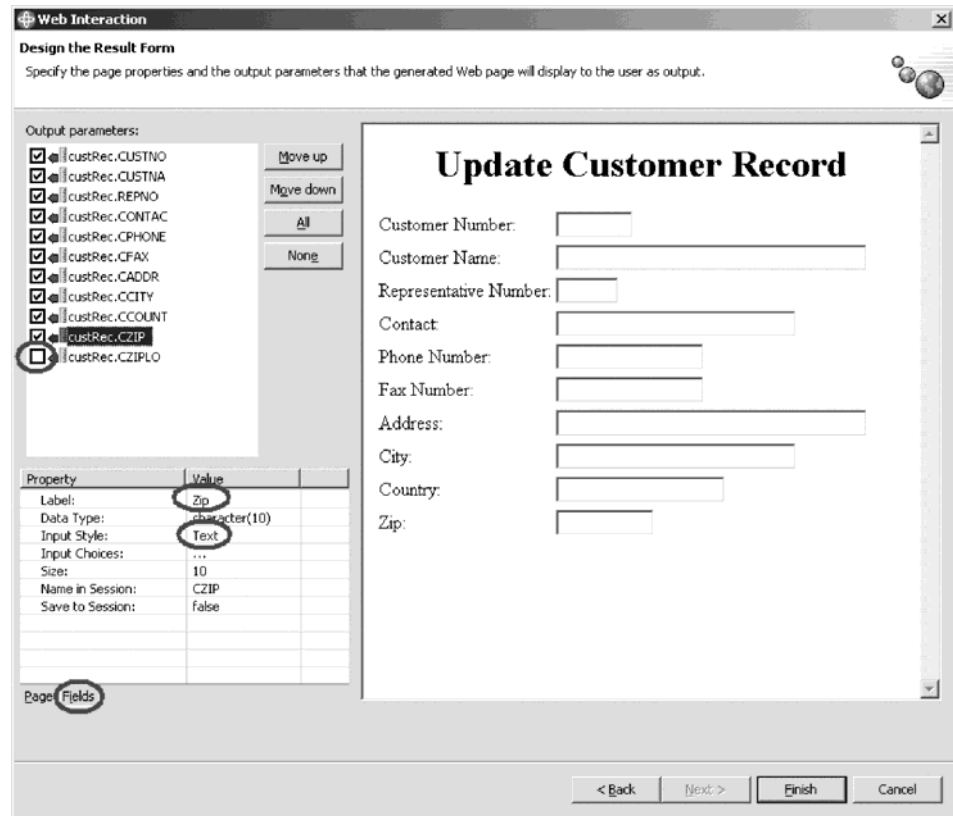
3. Click the **Fields** tab.



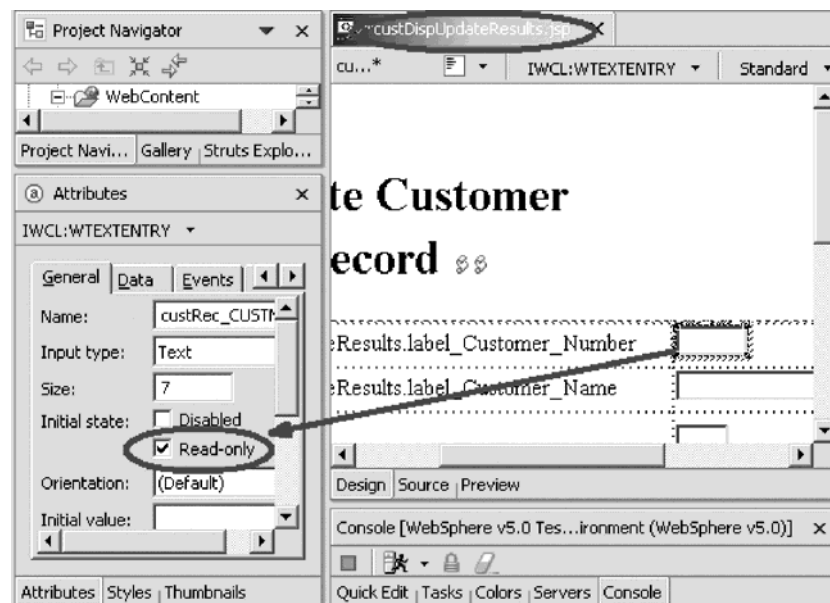
4. Under **Output parameters**, select custRec.CUSTNO.
5. In the Properties table, change the value of the Label property to Customer Number.
6. Select **Text** from the **Input Style** list.
7. Repeat for the following by changing their Label fields to their respective names in the table below, and by changing all their Input Style fields to Text.
8. Make sure that their Size and Max. Length fields are the same.

CUSTNA	Customer Name
REPNO	Representative Number
CONTAC	Contact
CPHONE	Phone Number
CFAX	Fax Number
CADDR	Address
CCITY	City
CCOUNT	Country
CZIP	Zip

Your page should now look like this:



9. You have chosen not to use the **CZIPLO** field. Clear this check box.
10. Click **Finish**.
11. Navigate to your new page `custDispUpdateResults.jsp` and double-click the file.
The Page Designer opens.
12. Select the text field for the Customer Number.



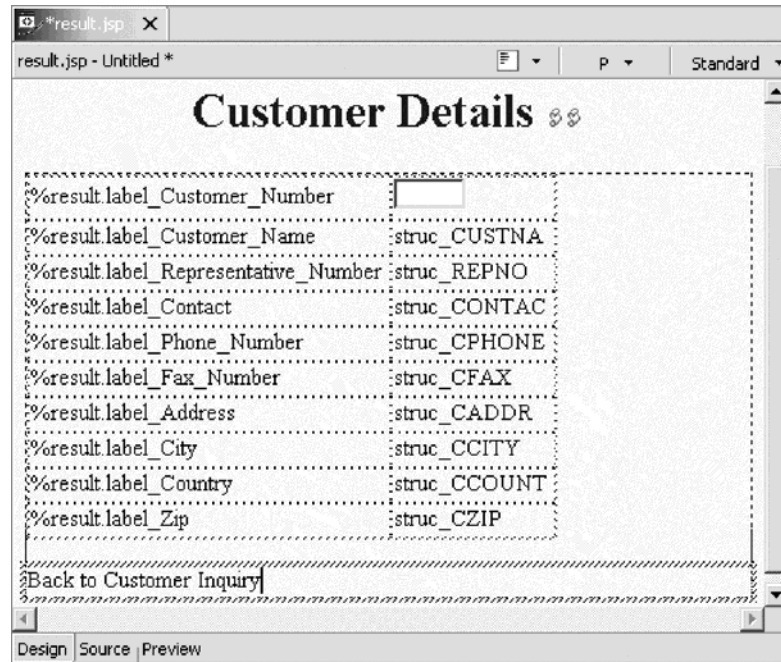
13. In the Attributes pane, set the field to **Read-only**. Select the **Read-only** check box.
You don't want to allow the user to change the customer number.
14. **Save** and close the JSP.

Exercise 4.2: Linking to the update record page

Now let's modify the customer details page so that you can link to the update record page.

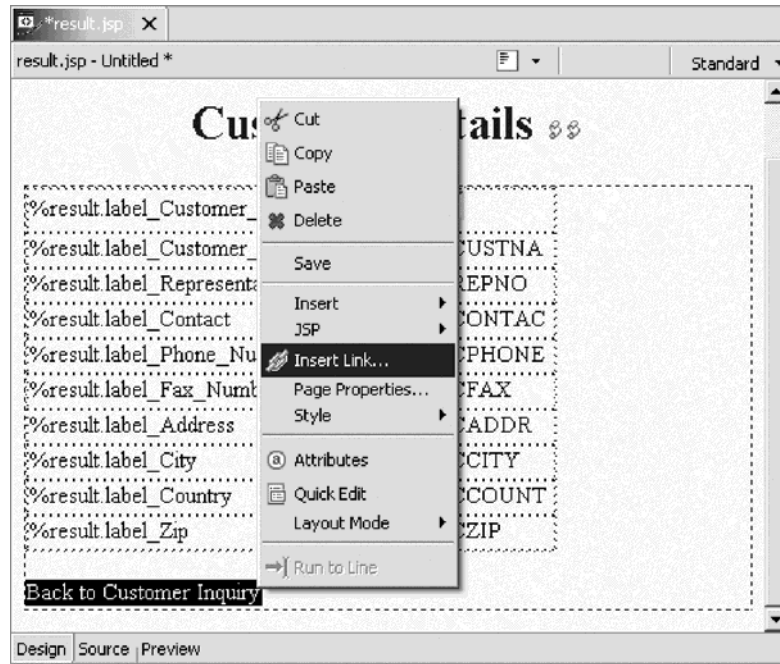
To link to the update record page:

1. In the Project Navigator, drill down to your result.jsp page and double-click.



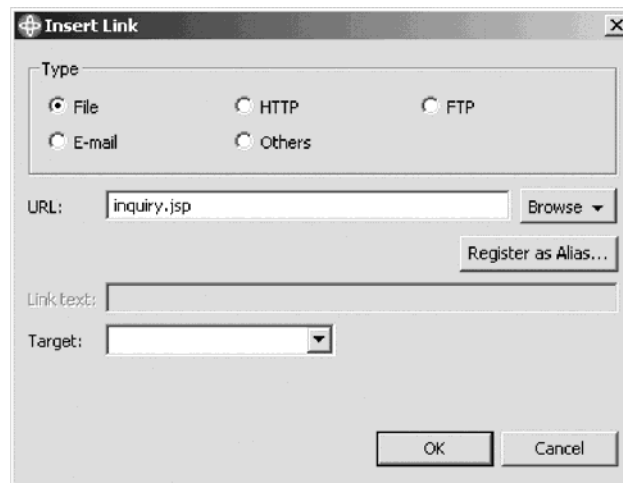
2. Select the **Submit** button and press the **Delete** button.
3. Select the **Reset** button and press the **Delete** button.
4. Type `Back to Customer Inquiry`.

5. Highlight the text **Back to Customer Inquiry**, right-click and click **Insert Link** on the pop-up menu.



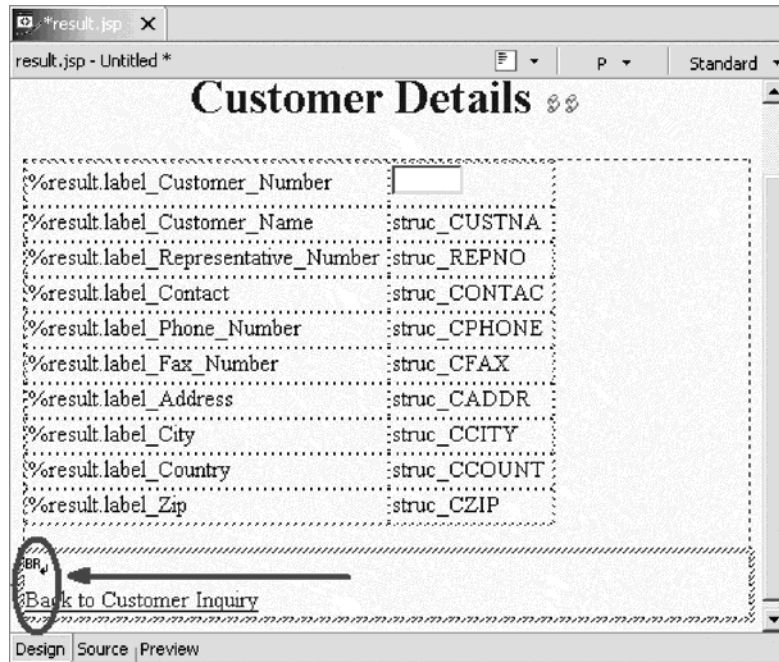
The Insert Link dialog opens.

6. In the **URL** field, type `inquiry.jsp`.

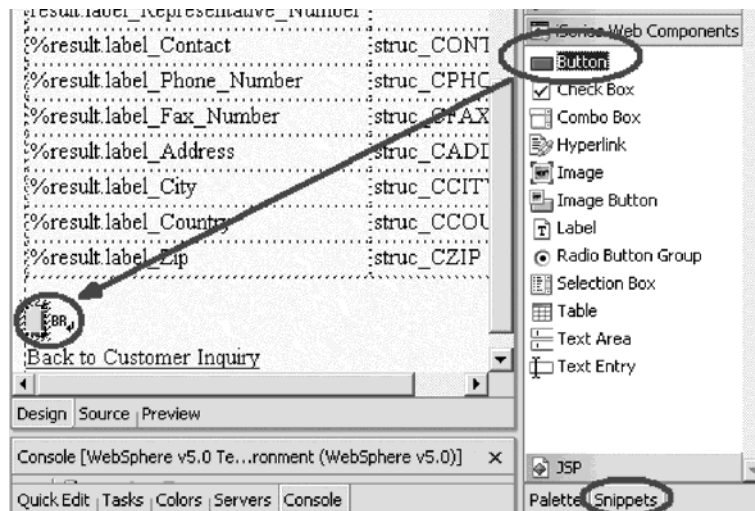


7. Click **OK**.

- Put your cursor right before the **Back to Customer Inquiry** link and press **Enter** on your keyboard.

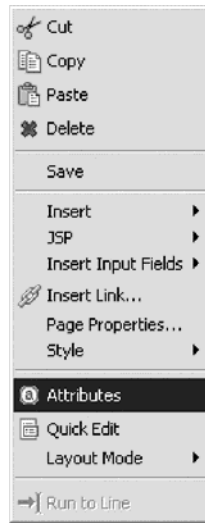


- In the right pane, click the **Snippets** tab and expand **iSeries Web Components** if not already expanded.
- Click and drag the **Button** component to the location shown.

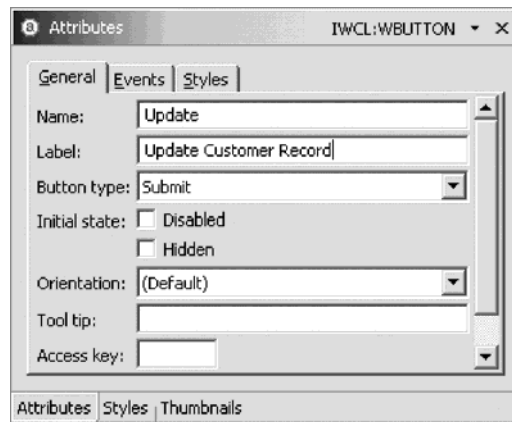


The Attributes pane in the bottom left of the workbench should automatically appear.

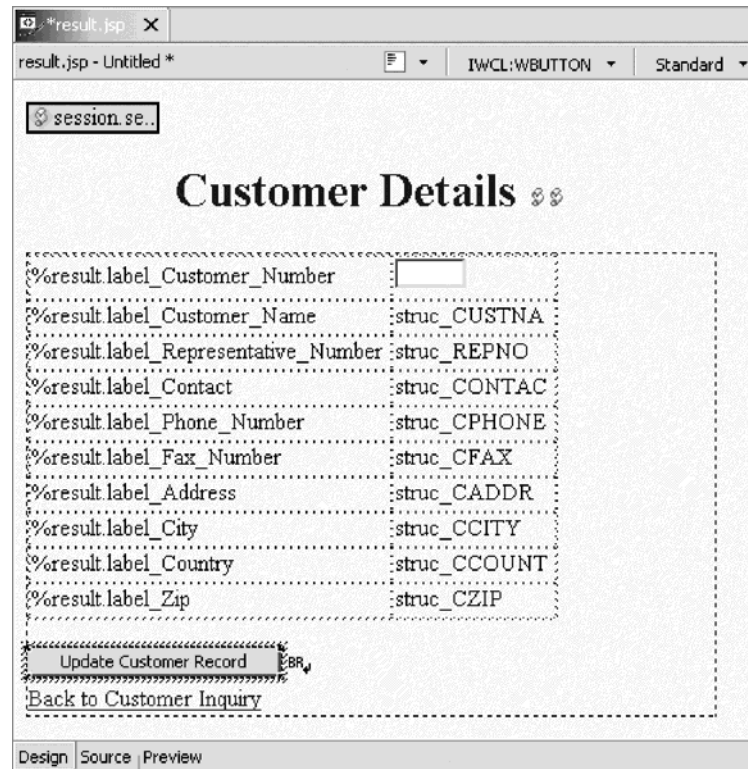
11. If not, right-click the button and click **Attributes** on the pop-up menu.



12. Fill in the Button's details as shown below:



Your Customer Details page should now look like this:



13. Save the file (Ctrl+S).

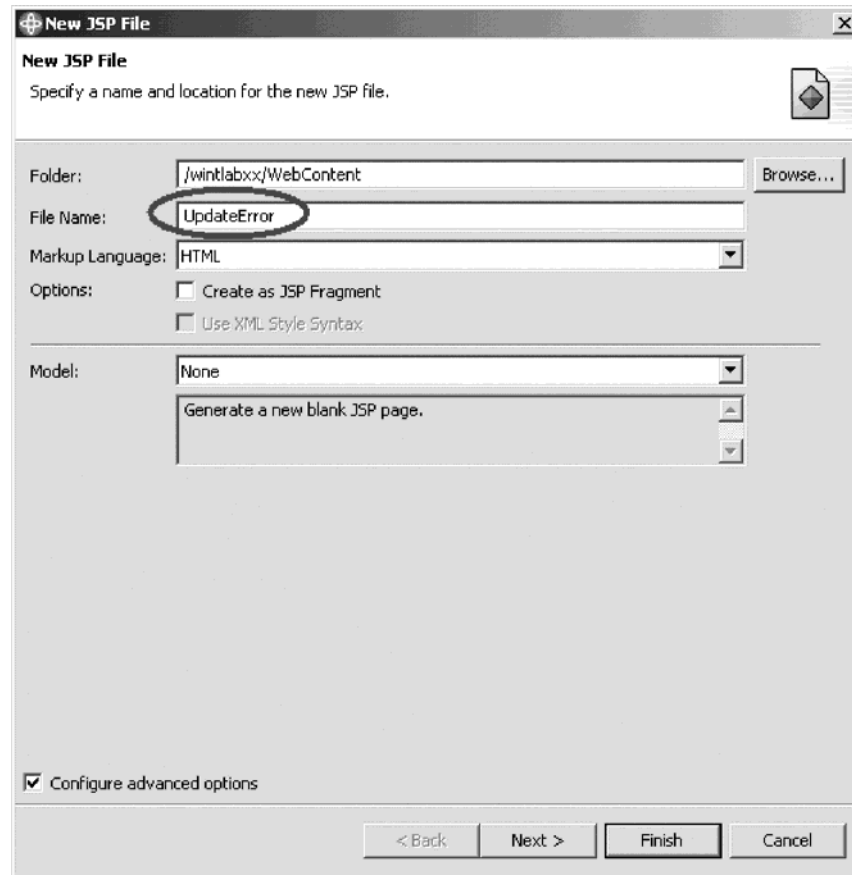
Exercise 4.3: Creating the update error page

Now let's create the error page that will be displayed if there is an error during the update process.

To create the update error page:

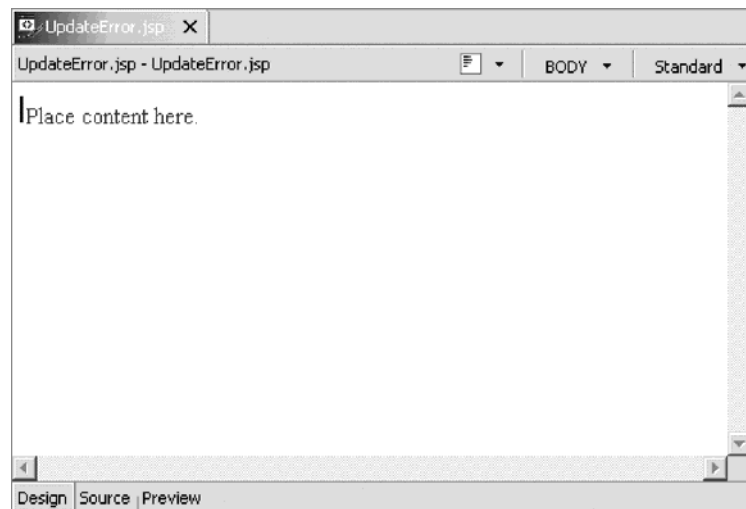
1. In the Project Navigator, expand your project if not already and drill down to the WebContent folder.
2. Right-click the WebContent folder inside your project.
3. Click **New > JSP File** on the pop-up menu.

The New JSP File dialog opens:



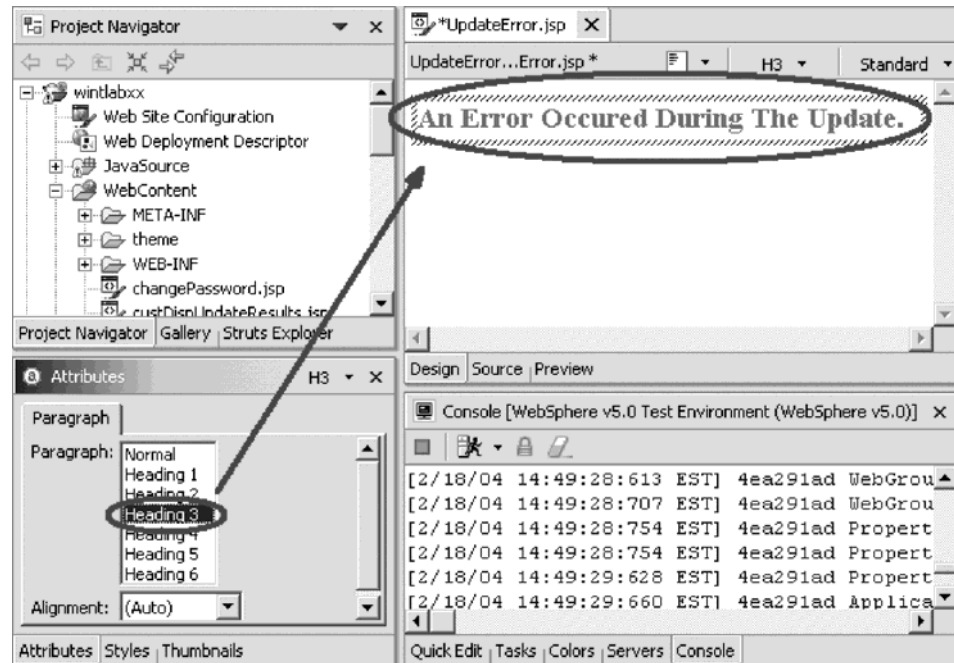
4. In the **File Name** field, type UpdateError.jsp.
5. Click **Finish**.

The Page Designer opens. Your window will look like this:

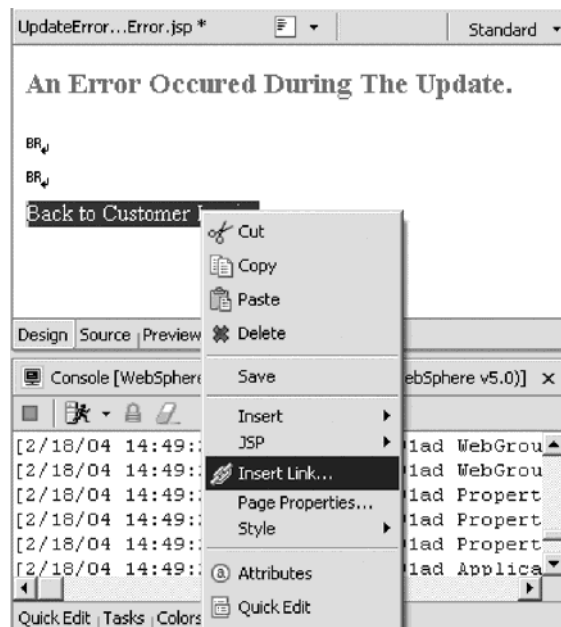


6. Click the words "Place content here" and change it to An error occurred during update.
7. Highlight your new text.

This opens the Attributes pane. It appears in the bottom right of the workbench.

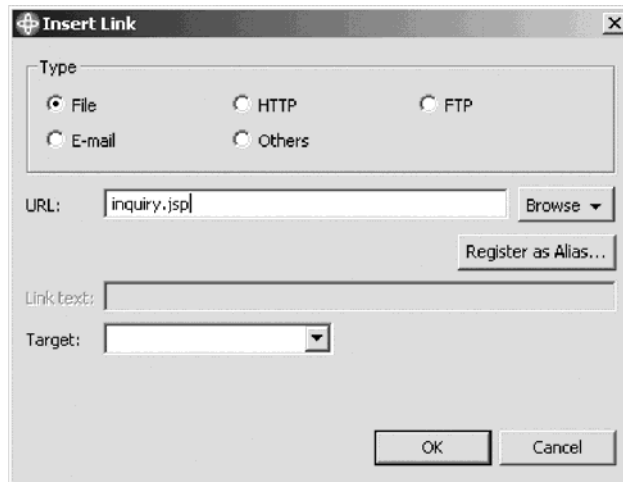


8. Select **Heading 3** from the Attributes pane.
9. Place your cursor below the heading you just created, and press the **Enter** key two times.
10. Type the text **Back to Customer Inquiry**.
11. Highlight the text and right-click.
Now you will define the link.



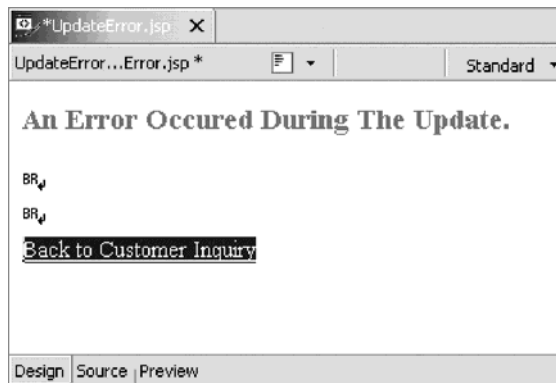
12. Click **Insert Link** on the pop-up menu.
The Insert Link dialog opens.

13. In the URL field, type inquiry.jsp.



14. Click OK.

Your **UpdateError.jsp** should now look like this:



15. Save the file.

You are now ready to use the Web Interaction Wizard to add the flow control parameter.

Recap

Congratulations! You have completed Module 4: Creating the customer update output page. You should now understand how to:

- Create another Web interaction.
- Link the new page in the Web interaction.
- Create an error page.

Now that you have created the update record output page, you can continue to Module 5: Creating the update complete output page.

Chapter 5. Module 5: Creating the update complete output page

In this module, you will learn how to create the customer update complete output page.

In order to create the customer update complete output page, there are several steps you will need to learn about and follow:

- Generating a new output page using the Web Interaction wizard
- Adding the flow control error page
- Linking the update record page to inquiry page
-

In order to accomplish these learning objectives, there are several steps that are involved, including:

Exercise 5.1: Generating another output page using the Web Interaction wizard

Exercise 5.2: Adding the flow control error page

Exercise 5.3: Linking the update complete page to the inquiry page

The exercises in this module must be completed in order. Start with Exercise 5.1 when you are ready to begin.

Length of time:

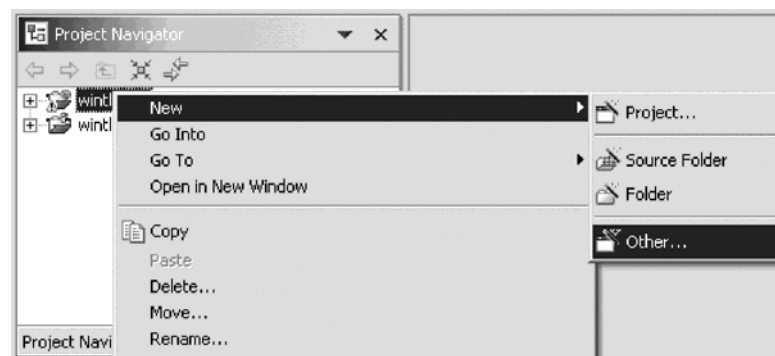
This module will take approximately 15 minutes to complete.

Exercise 5.1: Generating another output page using the Web Interaction wizard

Now you will create the Web Interaction that will actually update the record on the iSeries. It will read the input variables from the custDispUpdateResults.jsp.

To generate another output page:

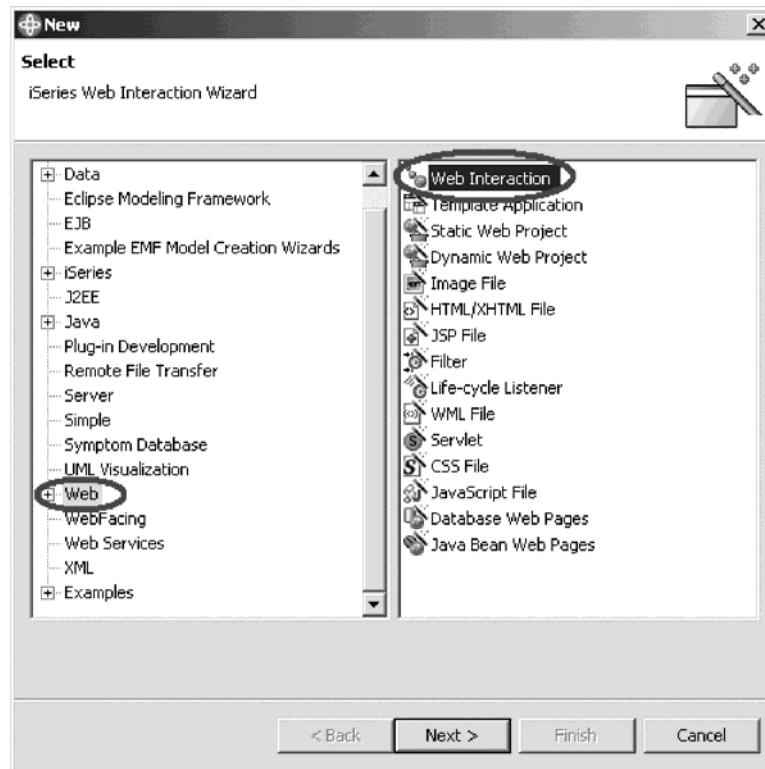
1. Right-click your project and click **New > Other** on the pop-up menu.



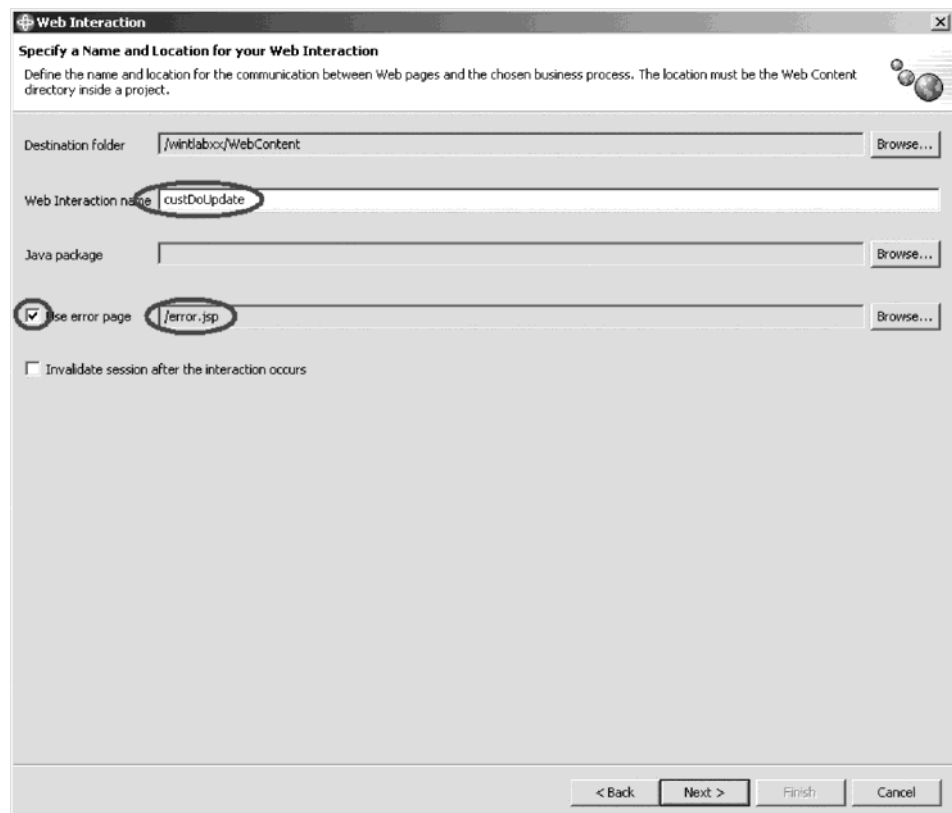
The Select page opens.

2. In the left pane, select **Web**.

3. In the right pane, select **Web Interaction**.

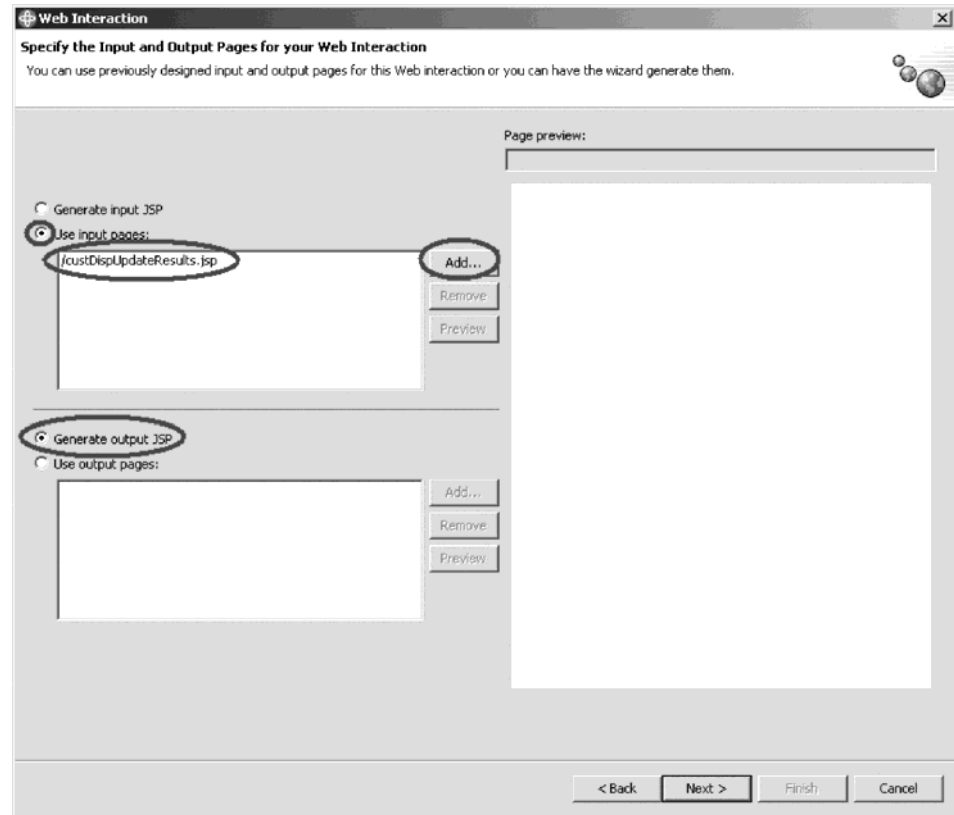


4. Click **Next**.
5. In the **Web Interaction name** field, type `custDoUpdate`.



6. Select the **Use error page** check box and select **error.jsp** on the Error page dialog. Click **OK**.
7. Click **Next**.

Now you define the input and output pages.

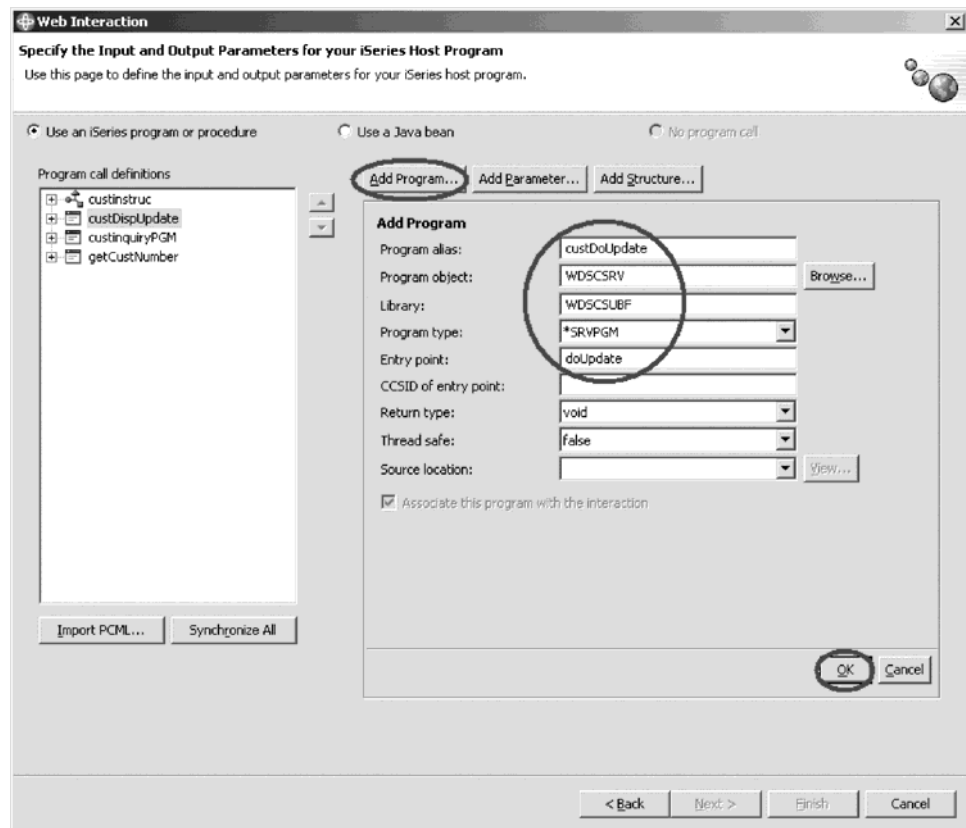


8. Leave the **Use input pages** radio button selected and click **Add**.
9. Select **custDispUpdateResults.jsp** on the Input page dialog. Click **OK**.
This is the page our original Web Interaction generated, you are now linking it with another Web Interaction.
10. Click the **Generate output JSP** radio button to tell the Web Interaction wizard to make an output page.
11. Click **Next**.
Now you will add a program.

To add a program:

1. Click **Add Program**.

2. Fill in the program details as shown.



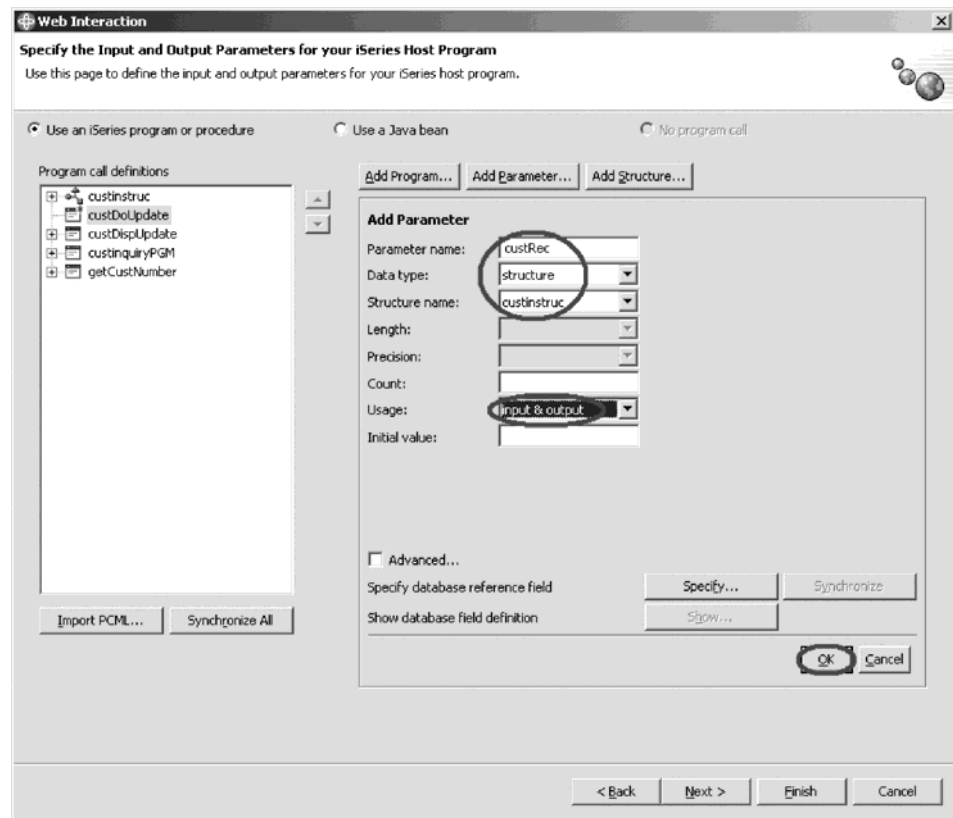
3. Click **OK** when finished.

You will see that the program has been added under **Program call definitions**.
Now you need to add the structure you defined in Exercise 2.6.

To add a parameter:

1. Click **Add Parameter**.

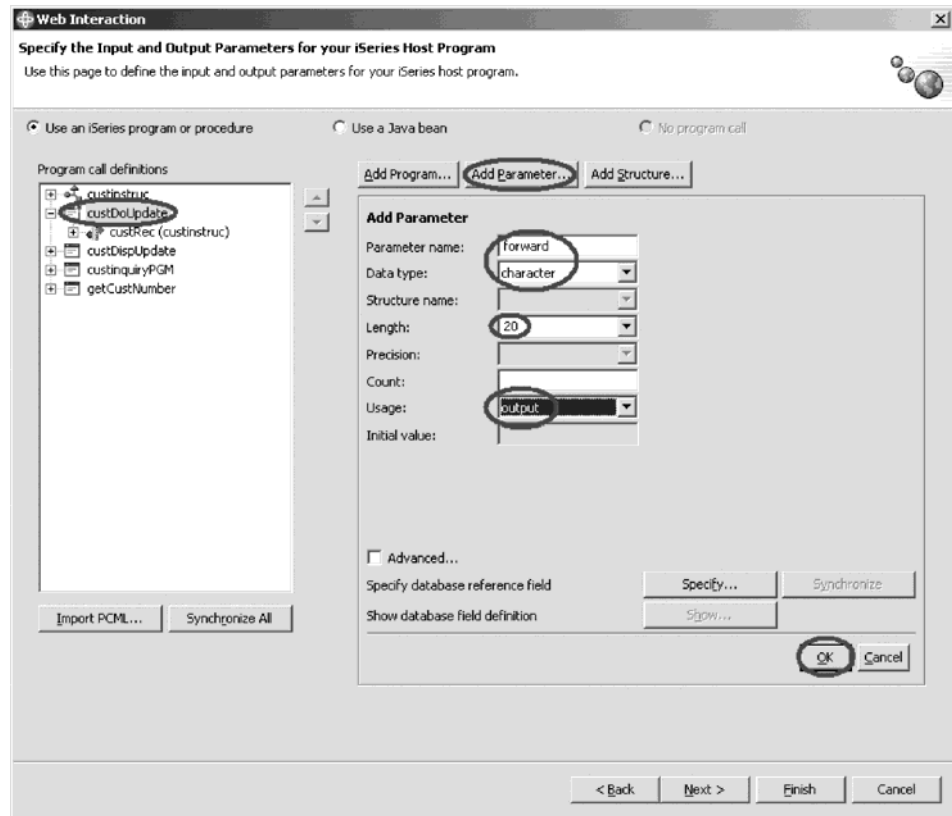
2. Fill in the details as shown:



3. Click **OK**.

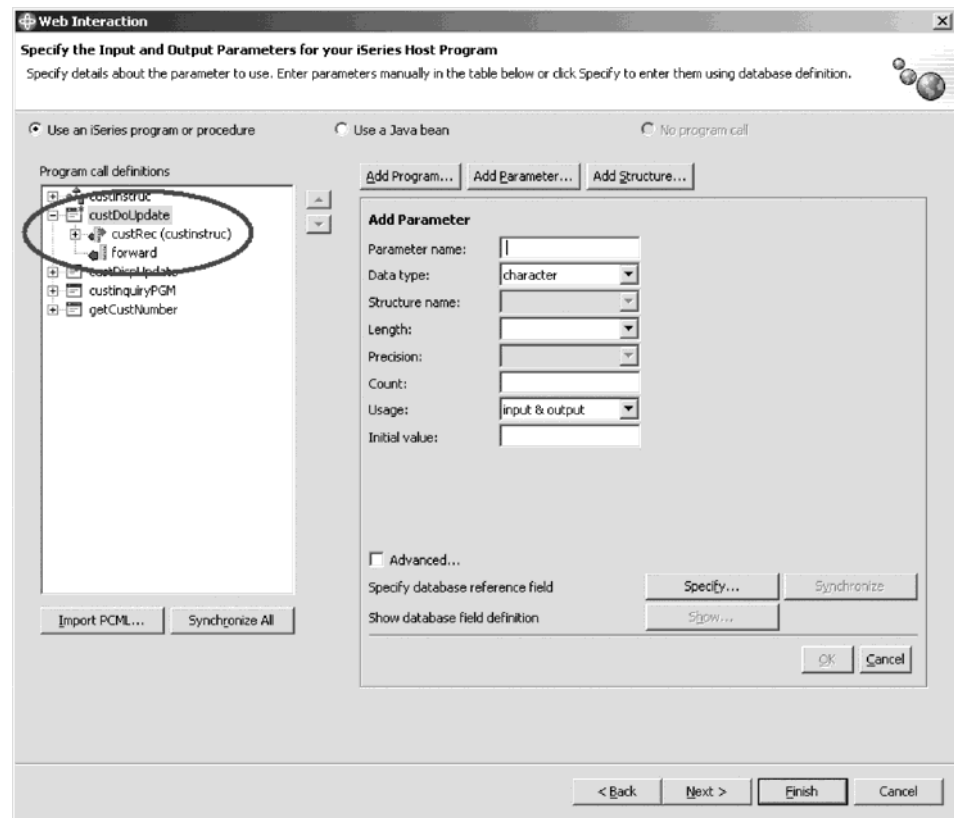
Now you will add the flow control parameter.

4. Fill in the parameter details as shown:



5. Click OK when finished.

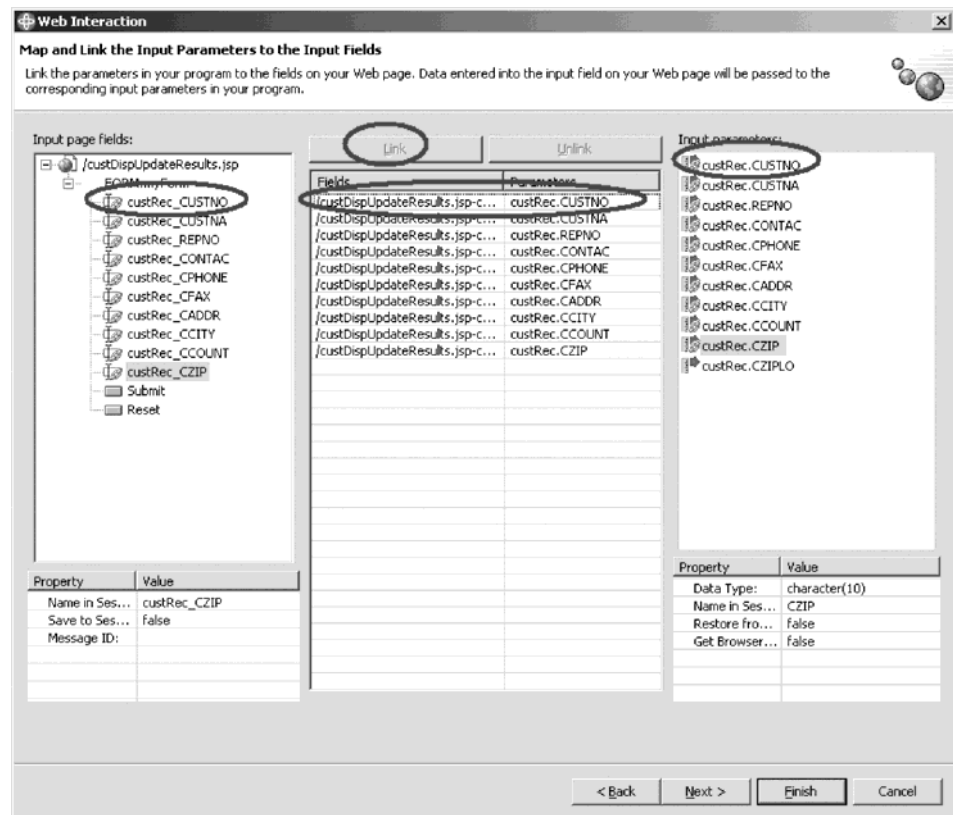
Your wizard page should now look like this:



You can see the new parameters you have defined for this interaction.

6. Click Next.

The input parameter page opens:



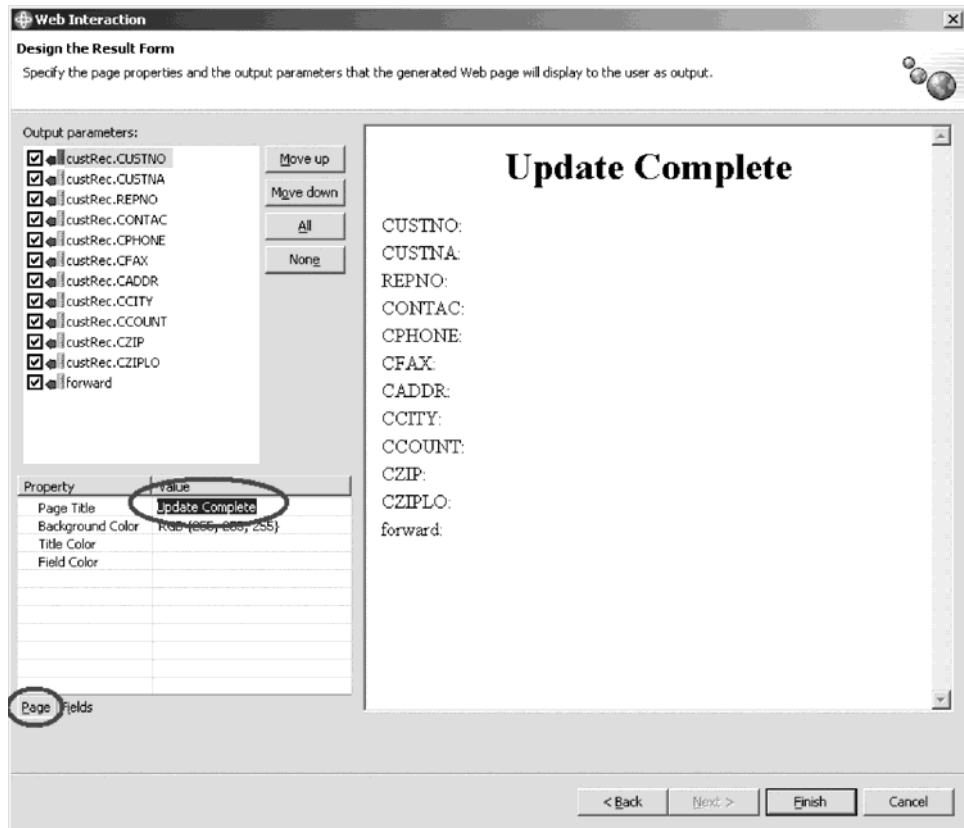
Now you will link the input page parameters.

7. Select an input page parameter under **Input page fields**.
8. Select the respective program input parameter under **Input parameters**.
9. Click the **Link** button in the middle of the page
You have just linked a parameter from the input page to the input parameter of your program.
10. Repeat for all remaining input page parameters.

Now you design the output page.

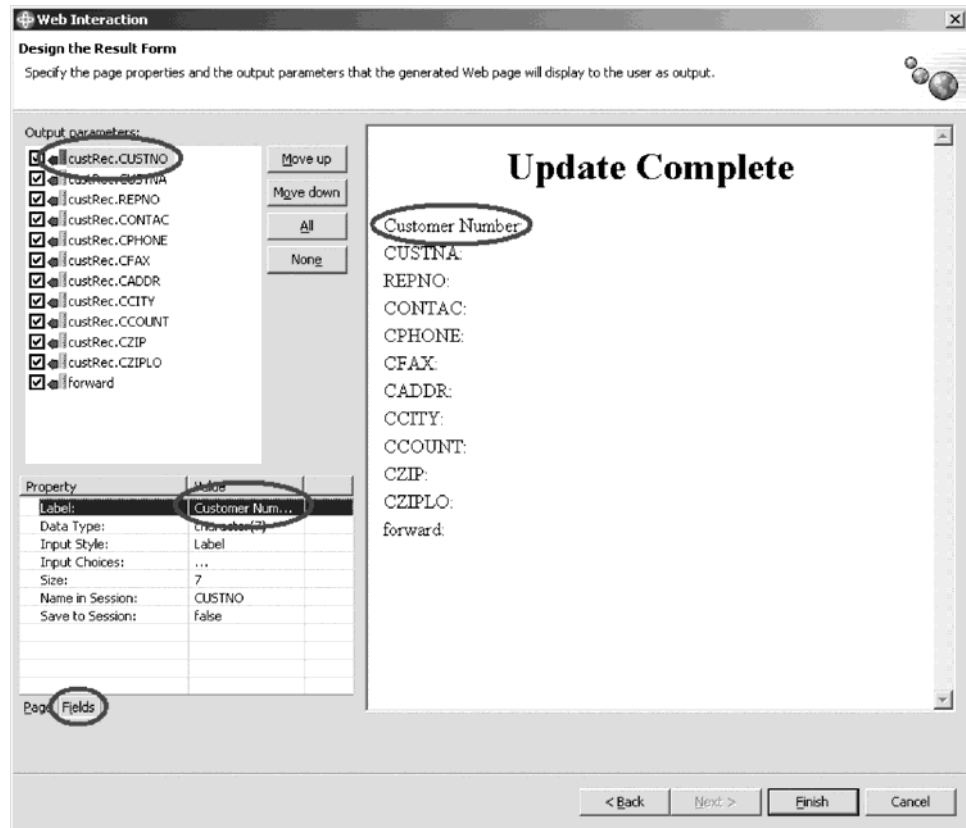
To design the output page:

1. Click Next.



2. In the Properties table, click the **Page** tab.
3. Change the value of the Page Title property to Update Complete.
Now change the field labels to be readable.

- Click the **Fields** tab.

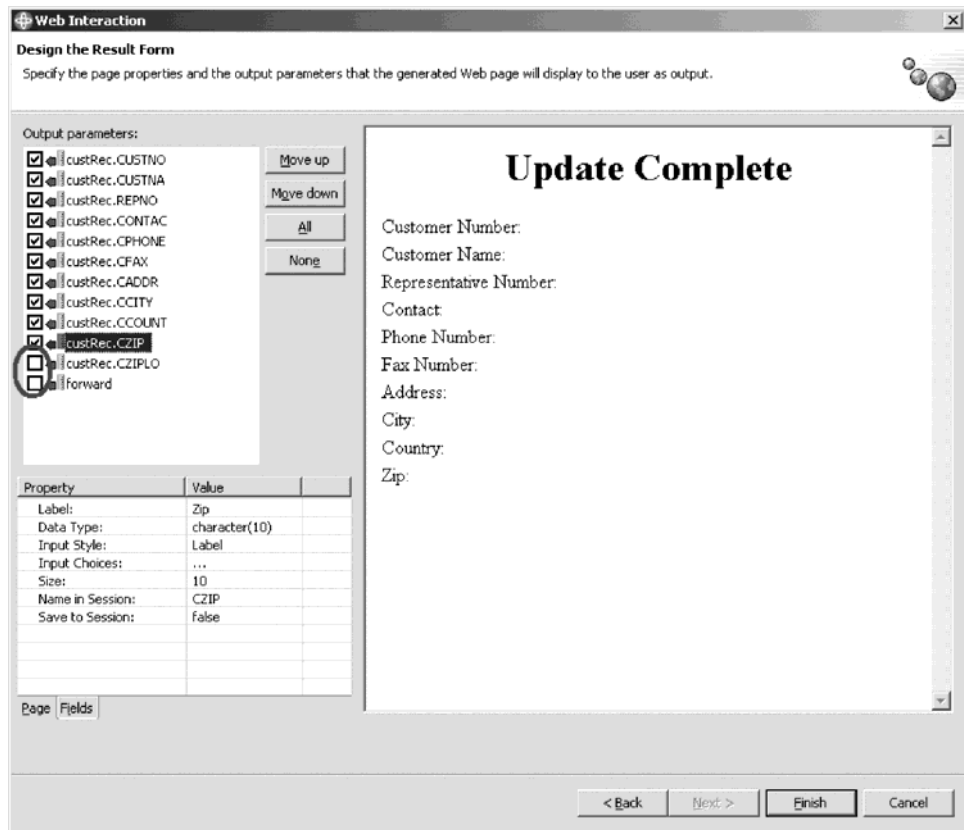


- Select **custRec.CUSTNO** from the left.
- Change the value of the Label property to Customer Number.
- Using the table below, change the values of the remaining properties:

CUSTNA	Customer Name
REPNO	Representative Number
CONTAC	Contact
CPHONE	Phone Number
CFAX	Fax Number
CADDR	Address
CCITY	City
CCOUNT	Country
CZIP	Zip

Now, let's disable the values you don't want to be displayed.

As a final step, you choose not to use the CZIPLO field and not to display the forward field.

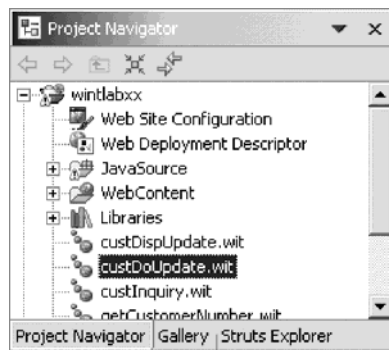


8. Clear their respective check boxes.
9. Click **Finish**.

Exercise 5.2: Adding the flow control error page

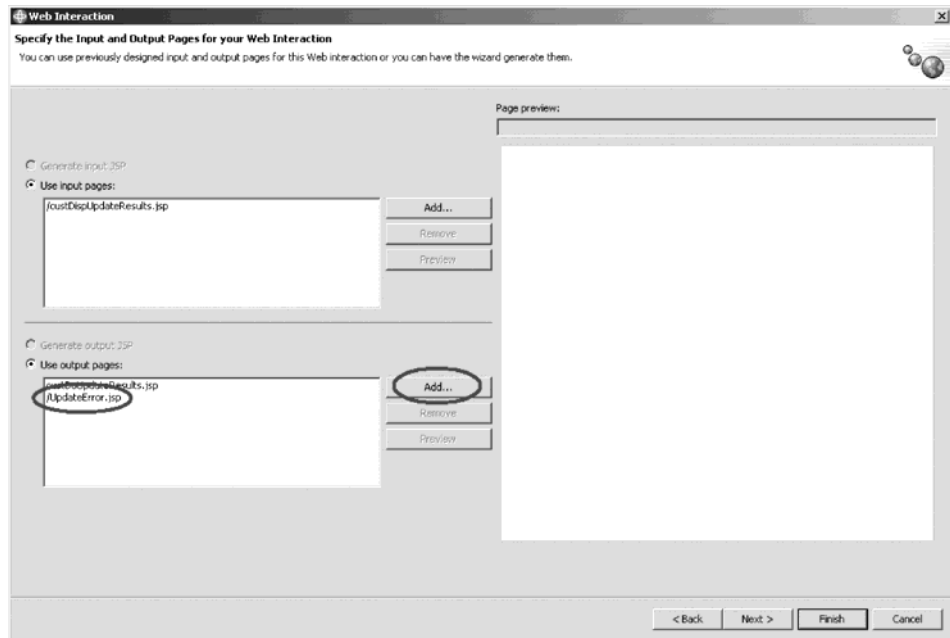
Now that the update complete page (custDoUpdateResults.jsp) is created, you can add the flow control error page.

1. Drill down through your project and find the Web interaction file named **custDoUpdate.wit**.

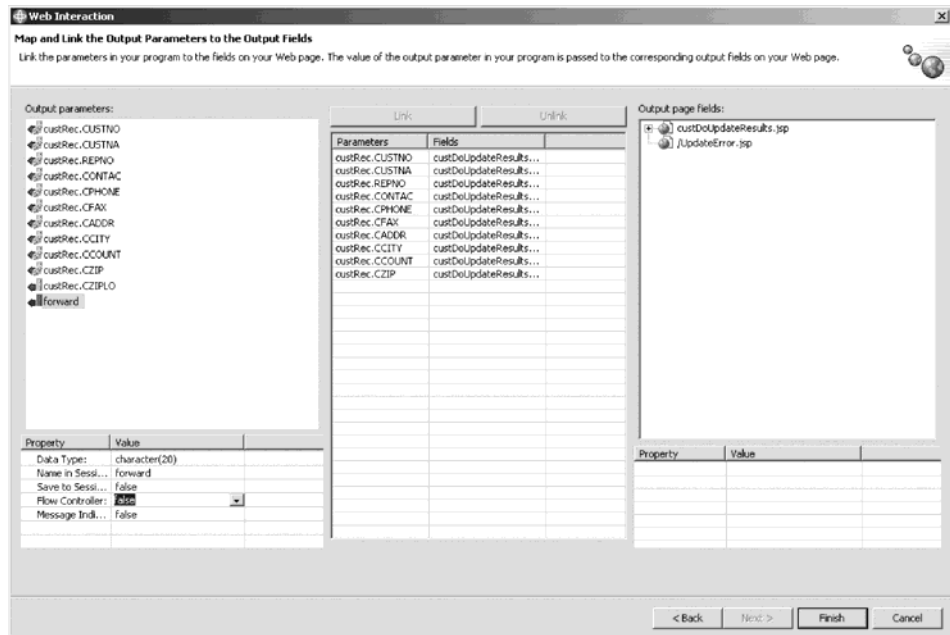


2. Double-click the Web interaction file.
The Web Interaction wizard opens.
3. Click **Next**.
Next you add the flow controlled error page.

- Click **Add** as shown:

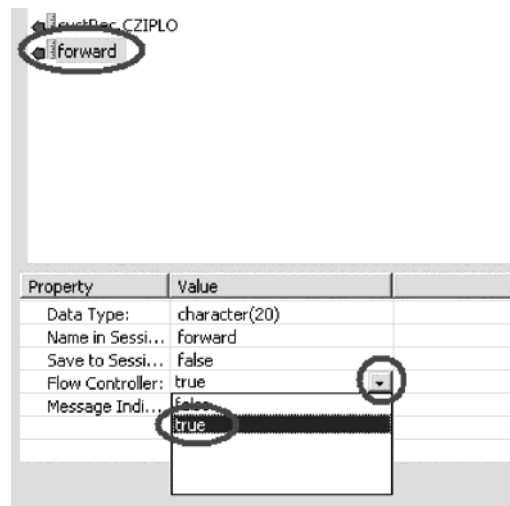


- Select **UpdateError.jsp** on the Output page dialog. Click **OK**.
- Click **Next** until you reach the Map and Link the Output Parameters to the Output Fields page:

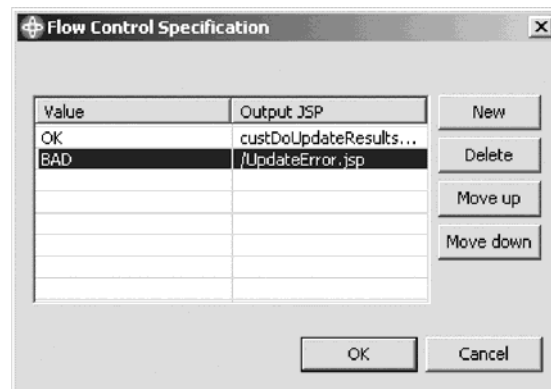


- Under **Output parameters**, select the forward parameter.

- Select the value **true** for the Flow Controller list:



- In the Flow Control Specification dialog, click **New** and type **OK** in the **Value** field.



- Click **New** again.
- Type **BAD** in the **Value** field and select **UpdateError.jsp** as the Output.jsp
- Click **OK**.
You return to the Web Interaction wizard.
- Click **Finish**.

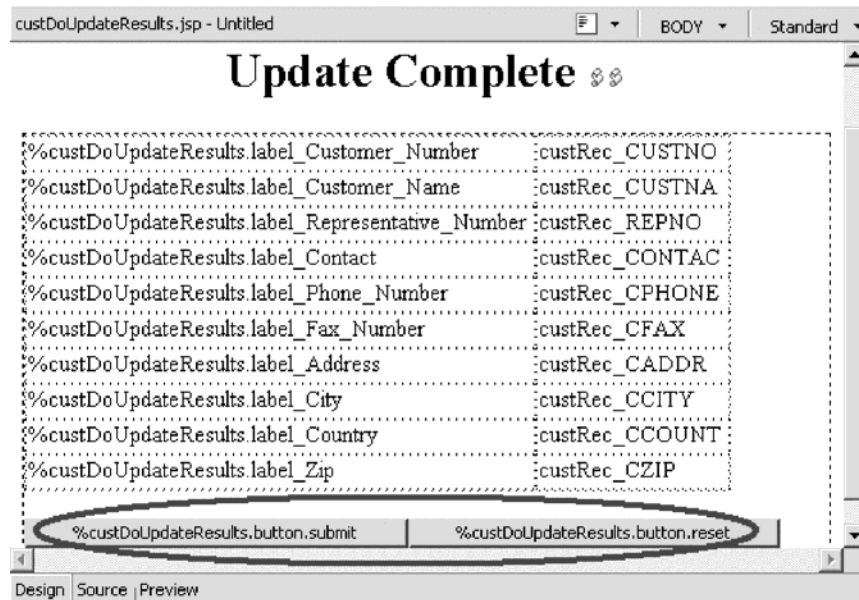
Exercise 5.3: Linking the update complete page to the inquiry page

Next you edit the update complete page (custDoUpdateResults.jsp) to add the link to the inquiry page.

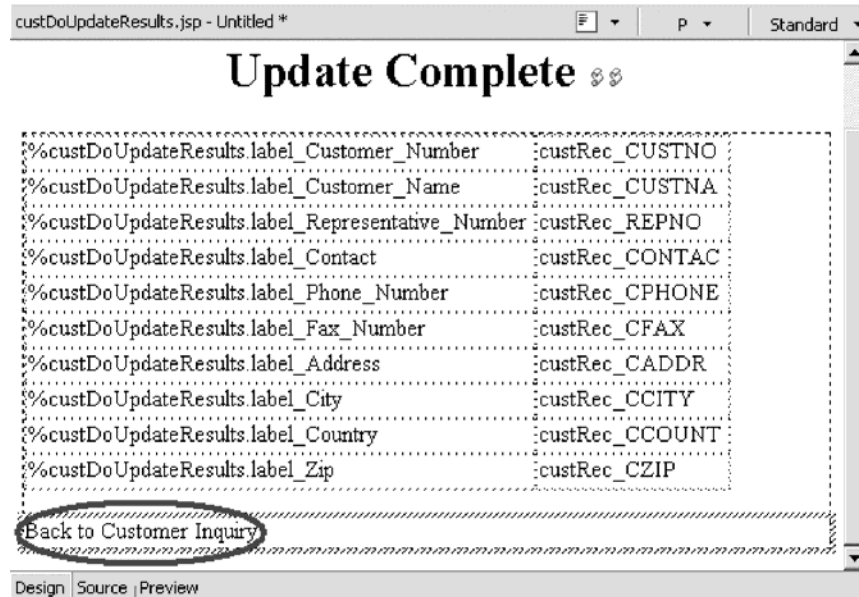
To link the update complete page to the inquiry page:

- Navigate to your **custDoUpdateResults.jsp** file and double-click.

The Page Designer opens.

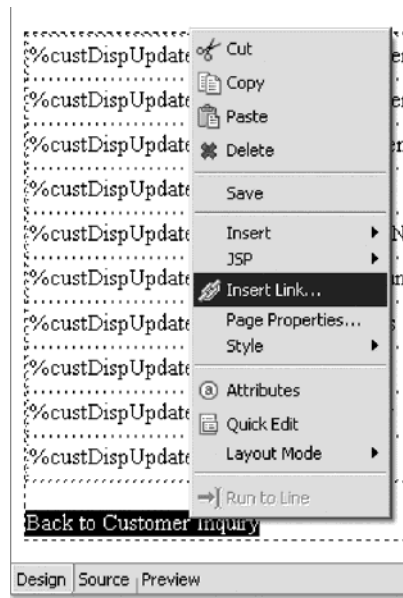


2. Select and delete the **Submit** and **Reset** buttons.
3. Type Back to Customer Inquiry where the Submit and Reset buttons were shown.



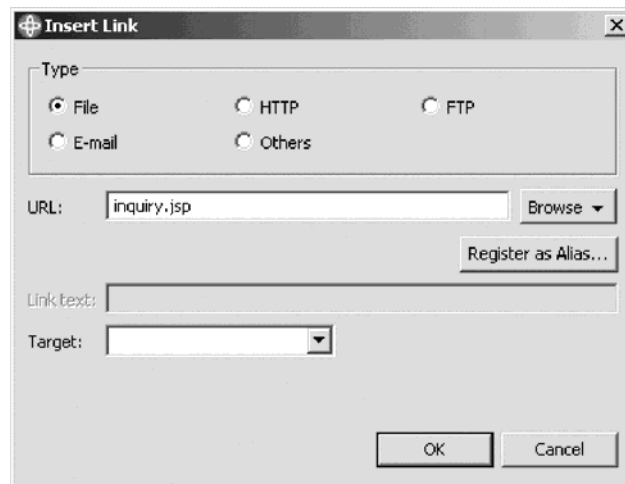
4. Highlight the text and right-click.

5. Click **Insert link** on the pop-up menu.



The Insert Link dialog opens.

6. In the **URL** field, type `inquiry.jsp`.



7. Click **OK**.
8. **Save** and close the JSP file.

Recap

Congratulations! You have completed Module 5: Creating the update complete output page. You should now understand how to:

- Generate a new output page using the Web Interaction wizard
- Add the flow control error page
- Link the update complete page to the inquiry page.

Now that you have created the update complete output page, you can continue to Module 6: Running the updated Web application.

Chapter 6. Module 6: Running the Web application

In this module, you will learn how to run the updated Web application.

In order to run the updated Web application, there are several steps you will need to learn about and follow:

- Restarting the server
- Testing the Web application

Note: To access iSeries data and resources you will need to have a job description setup and associated with your user profile on your iSeries system.

In order to accomplish these learning objectives, there are several steps that are involved, including:

Exercise 6.1: Restarting the server

Exercise 6.2: Testing the Web application

The exercises in this module must be completed in order. Start with Exercise 5.1 when you are ready to begin.

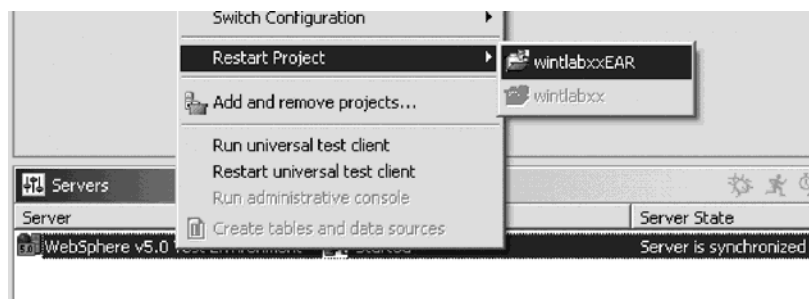
Length of time:

This module will take approximately 15 minutes to complete.

Exercise 6.1: Restarting the server

If the server is still running from the last time you ran the Web application, it needs to be restarted. The server will pick up changes in the Web application only when it is restarted. If the server is not started, then you must start the server before you can restart the project.

1. In the Web perspective locate the Servers view at the bottom of the workbench. Click the **Servers** tab.
2. Right-click the server and select **Start** on the pop-up menu.
The server will show Started status when the start operation completes.
3. Right-click the server and select **Restart Project** on the pop-up menu.



4. Then click your project on the pop-up menu.
5. Continue with the Testing section.

Exercise 6.2: Testing the Web application

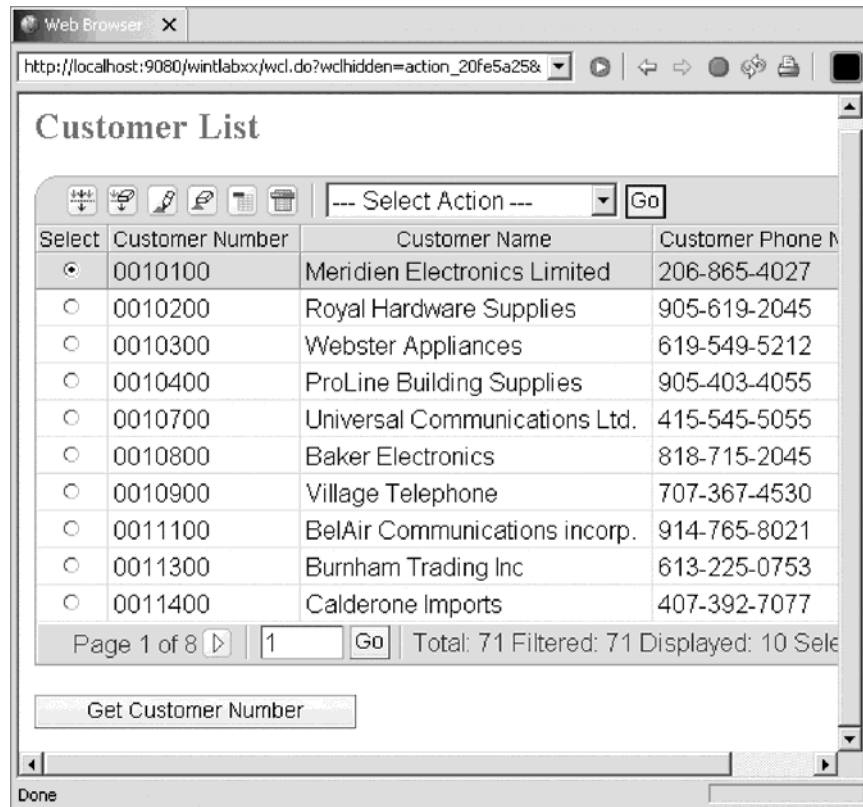
To test the customer list page:

1. Right-click **Inquiry.jsp** and click **Run on Server** on the pop-up menu.
2. Click **OK** to accept the defaults on the dialog that opens.

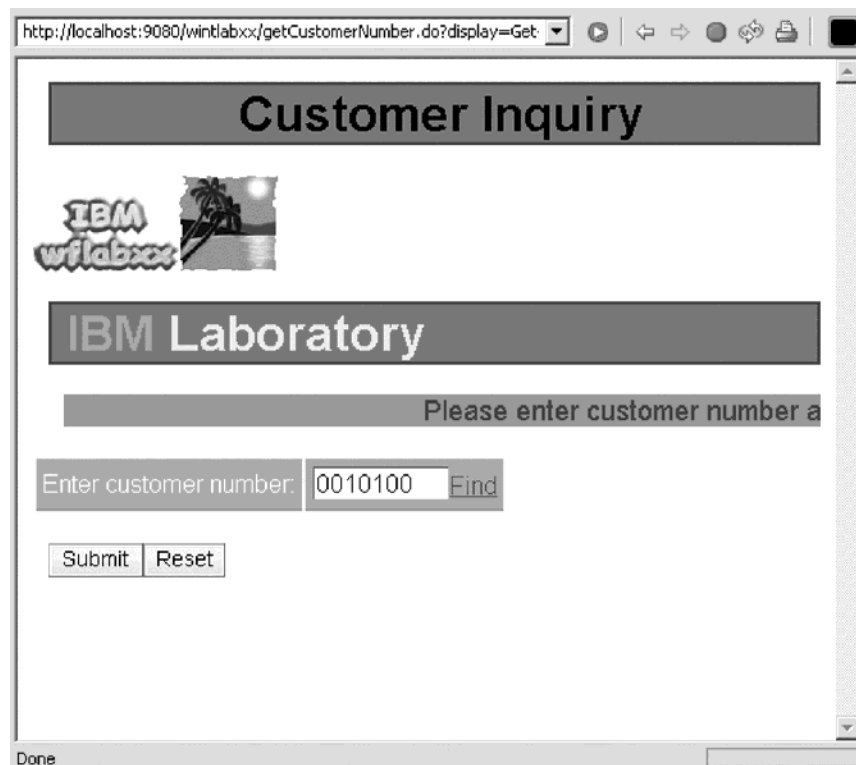


3. Click the **Find** link.
The Customer list page opens.

4. Select the first entry on the list.

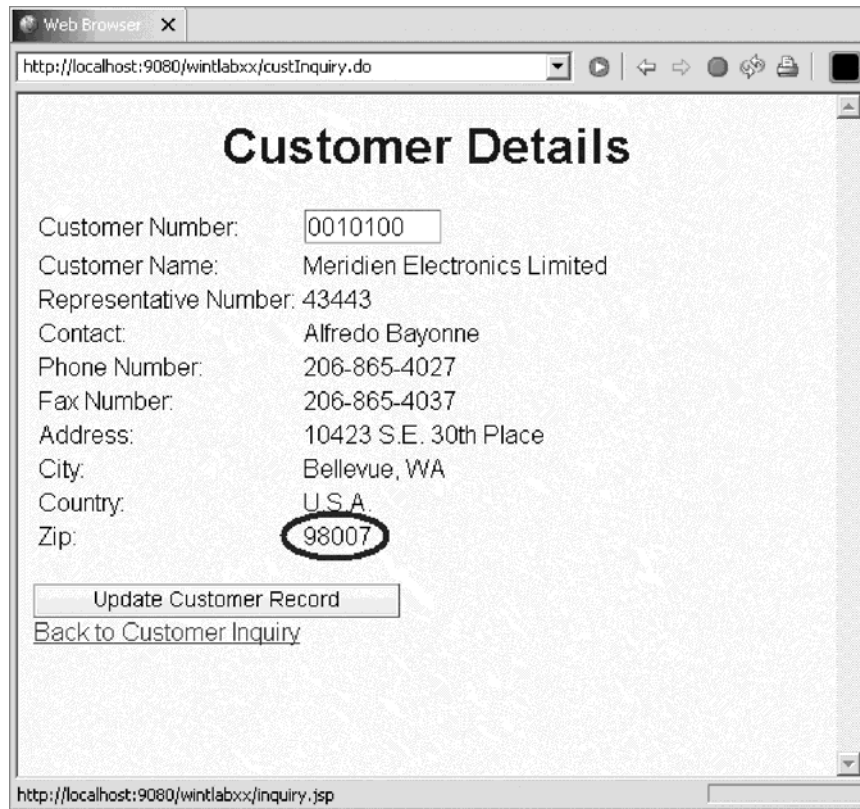


5. Click the **Get Customer Number** button.



6. Click the **Submit** button.

The Customer details page opens showing the details of the selected customer.



To test the update record and update complete pages:

1. When the information is displayed, click the **Update Customer Record** button.

Web Browser x
http://localhost:9080/wintlabxx/custDispUpdate.do

Update Customer Record

Customer Number: 0010100
Customer Name: Meridien Electronics Limited
Representative Number: 43443
Contact: Alfredo Bayonne
Phone Number: 206-865-4027
Fax Number: 206-865-4037
Address: 10423 S.E. 30th Place
City: Bellevue, WA
Country: U.S.A.
Zip: 98006

Done

2. Change some of the fields and click the **Submit** button.
You should see the confirmation page with the new values.

Web Browser x
http://localhost:9080/wintlabxx/custDoUpdate.do

Update Complete

Customer Number: 0010100
Customer Name: Meridien Electronics Limited
Representative Number: 43443
Contact: Alfredo Bayonne
Phone Number: 206-865-4027
Fax Number: 206-865-4037
Address: 10423 S.E. 30th Place
City: Bellevue, WA
Country: U.S.A.
Zip: 98006

[Back to Customer Inquiry](#)

3. Click the **Back to Customer List** link and you will return to the customer list page.

Recap

Congratulations! You have completed Module 6: Running the updated Web application. You should know understand:

- How to restart the server
- How to test the updated Web application.

In this tutorial, you learned how easy it was to create a new input page to view a list of customers (custList.jsp) and use an existing output page (result.jsp) to create new output pages to update customer records and confirm updates were completed on the iSeries (custDispUpdateResults.jsp and custDoUpdateResults.jsp) for an existing e-business customer inquiry application. While you created these pages, you learned how easy it was to use the iSeries wizards in Development Studio Client to link pages together to form an application with multiple Web interactions. You then learned how easy it was to run the updated customer inquiry application in the WebSphere Test Environment that is part of Development Studio Client.

Appendix. Notices

Note to U.S. Government Users Restricted Rights — Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

Lab Director
IBM Canada Ltd. Laboratory
8200 Warden Avenue
Markham, Ontario
Canada
L6G 1C7

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 1992, 2004. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

Programming interface information is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks

IBM
iSeries
WebSphere

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

ActiveX, Microsoft, SourceSafe, Visual C++, Visual SourceSafe, Windows, Windows NT[®], Win32, Win32s and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX[®] is a registered trademark of The Open Group.

Other company, product, and service names may be trademarks or service marks of others.



Printed in USA