

Programming Assignment 6
COMP150CBD
Creating a Session Beans in Eclipse/Lomboz
Due before class on Thursday, November 4, 2004

Updated 10/24/2004 at 8:25 pm

Updated 10/28/2004 at 4:20 pm

In this homework, you will create both a stateless and a stateful session bean. You will not need to submit any code for the stateless session bean; instead, you will answer written questions and submit only the answers to those. For the stateful session bean part (Part II) you will submit the code for your server. This assignment is worth 75 points.

Part I: Creating a Stateless Session Bean using Eclipse/Lomboz

Work through the tutorial at <http://www.tusc.com.au/tutorial/html/chap3.html> and answer the questions included below as you go along. We have also included corrections to each section of the tutorial.

Create J2EE Project

Errata

1. The tutorial tells you to check your library settings as you did for the previous assignment. You don't need to do this with Eclipse >= 3.0.

Questions

1. What does adding the Web Module name do? What is a Web Module?
2. What does adding the Ejb Module name do? What is an Ejb Module?

Create Stateless Bean

Notes

1. Make sure you went through the database setup in the last assignment, Programming Assignment 5. If your tables aren't set up correctly and filled with data, this tutorial won't work.

Errata

1. Many more comments than what are shown in the tutorial are created when you create the stateless session bean. Ignore them, they are not important for this assignment.
2. There are 8 files, not 7, under the MyStoreMgr/META-INF node. Also, some of the names are different.
3. There is no ejbGenerate.xml file under the MyStoreMgr/META-INF node. Use xdoclet.xml instead.
4. There is no jboss.xml file.

Questions

1. What is the difference between a stateless session bean and a stateful session bean? Consider a Calculator bean that takes as input a string with an arithmetic expression, evaluates the expression, and returns the value of the expression.

For example, if you send the bean "2+5", it returns "7". Which type of session bean would you use for this?

Setup DAO

Errata

1. You don't need to customize `ejbGenerate.xml/xdoclet.xml` for setting up a DAO. That is automatically generated for you. You should just read through the descriptions of `ejbGenerate.xml` remembering that the definitions are now placed in `xdoclet.xml`.

Create DAO Interface

Questions

1. When you add the tag to generate the DAO interface to the `StoreAccessBean` class, you are telling Lombok to generate a certain file. What specific file does that instruction generate? Where is it located?
2. When you add the EJB to module, what does this actually do? What file does it modify? Hint: Check your xml files.

Add Business Method

Questions

1. Why did we set Method Type as Business and Interface Type as Remote when we created the new method? What do those mean in terms of the EJB interfaces we discussed in class (`EJBHome`, `EJBObject`, `EJBLocalHome`, `EJBLocalObject`)?

Implement DAO Interface

Errata

1. You also need to import `au.com.tusc.session.StoreAccessDAO`.

Add Callback Methods

Errata

1. The tutorial doesn't state it, but you are modifying the `StoreAccessBean.java` file in this section.
2. When you add the protected member variable, you need to declare it with the full package name: `protected javax.ejb.SessionContext ctx`

Questions

1. Why do we need these callback methods? What is a `SessionContext`?

Create your Test Client

Questions

1. When you call the `loginUser()` method of your bean, with which EJB interface are you interacting: `EJBHome`, `EJBObject`, `EJBLocalHome`, or `EJBLocalObject`?
2. List the classes and methods that are invoked when you call `loginUser()` in order of invocation. You will need to examine the generated files in `ejbsrc` to understand how everything fits together.

Part II: Creating a Stateful Session Bean in Eclipse/Lomboz

Create a server for calculating compound interest for a savings account. The interest calculator must be able to tell what the balance will be after some number of years given a starting balance, an interest rate. It must also be able to calculate and display the amount of interest earned over a period of years. Use the following formula for calculating compound interest:

$$A = P \times (r + 1)^N,$$

where A = Amount (future value),
P = Principal,
r = Rate of Interest, and
N = Number of Years compounding

Your server should conform to the Remote interface provided on the assignment page. We have also provided a client for the InterestCalculator on the assignment page. You can use this to make sure your EJB server is working properly and provides the correct Remote interface.

Your server will need to define the following as private attributes (check the client for details):

- Interest rate, which is input by the user of the client
- principal amount, which is input by the user of the client
- length of term in years, which is input by the user of the client
- session context, which is of type SessionContext. Using the SessionContext getEJBObject method provides a reference to the current EJBObject similarly to the use of "this" reference, which isn't allowed in EJB.

And perform the following operations:

- public void setPrincipal(double)
- public void setInterestRate(double)
- public void setTerm(int)
- public double getBalance()
- public double getInterestEarned()

The following are required methods:

- public void setSessionContext(SessionContext) – This is a callback method that provides any methods needed for finding information about the container
- public void ejbCreate() – invoked by the create method in the home interface. Must have the same number and type of arguments as the create method. This is where any initialization code is provided.
- public void ejbRemove() – invoked when the remove method of the home interface is invoked. This method should free up any resources that are used.