Writing Enterprise Applications with Java 2 Platform, Enterprise Edition - part2

Document Information

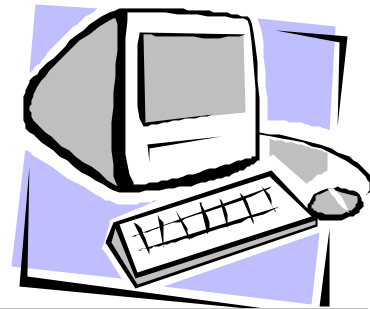|	
Document title:	Writing Enterprise Applications with Java 2 Platform, Enterprise Edition - part2
Document file name:	Writing_Enterprise_Applications_2_JunoYoon
Revision number:	<1.0>
Issued by:	< > junoyoon@orgio.net
Issue Date:	<2001/04/16>
Status:	final

Content Information

Audience	
Abstract	J2EE Entity EJB .-
Reference	Writing Enterprise Applications with JavaTM Platform, Enterprise Edition
	By Monica Pawlan
	(http://developer.java.sun.com/developer/online Training/J2EE/Intro/)
Benchmark information	

Table of Contents

Lesson 2 –   Entity Bean

            Entity Bean   , Lesson 1   .
BonusServlet   Entity Bean   , ID
                                                             .
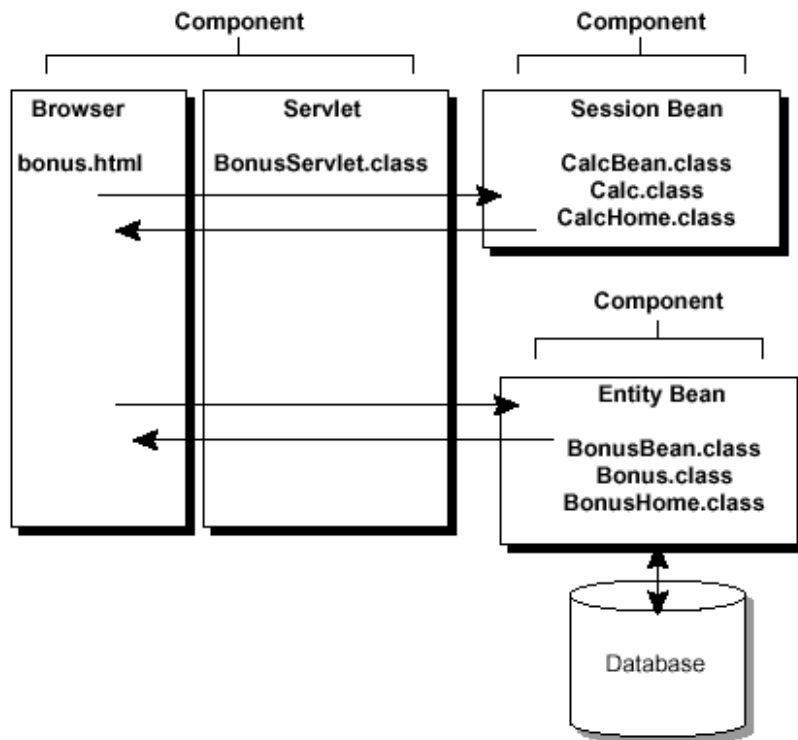
        Lesson 1   , 5
                    .

J2EE SDK   Cloudscape   Entity Bean
                                                                          .

                        ,   SQL
        JDBC   .   SQL
                                                                    .

Entity Bean

        Entity Bean   ,
                    . Entity Bean
            Entity Bean   ,
            .   SQL   JDBC
                        .

Entity Bean   .

        o   Entity Bean   , Entity Bean
                                                        .

        o   ,
                        Commit
                    .

BonusHome

Session Bean   CalcHome   Entity Bean   BonusHome
, findByPrimaryKey   .
ID   .
            ID
        .

create   . BonusServlet
                            create   ,   BonusBean
            ejbCreate   . BonusHome.create
BonusBean.ejbCreate   .
                                    Entity Bean
Entity Bean   .   (ID   )   ,
java.rmi.RemoteException   BonusServlet   .

```
package Beans;
import java.rmi.RemoteException;
import javax.ejb.CreateException;
import javax.ejb.FinderException;
import javax.ejb.EJBHome;
public interface BonusHome extends EJBHome {
    public Bonus create(double bonus, String socsec)
        throws CreateException, RemoteException;
    public Bonus findByPrimaryKey(String socsec)
        throws FinderException, RemoteException;
}
```

Bonus

                            ,   Create   ,
                Entity Bean   . Bonus   Entity
Bean   getBonus   getSocSec
            .

```
package Beans;
import javax.ejb.EJBObject;
import java.rmi.RemoteException;
public interface Bonus extends EJBObject {
    public double getBonus() throws RemoteException;
    public String getSocSec() throws RemoteException;
}
```

BonusBean

BonusBean   Entity Bea n(CMP)   .   , Entity Bean

,

.   Entity Bean
                ,   BonusBean   . Bean
                                                                                  .

    o   Writing Advanced Applications   3
           developer.java.sun.com/developer/onlineTraining/Programming/JDCBook
    o   Java 2 Enterprise Edition Developer's Guide   4
           java.sun.com/j2ee/j2sdkee/techdocs/guides/ejb/html/DevGuideTOC.html

BonusServlet   BonusHome.create   ,   BonusBean.setEntityCotext
                . setEntityContext   EntityContext
        Bean
                          .

Next   ejbCreate   . ejbCreate   Bean
                                                                                  .

ejbPostCreate   ejbCreate   Next   Bean
                          .   .

Bean

.   Bean

Bean

.   Bean   ,

Bean   .

.

o   ejbPassivate   ejbActivate   ,   Bean   /

.

,   .

o   ejbRemove   ,   remove

,   .

o   ejbLoad   ejbStore   ,   Bean

.

getBonus   getSocSec

.   , set<type> (   )

,   Bean

,   .   ,

.

```
package Beans;
import java.rmi.RemoteException;
import javax.ejb.CreateException;
import javax.ejb.EntityBean;
import javax.ejb.EntityContext;
public class BonusBean implements EntityBean {
  public double bonus;
  public String socsec;
  private EntityContext ctx;
  public double getBonus() {
    return this.bonus;
  }
  public String getSocSec() {
  return this.socsec;
  }

  public String ejbCreate(double bonus, String socsec)
    throws CreateException{
  //Called by container after setEntityContext
    this.socsec=socsec;
    this.bonus=bonus;
    return null;
  }
```

```java
public void ejbPostCreate(double bonus, String socsec) {
//Called by container after ejbCreate
}
//These next methods are callback methods that
//are called by the container to notify the
//Bean some event is about to occur
public void ejbActivate() {
//Called by container before Bean
//swapped into memory
}
public void ejbPassivate() {
//Called by container before
//Bean swapped into storage
}
public void ejbRemove() throws RemoteException {
//Called by container before
//data removed from database
}
public void ejbLoad() {
//Called by container to
//refresh Entity Bean's state
}
public void ejbStore() {
//Called by container to save
//Bean's state to database
}
public void setEntityContext(EntityContext ctx){
//Called by container to set Bean context
}
public void unsetEntityContext(){
//Called by container to unset Bean context
}
}
```

Lesson 1   BonusServlet   init   doGet   . init
         Session Bean   CalcBean   Entity Bean   BonusBean   .

```java
public class BonusServletextends HttpServlet {
  CalcHome homecalc;
  BonusHome homebonus;
  Bonus theBonus, record;
  public void init(ServletConfig config)
    throws ServletException{
    try {
```

```
                    InitialContext ctx = new InitialContext();
                    Object objref = ctx.lookup("bonus");
                    Object objref2 = ctx.lookup("calcs");
                    homebonus= (BonusHome)
                    PortableRemoteObject.narrow(
                       objref, BonusHome.class);
                    homecalc=(CalcHome)
                    PortableRemoteObject.narrow(
                       objref2, CalcHome.class);
                 } catch (Exception NamingException) {
                   NamingException.printStackTrace();
                 }
              }
            ....
              }
```

doGet   try   CalcBean   BonusBean   .
                           calcBonus   , Entity Bean
                                                              , BonusHome.create
                 .   BonusHome.findByPrimaryKey
              (ID   )   . Next,   ,
         ,   HTML
              .

catch   (ID   )   .
                                                          ,   ID
                      Entity Bean   .
                                  ,   ,   ,
              HTML   .


```
          try {
              Calc theCalculation;
            //Retrieve Bonus and Social Security Information
              String strMult = request.getParameter(
                     "MULTIPLIER"); //Calculate bonus
              Integer integerMult = new Integer(strMult);
              multiplier = integerMult.intValue();
              socsec = request.getParameter("SOCSEC");
            //Calculate bonus
              double bonus = 100.00;
              theCalculation = homecalc.create();
              calc = theCalculation.calcBonus(multiplier, bonus);
            //Create row in table
              theBonus = homebonus.create(calc, socsec);
              record = homebonus.findByPrimaryKey(socsec);
            //Display data
```

```
            out.println("<H1>Bonus Calculation</H1>");
            out.println("<P>Soc Sec passed in: " +
                    theBonus.getSocSec() + "<P>");
            out.println("<P>Multiplier passed in: " + multiplier +
                    "<P>");
            out.println("<P>Bonus Amount calculated: " +
                    theBonus.getBonus() + "<P>");
            out.println("<P>Soc Sec retrieved: " +
                    record.getSocSec() + "<P>");
            out.println("<P>Bonus Amount retrieved: " +
                    record.getBonus() + "<P>");
            out.println("</BODY></HTML>");
        //Catch duplicate key error
      } catch (javax.ejb.DuplicateKeyException e) {
            String message = e.getMessage();
        //Display data
            out.println("<H1>Bonus Calculation</H1>");
            out.println("<P>Soc Sec passed in: " + socsec +
                     "<P>");
            out.println("<P>Multiplier passed in: " +
                    multiplier + "<P>");
            out.println("<P>Bonus Amount calculated: " +
                    calc + "<P>");
            out.println("<P>" + message + "<P>");
            out.println("</BODY></HTML>");
      } catch (Exception CreateException) {
            CreateException.printStackTrace();
      }
```

Entity Bean   .   ,

Lesson 1   .

Entity Bean

Unix

```
#! /bin/sh
cd /home/monicap/J2EE
J2EE_HOME=/home/monicap/J2EE/j2sdkee1. 2.1
CPATH=. :$J2EE_HOME/lib/j2ee.jar
javac -d . -classpath "$CPATH"
Beans/BonusBean.java Beans/BonusHome.java Beans/Bonus.java
```

Windows

```
cd \home\monicap\J2EE
set J2EE_HOME=\home \monicap\J2EE\j2sdkee1. 2.1
```

```
set CPATH=. ;%J2EE_HOME%\lib\j2ee.jar
javac -d . -classpath %CPATH% Beans/BonusBean.java Beans/BonusHome.java
Beans/Bonus.java
```

Unix

```
cd /home/monicap/J2EE/ClientCode
J2EE_HOME=/home/monicap/J2EE/j2sdkee1. 2.1
CPATH=. :$J2EE_HOME/lib/j2ee.jar:/home/monicap/J2EE
javac -d . -classpath "$CPATH" BonusServlet.java
```

Windows

```
cd \home\monicap\J2EE\ClientCode
set J2EE_HOME=\home\monicap\J2EE\j2sdkee1. 2.1
set CPATH=. ;%J2EE_HOME%\lib\j2ee.jar; \home\monicap\J2EE
javac -d . -classpath %CPATH% BonusServlet.java
```

, J2EE   ,   , Cloudscape

.   .

```
j2ee -verbose
deploytool
cloudscape -start
```

, J2EE   .

Unix

```
j2sdkee1. 2.1/bin/j2ee -verbose
j2sdkee1. 2.1/bin/deploytool
j2sdkee1. 2.1/bin/cloudscape -start
```

Windows

```
j2sdkee1. 2.1\bin\j2ee -verbose
j2sdkee1. 2.1\bin\deploytool
j2sdkee1. 2.1\bin\cloudscape -start
```

.

o
o   Entity Bean

WAR (Web Archive)   , BonusServlet   bonus.html   . BonusServlet
                        ,   J2EE
             .

  o Local Applicatons Window: BonusApp   .
  o [Tools]   : [Update Application Files]   .

    :   Lesson   BonusApp
                     .

Entity Bean

Entity Bean   EJB JAR   , Lesson 1   Session
Bean   .

    :   , Lesson 1   , Entity
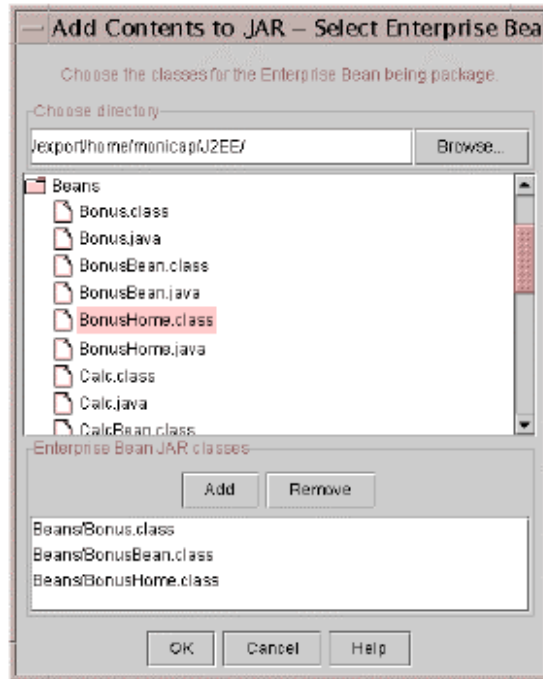 Bean   Session Bean   JAR   .   Bean
         JAR   .

[File]
:
  o [New Enterprise Bean]   .

[Introduction]   **:**

  o [Next >]   .

[EJB JAR]   **:**

  o [Enterprise Bean will go in field]   BonusApp
           .

  o   BonusJar   .
  o   [Add]   .

[Add Contents to JAR]   **:**

  o   ,   .
  o Bonus.class   .
  o [Add]   .
  o BonusBean.class   .
  o [Add]   .

- o   BonusHome.class   .
- o   [Add]   .
- o   [OK]   .

1. BonusJar

[EJB JAR]   **:**

- o   [Next >]   .

[General]   **:**

- o   Beans.BonusBean   .
- o   Beans.BonusHome   .
- o   Beans.Bonus   .
- o   Beans.BonusBean   .
- o   [Entity]   .
- o   [Next >]   .

[Entity Settings]   **:**

- o   [Container-Managed persistence](   )   .
- o   bonus   socsec   .
- o   java.lang.String   .

.

Primitive Type (ex: int, float, char   )

.

o   **primary key field**   socsec   .
o   [Next >]   .

[Environment Entries]   :

o   [Next >]   .   Entity Bean   (   )
.

[Enterprise Bean References]   :

o   [Next >]   .   Entity Bean   Bean
.

[Resource References]   :
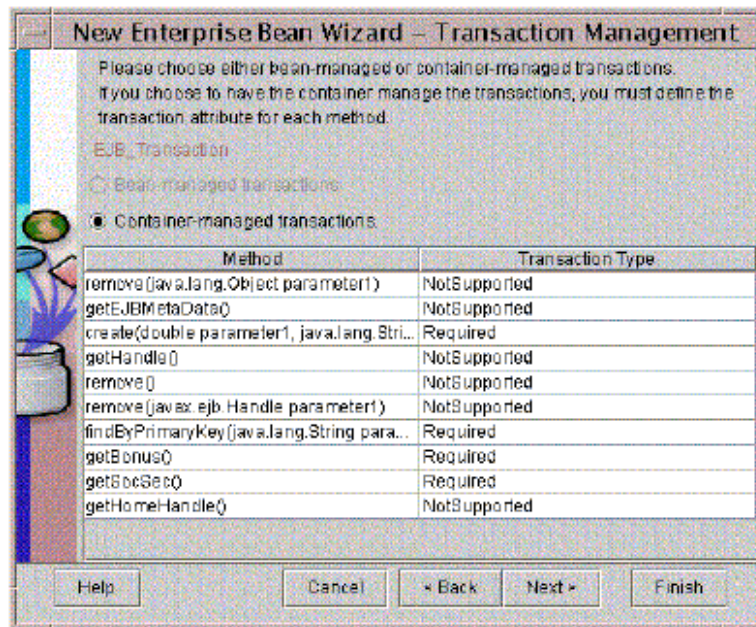
o   [Next >]   .   Entity Bean   ,   JavaMail Session
.

[Security]   :

o   [Next >]   .   Entity Bean   **security roles**
.

[Transaction Management]   :

o   [Container-managed transactions]   .
o   create, findByPrimaryKey, getBonus, getSocSec
required   .

.

Commit   .   ,[Enterprise Java B
eans Developer's Guide]   6   . (java.sun.com/j2ee/j2sdkee/te
chdocs/guides/ejb/html/DevGuideTOC.html)

2

- o   [Next >]   .
- o   [Finish]   .

Local Applications:

- o   BonusApp   .
- o   [JNDI names]   .
- o   BonusBean   JNDI   bonus   .
- o   .
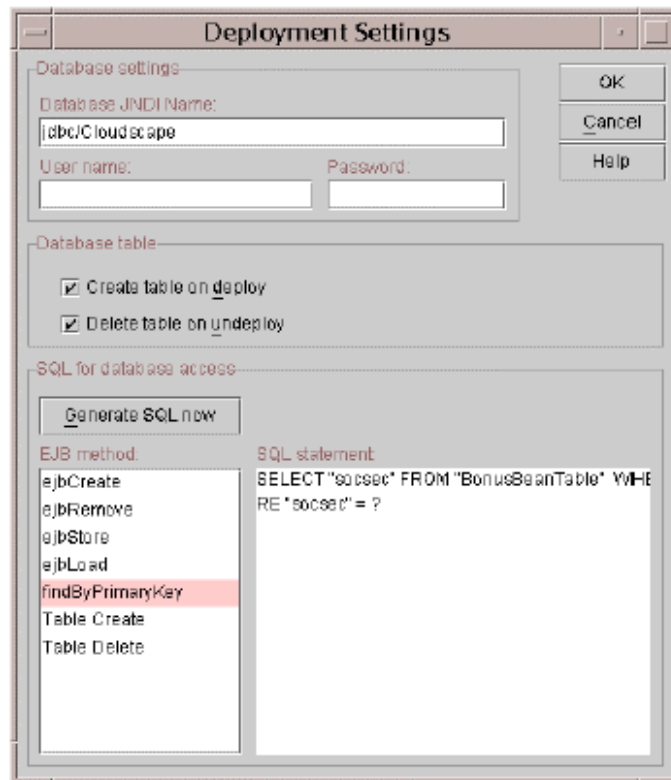
J2EE   , Entity Bean
    SQL   .
   .

Local Applications   :

- o   BonusBean   .

Inspecting window   :

- o   [Entity]   .
- o   [Deployment Settings]   .

[Deployment Settings]   **:**

o   [Database JNDI name]   jdbc/Cloudscape   .

o   .

o   [Create table on deploy]   [Delete table on Undeploy]

          .

o   [Generate SQL now]   .

          :   ,   .

          3. SQL

o   SQL   EJB   findByPrimary
    Key   .   , SQL   . SEL
    ECT "socsec" FROM "BonusBeanTable" WHERE "socsec"=?
              .   (?)   findByPrimaryKey
              .

o   [OK]   .

J2EE   Verify   Deploy

Verify:

- o BonusEar   [Tool]   [Verifier]   .
- o   [OK]   .

                        .

            , Verifier   .

     : Version 1.2   , tests app.WebURI

            ,   .

        :

- o [Tool]   [Deploy Application]   .

    : [Return Client Jar]   ,   .
              ,
                        .   HT
ML   ,   .
                    JAR
  .

- o [Next >]   . CalcBean   [JNDI names]   calcs   BonusBean   [JND
  I names]   bonus   .
               .
- o [Next >]   .   BonusRoot
          .   .
- o [Next >]   .
- o [Finish]   ,   .
- o   [OK]   .

J2EE

Web   ,   8000   . bonus.html
http://localhost:8000/BonusRoot/bonus.html   . social   security   number
multiplier   Submit   . BonusServlet
      ,   HTML   .

Bonus Calculation
Soc Sec passed in: 777777777
Multiplier passed in: 25
Bonus Amount calculated: 2500.0
Soc Sec retrieved: 7777777777
Bonus Amount retrieved: 2500.0

bonus.html   , social security number   multiplier   2   ,
                                 .

Bonus Calculation
Soc Sec passed in: 777777777
Multiplier passed in: 2
Bonus Amount calculated: 200.0
Duplicate primary key.