Debugging Applications in IBM Rational Application Developer

What this exercise is about

IBM Rational Application Developer v6.0 includes an Integrated Test Environment which can be used to debug your J2EE 1.4 applications. In this exercise will have you debug a J2EE 1.4 application running in the Integrated Test Environment using Cloudscape.

Lab Requirements

List of system and software required for the student to complete the lab.

- IBM Rational Application Developer v6.0, with the WebSphere Application Server v6.0 Integrated Test Environment is also required.
- Windows 2000 Professional Service Pack 4 is required for this lab exercise.
- The lab source files (LabFiles60.zip) must be extracted to the root directory (i.e., C:\).
- Experience with previous versions of Rational Application Developer and the J2EE programming model are also required.

What you should be able to do

At the end of this lab you should be able to:

- Prepare a J2EE 1.4 application for testing on the Integrated Test Environment Application Server.
- Set up Cloudscape as a data source for your application before testing the application.
- Test you application on the Integrated Test Environment Application Server.
- Setup your Integrated Test Environment Server for debugging.
- Debug you application in IBM Rational Application Developer.

Introduction

The J2EE 1.4 application which you will be debugging is a simple banking application named WebSphereBank. It is composed of EJB, Web, and Application Client modules. In the EJB module there is a session bean and an entity bean. Before being able to debug the application, you will need to start your sever in debug mode.

In this lab, you will be debugging the WebSphereBank application running on the WebSphere Application Server v6.0 Integrated Test Environment within IBM Rational Application Developer v6.0. The process you will use to setup and debug the application on the WebSphere Application Server v6.0 Integrated Test Environment within IBM Rational Application Developer v6.0 is very similar to the procedures required to test an application running on a remote application server.

Exercise Instructions

**** NOTE ** Solution instructions are normally provided at the end. The solution is not provided in this case because you need to do the exercises in order to understand Java tools and there is no final solution to import. To go through the lab, start at Part One assuming you have met the requirements in the section "User Requirements" stated above.**

Reference Variable	Windows Location	AIX/UNIX Location
<LAB_FILES>	C:\Labfiles60	/tmp/Labfiles60
<RAD_HOME>	C:\Program Files\IBM\Rational\SDP\6.0	

Part 1: Workspace Setup

____ 1. Backup your server configuration. This will preserve your current server configuration. You will restore your server configuration at the end of the lab exercise.

__ a. Open a Windows Command Prompt and navigate to the following directory:

<RAD_HOME>**\runtimes\base_v6\bin**

__ b. Backup the server configuration by issuing the following command:

```
backupConfig "c:\Program Files\IBM\Backup.zip"
```

____ 2. Start Rational Application Developer.

__ a. Start->Programs->IBM Rational ->IBM Rational Application Developer V6.0 ->Rational Application Developer.

__ b. For workspace, specify **c:\Labfiles60\IRAD_DebugApplLocal\workspace.**

NOTE: If you receive an Auto Launch Configuration Change Alert window, select **Yes** to change the auto-launch workspace to the path of the current lab.

____ 3. Open the J2EE Perspective if it is not already open

__ **a.** From the menu select Window > Open Perspective > Other

__ **b.** Select J2EE and click **OK**

Part 2: Application and Cloudscape Setup

____ 1. Back in IBM Rational Application Developer, import the WebSphereBank application for testing.

__ a. Select File > Import...

__ b. Select EAR file and click **Next.**

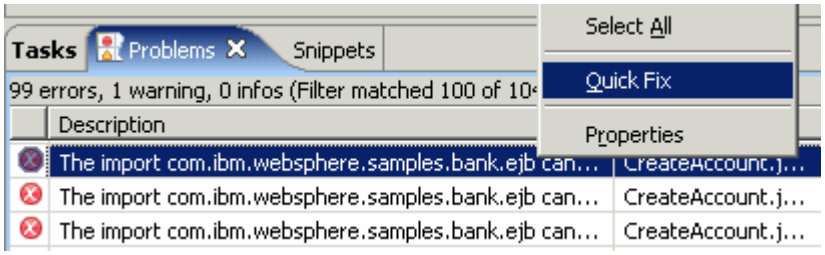
__ c. Click **Browse...** and navigate to **c:\LabFiles60\IRAD_DebugApplLocal\WebSphereBank.ear** and click **Open**.

__ d. For the Project name, enter **WebSphereBank.**

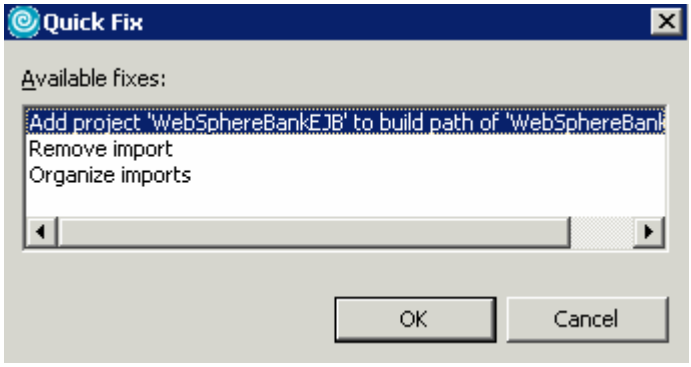
__ e. Click **Finish**.

____ 2. Add Java Build Path to clean up errors. If you select the Problems tab, you will notice a list of errors and warnings.

__ a. Right click on an error and select **Quick Fix.**

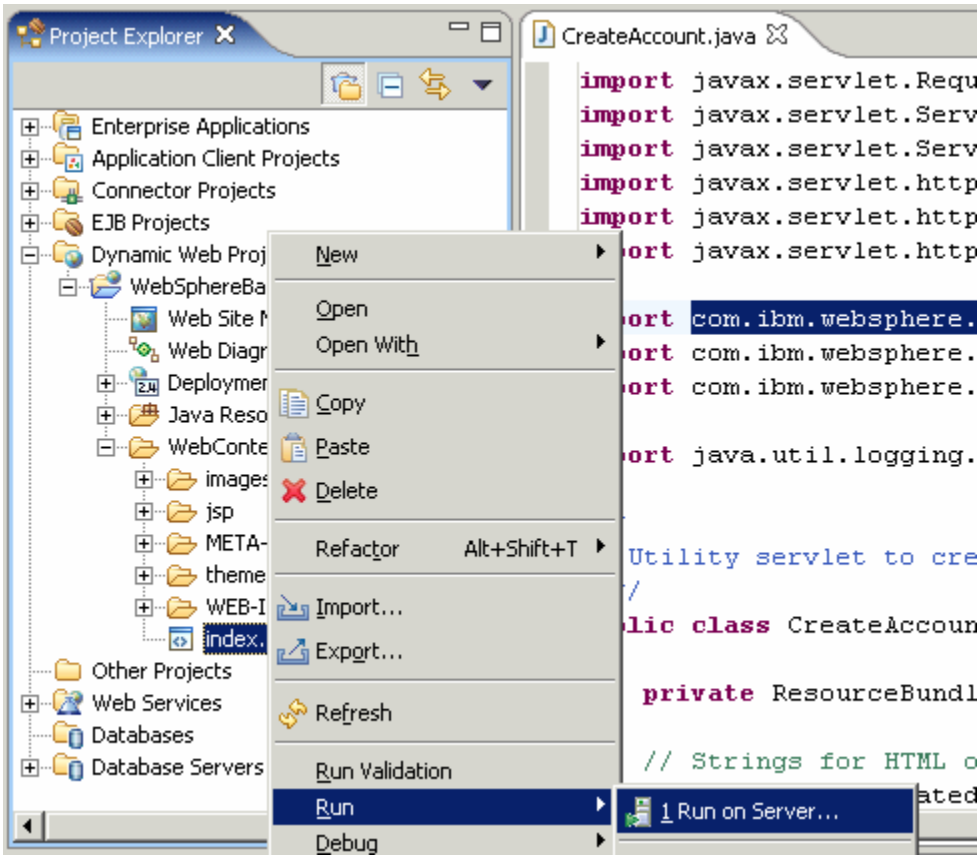


__ b. A list of available fixes will appear. Select “Add project ‘WebSphereBankEJB’ to build path of ‘WebSphereBankWeb’” and select **OK.** The errors should disappear.



____ 3. Start the server with the WebSphereBank project while initializing the database and datasource.

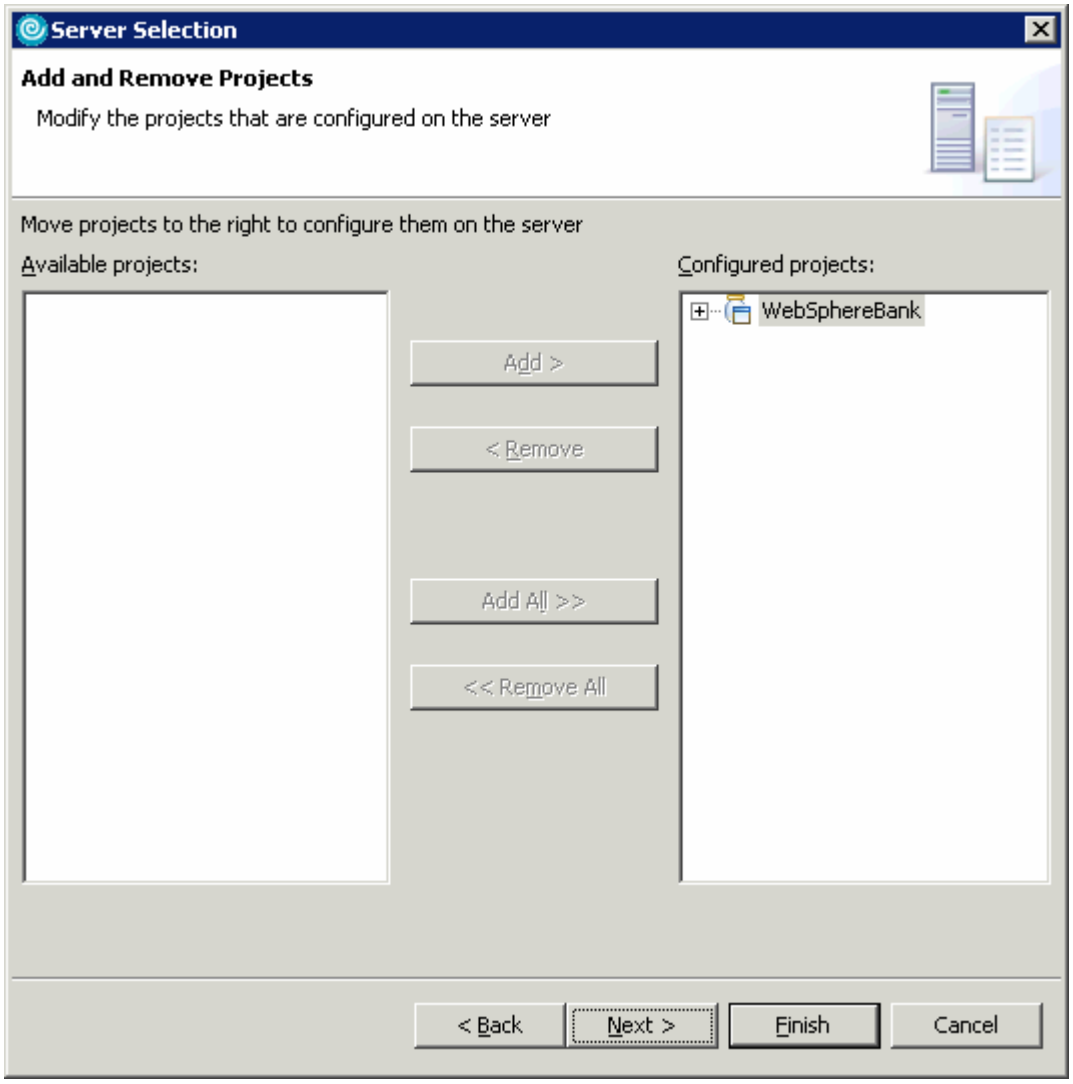
__ a. In the Project Explorer view, navigate to **Dynamic Web Projects > WebSphereBankWeb > WebContent** and right click on index.html.



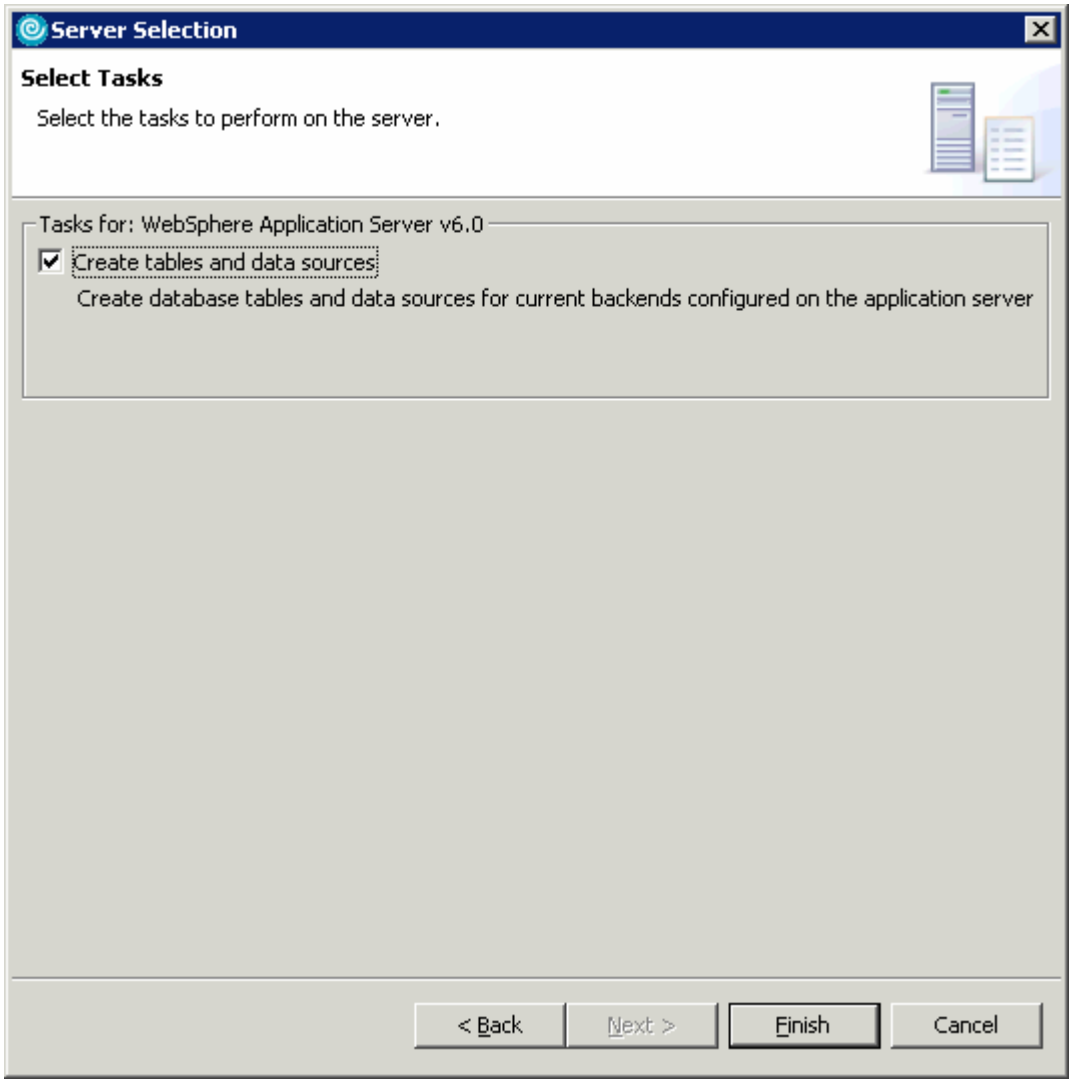
__ b. Select the **Run > Run on Server** option to open the Server Selection window.

__ c. On the Define a New Server page of the Server Selection window make sure the **Choose an existing server** option is selected as well as the server WebSphere Application Server v6.0. Then click on the **Next** button.

__ d. On the **Add and Remove Projects** page, make sure that WebSphereBank is added to the Configured projects list on the right side of the window. Then click on the **Next** button.

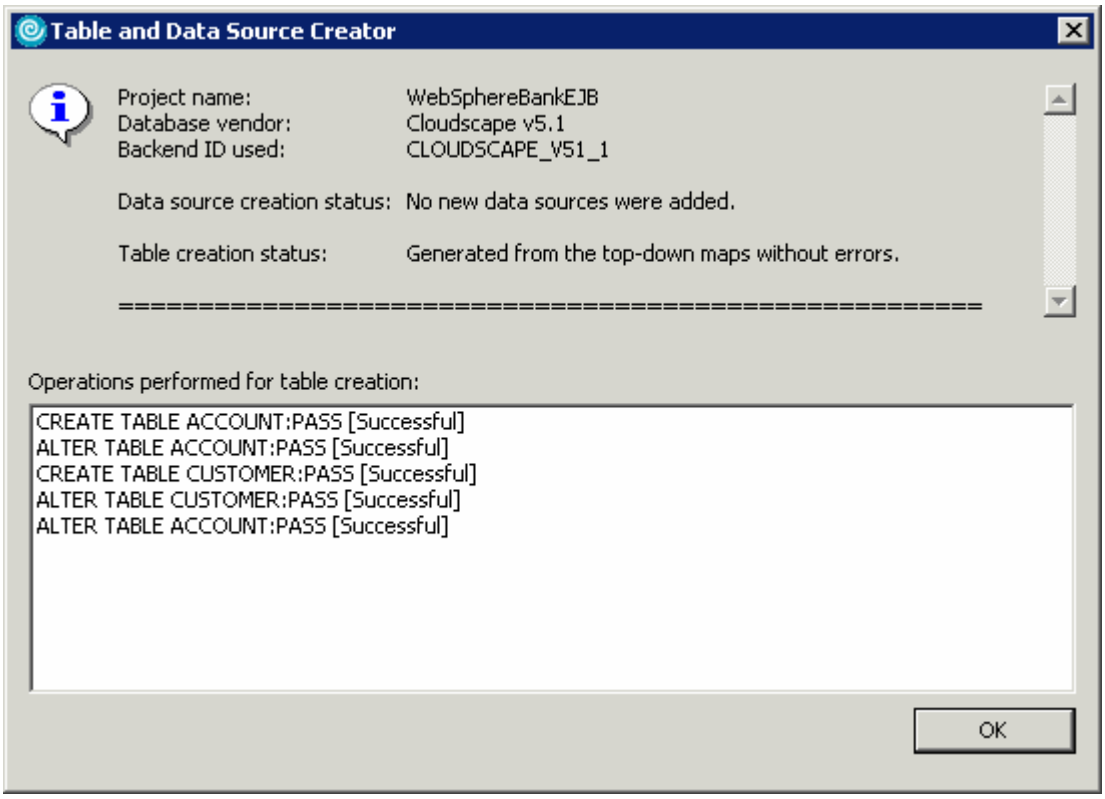


__ e. On the **Select Tasks** page, click on the Create tables and data sources checkbox. Then click the **Finish** button.



__ f. Click **Finish**.

__ g. You should see a window like the one below if the database has been setup successfully.



NOTE: After the code has been generated, browse to EJBProjects > WebSphereBankEJB > ejbModule. Notice the classes and the packages which were created. In the com.ibm.websphere.samples.bank.ejb.websphere_deploy package are common classes used to support persisting entity beans running in WebSphere regardless of the data source. The com.ibm.websphere.samples.bank.ejb.websphere_deploy.CLOUDSCAPE_V5_1 classes are specific for persisting entity beans to a Cloudscape database.

__ h. Click **OK**.

Part 3: Test Application

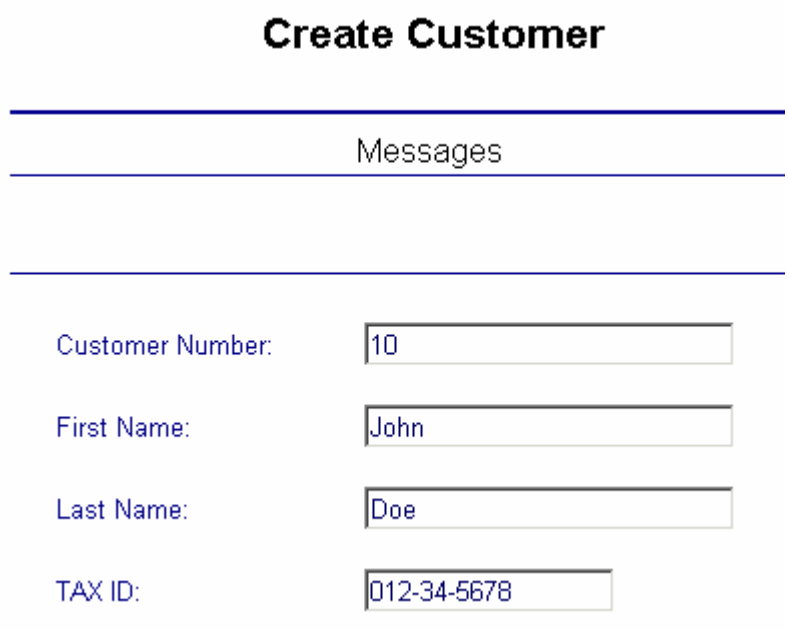
____ 1. After the application has made it to the server, open your browser to the WebSphereBank homepage. Open a Web Browser and navigate to the following URL:

`http://localhost:9080/WebSphereBankWeb`

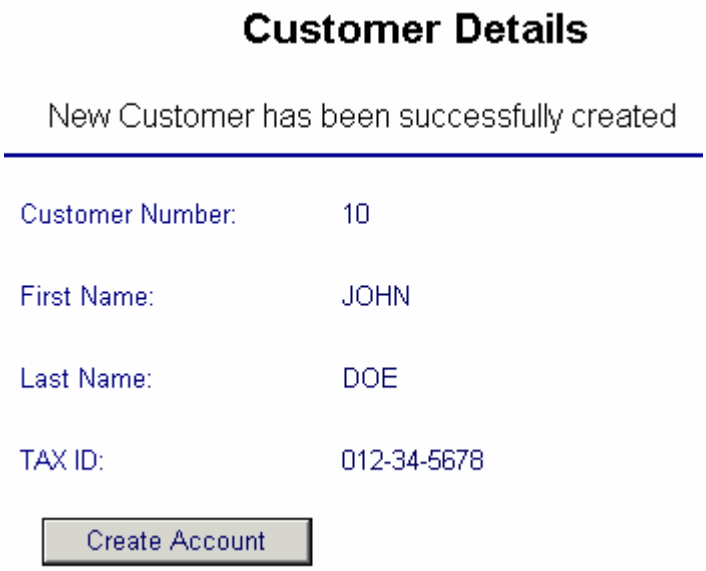
It may take some time, but eventually the homepage of the WebSphereBank Application will come up. The WebSphereBank Application is composed of several processes (Create Customer, Create Account, Transfer funds, etc). First you will test the application by creating a customer, then creating two accounts before transferring funds between the two accounts.

____ 2. Click on the **Create Customer** link.

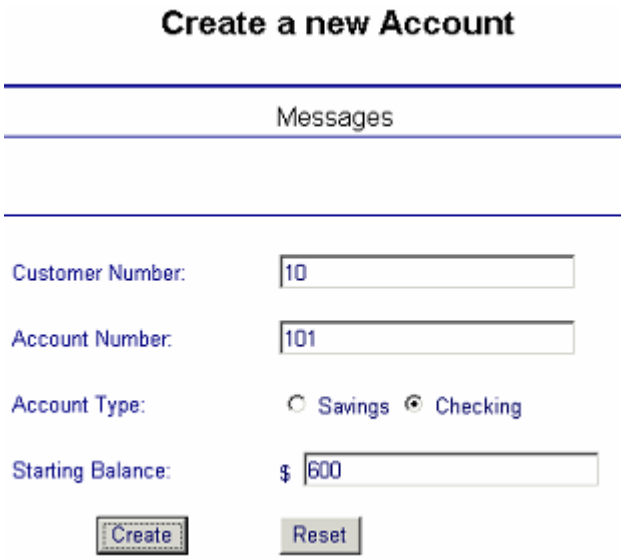
____ 3. Enter Customer Number, Name and Tax ID. Click **Create**.



____ 4. You will see details for customer created. Click **Create Account**.

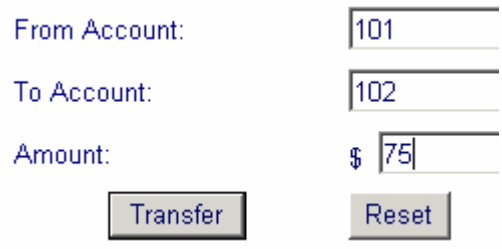


____ 5. Enter **101** for the Account Number, checking for account type and **600** for the starting balance. Click **Create**.



____ 6. Create a savings account with a number of **102** and a balance of **400**.

____ 7. With two accounts created, you can now transfer funds between these two accounts. Click on the **Transfer Funds** link. For the From Account enter **101,** for the To Account enter **102**, and for the amount enter **75**.



____ 8. Select **Transfer**.

____ 9. The messages displayed will reflect incorrect account balances. We will debug this application to determine the cause of the error.

Part 4: Server Setup

____ 1. Setup the server to run in debug mode.

__ a. Open a Web Browser and navigate to the following URL:

`http://localhost:9060/ibm/console`

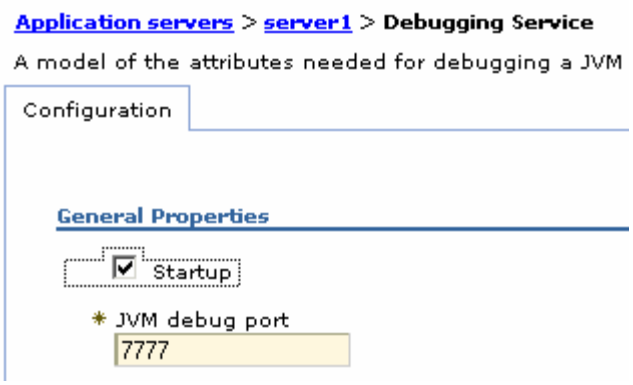
__ b. When prompted for a User ID, enter **wsdemo** to log in.

__ c. Navigate to servers->Application servers.

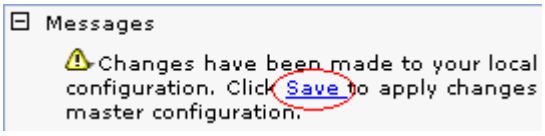
__ d. Click on server1.

__ e. Scroll down to Additional Properties and click on the Debugging Service link**.**

__ f. Check the **Startup box**, and select **OK**.



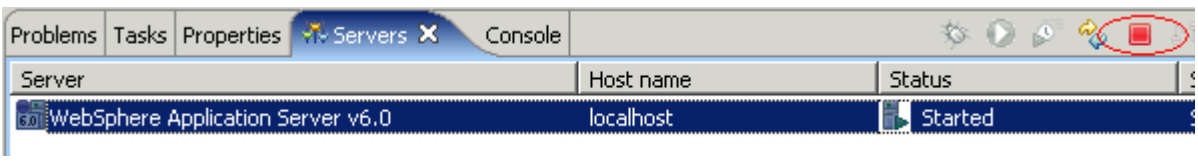
__ g. Click **Save** to save your changes.



__ h. Click the **Save** button.

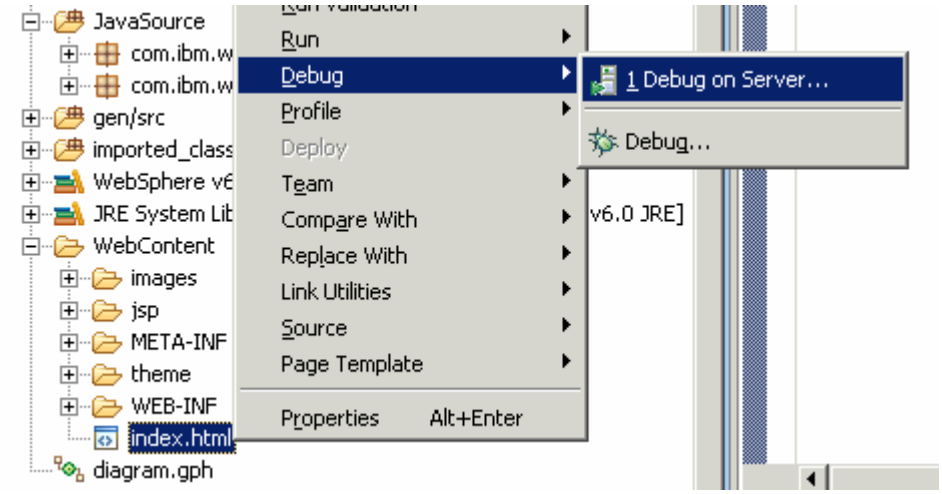
____ 2. Restart WebSphere Application Server in Debug Mode.

__ a. From the Servers Tab, select the server and select the **Stop the server** icon.



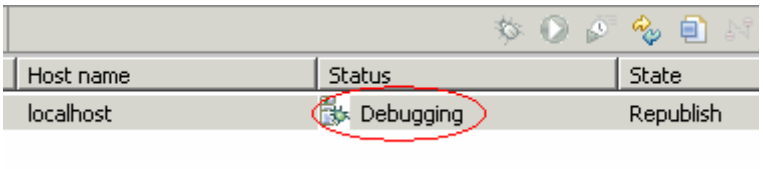
__ b. When the server has stopped, go to the Project explorer and navigate to Dynamic Web Projects -> WebContent -> index.html.

__ c. Right click on the index.html entry and select Debug > Debug on Server….



__ d. In the Server Selection window, confirm that the **WebSphere Application Server v6.0** option is selected under localhost, then click the **Finish** button.

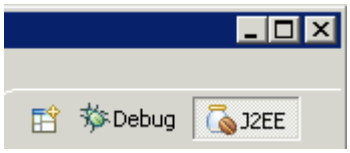
__ e. When the server start has completed, you should see that the status of the server now indicates debugging.



Part 5: Debug Application

____ 1. Click yes on the Perspective Switch Confirmation window.

____ 2. Switch back to the J2EE perspective now that the Debug perspective is loaded by clicking on the J2EE Icon in the upper right hand corner of the Rational Application Developer window.

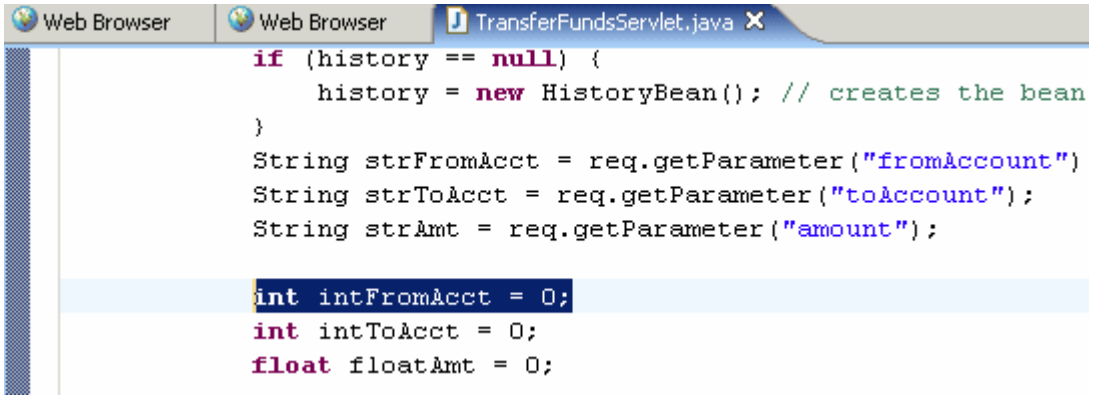


____ 3. Set your break points.

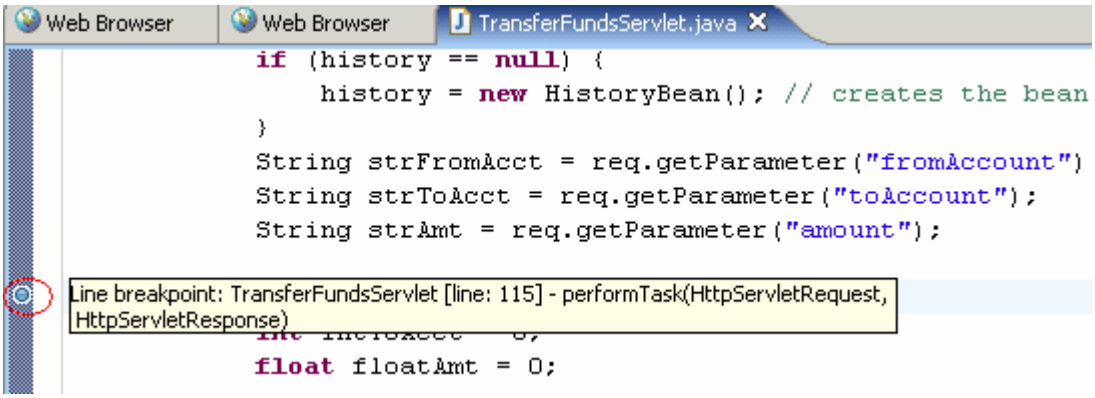
__ a. In the Project Explorer, navigate to Dynamic Web Projects > WebSphereBankWeb > Java Resources > JavaSource > com.ibm.websphere.samples.bank.web

__ b. Double click on the TransferFundsServlet.java source file.

__ c. Scroll down to the statement shown below:



__ d. Double click on side bar to set line break point.



__ e. Scroll down further and set a break point at line 161.

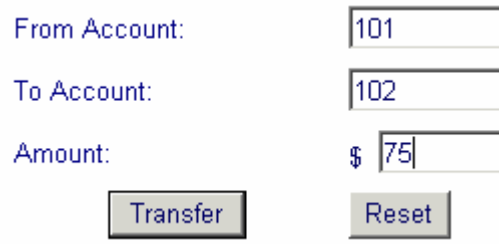
```
        //After page validation if there are no errors
        if (errmsg.size() == 0) {
            try {
                TransferLocal transfer = transferHome.create();
                transfer.transferFunds(intFromAcct, intToAcct, floatAmt);

                fromBalance = dF.format(transfer.getBalance(intFromAcct));
                toBalance = dF.format(transfer.getBalance(intToAcct));
```

____ 4. Open WebSphereBank application in your browser.

__ a. In a new web browser window, open the following URL:
http://localhost:9080/WebSphereBankWeb

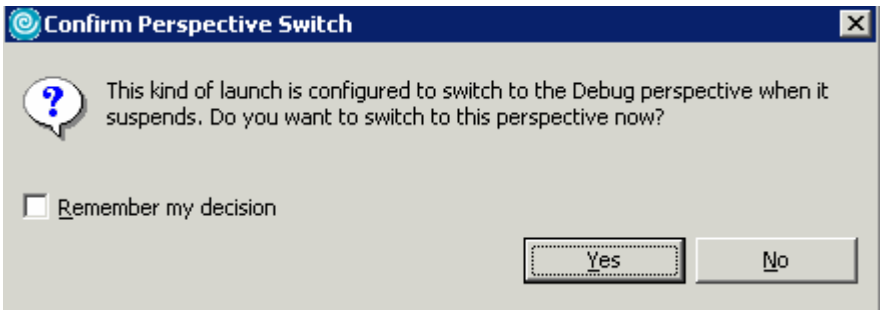
____ 5. Initiate a funds transfer.

__ a. Click on the **Transfer Funds** link. For the From Account enter **101,** for the To Account enter **102**, and for the amount enter **75**.

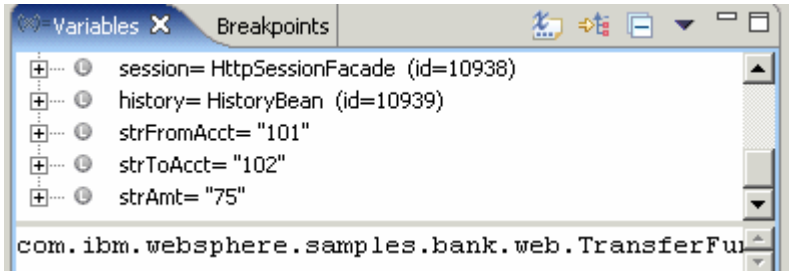


__ b. Click **Transfer**.

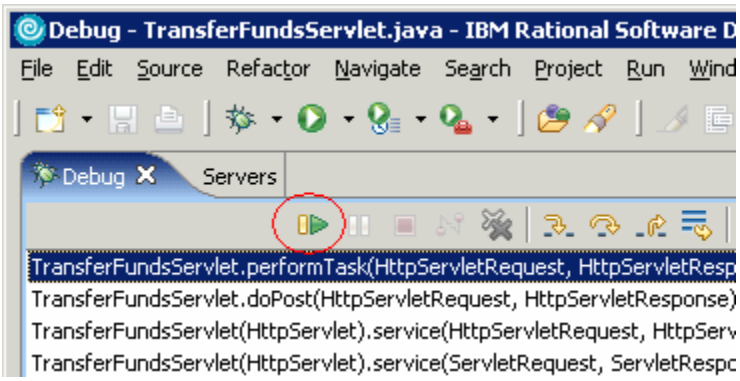
__ c. You should see the screen below indicating your breakpoint has been hit. Click **Yes** to switch to the debug perspective.



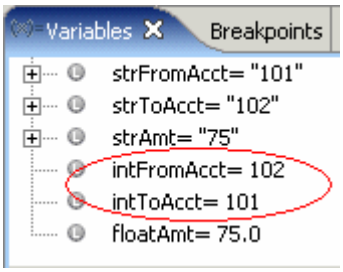
__ d. Click on the Variables tab and scroll down to the bottom to display the variables for FromAcct, ToAcct and Amt.



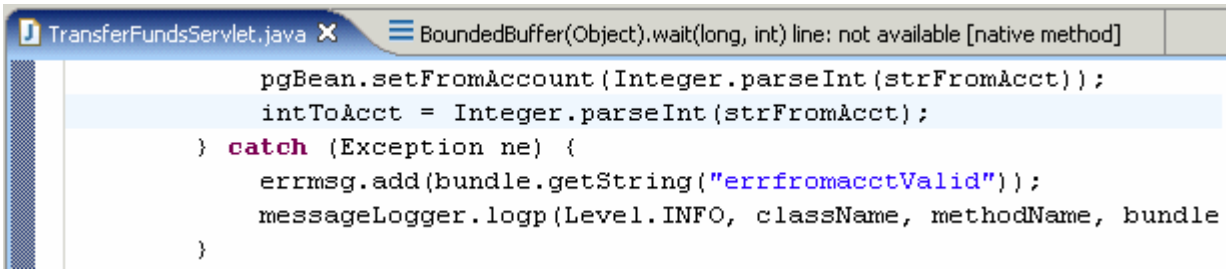
____ 6. Click on the Debug tab and click on the **Resume** icon.



____ 7. You stop at the next break point. Now look at the variables. You will notice that the intFromAcct and intToAcct values are incorrect.

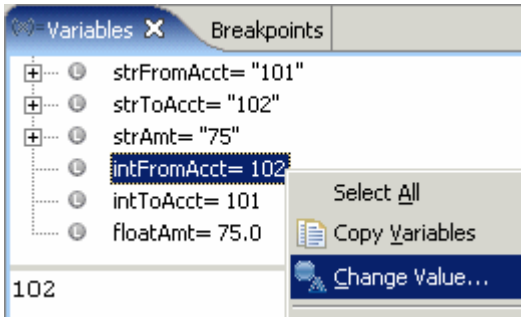


____ 8. Scrolling up from the break point, you can see that intToAcct is being set from strFromAcct. Similarly, intFromAcct is being set from strToAcct**.**



____ 9. Correct values.

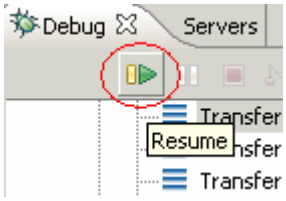
__ a. Right click on intFromAcct variable, select **Change Value**, and an edit window will appear.



__ b. Change the value from **102** to **101**. Select **OK**.

__ c. Repeat the process changing the value of intToAcct from **101** to **102**.

__ d. Click the **Resume** icon



__ e. Messages in web browser will indicate transfer completed. New account balances will reflect the correct transfer request.

____ 10. You can now stop your server and exit Rational Application Developer.

Part 6: Restore your server

____ 1. Restore your server configuration. This will return your server configuration to its original state.

__ a. Open a Windows Command Prompt and navigate to the following directory:

<RAD_HOME>\runtimes\base_v6\bin

__ a. Restore the server configuration by issuing the following command:

```
restoreConfig backupConfig "c:\Program Files\IBM\Backup.zip"
```

What you did in this exercise

In this exercise, you debugged a J2EE 1.4 application. You installed the application, setup your database and data source, tested the application, set break points, looked at code, inspected variables and changed there value.

This page is left intentionally blank.