

```
1 // Class HourlyWorkerDM.java
2 // Copyright 2001 ASR Strategic Resources
3 // AITP NCC 2001 Java Competition Event
4 // Java Contest Solution
5 // Design: Don Baldwin & Craig Slinkman PhD
6 // Code: Eugene Wasserman & Don Baldwin
7
8 package dm;
9 import hr.HourlyWorker;
10
11 import java.sql.SQLException;
12 import java.sql.ResultSet;
13 import java.util.Vector;
14
15 /**
16  * Data management class to deal with HourlyWorker objects.
17  */
18 public class HourlyWorkerDM extends EmployeeDM {
19
20
21     /**
22      * Create a new hourlyWorker object in the database
23      */
24     public void save(HourlyWorker hourlyWorker) throws SQLException {
25         String sql = "insert into Employee" +
26             " (id, FirstName, LastName) values" +
27             "(" + hourlyWorker.getId() + ", '" + hourlyWorker.getFirstName() +
28             "', '" + hourlyWorker.getLastName() + "'" +
29             ")";
30         getStatement().executeUpdate(sql);
31
32         sql = "insert into HourlyWorker" +
33             " (id, wage, hours) values" +
34             "(" + hourlyWorker.getId() + ", " + hourlyWorker.getWage() +
35             ", " + hourlyWorker.getHours() +
36             ")";
37         getStatement().executeUpdate(sql);
38
39     }
40
41     /**
42      * Save an existing HourlyWorker object's info in the database.
43      */
44     public void update(HourlyWorker hourlyWorker) throws SQLException {
45         getStatement().executeUpdate("update Employee set " +
46             " FirstName='" + hourlyWorker.getFirstName() + "', " +
```

```
47     " LastName='" + hourlyWorker.getLastName() + "'"+
48     " where id=" + hourlyWorker.getId());
49
50     getStatement().executeUpdate("update HourlyWorker set " +
51     " wage=" + hourlyWorker.getWage() + ", " +
52     " hours=" + hourlyWorker.getHours() +
53     " where id=" + hourlyWorker.getId());
54 }
55
56 /**
57  * Read the HourlyWorker object with the passed id from the database.
58  */
59 public HourlyWorker read(int id) throws SQLException {
60
61     ResultSet rs = getStatement().executeQuery(
62     " select e.id, FirstName, LastName, wage, hours from Employee e, HourlyWorker h" +
63     " where (e.id = h.id) and e.id=" + id +
64     " order by LastName, FirstName"
65     );
66     if(!rs.next())
67         return null;
68     HourlyWorker hourlyWorker = new HourlyWorker(rs.getInt("id"), rs.getString("FirstName"),
69     rs.getString("LastName"), rs.getDouble("wage"), rs.getDouble("hours"));
70
71     return hourlyWorker;
72 }
73
74 /**
75  * Delete a new HourlyWorker object in the database
76  */
77 public void delete(int id) throws SQLException {
78     getStatement().executeUpdate("delete from Employee where id=" + id);
79     getStatement().executeUpdate("delete from HourlyWorker where id=" + id);
80 }
81
82 /**
83  * List the HourlyWorker objects in the database
84  */
85 public HourlyWorker[] list() {
86     HourlyWorker list[] = null;
87     Vector listVec = new Vector();
88     try {
89         ResultSet rs = getStatement().executeQuery(
90         " select * from Employee, HourlyWorker" +
91         " where (Employee.id = HourlyWorker.id)" +
92         " order by LastName, FirstName"
```

```
93         );
94         while(rs.next()) {
95             listVec.addElement(new HourlyWorker(rs.getInt("id"), rs.getString("FirstName"),
96             rs.getString("LastName"), rs.getDouble("wage"), rs.getDouble("hours")));
97         }
98     } catch (SQLException sqle) { sqle.printStackTrace();}
99     list = new HourlyWorker[listVec.size()];
100     Object listObj[] = listVec.toArray();
101     for(int i=0; i<listObj.length; i++)
102         list[i] = (HourlyWorker)listObj[i];
103     return list;
104 }
105
106
107 } // END Class HourlyWorkerDM.java
```