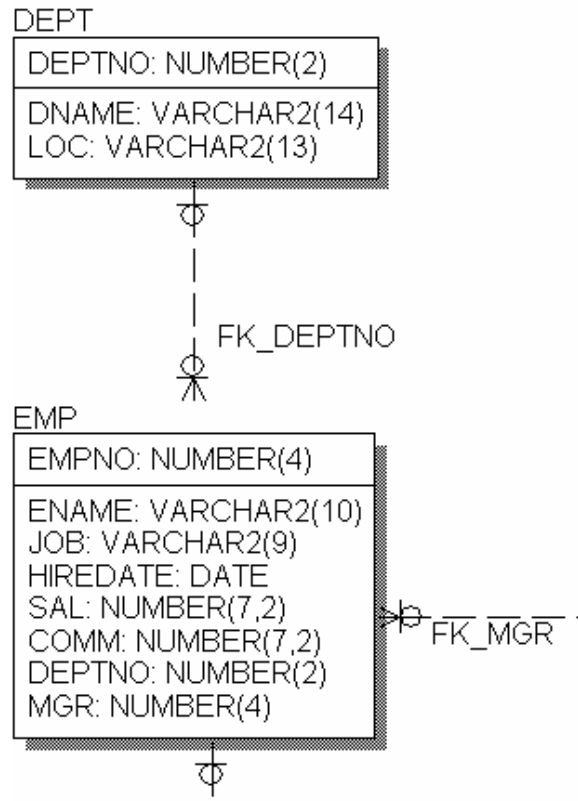


Introduction to SQLTags: Lab 4—Building JSP Pages

Goal: To create and modify simple JSP pages that use SQLTags generated tag libraries based on the following data model.

Sample Data model

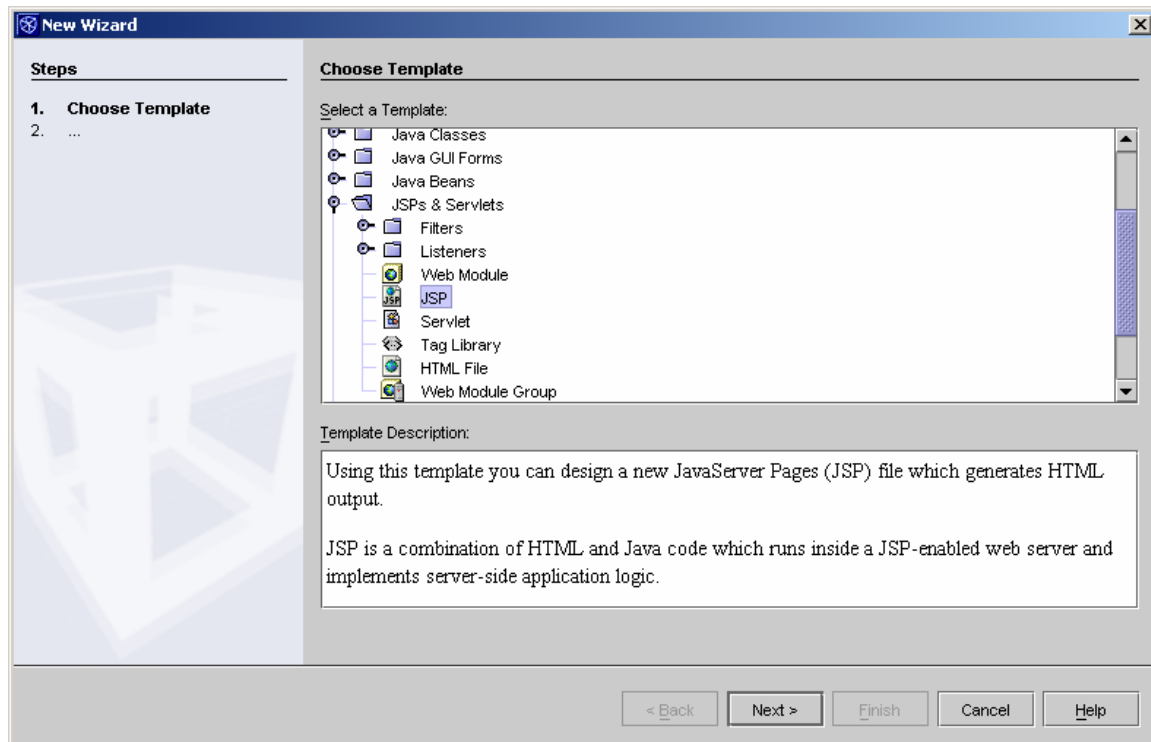


Launch NetBeans IDE and open the “sqltags-1-0-6” project to begin the exercises.

Simple Tag

Goal: To create a simple JSP page that queries the database using a generated tag and to become familiar with the NetBeans IDE.

To create a new JSP file, right-click on the `sqltags-1-0-6` directory and select **"new->All Templates"**. Expand the **"Jsp & Servlets"** item and select **"JSP"** and click on **"Next."** Give it the file the name, `"class1"` (do not add the `".jsp"` suffix), and click **"Finish."**



Delete the two lines that contain the `"jsp:useBean"` and the `"jsp:getProperty"` tags and compile the file by hitting F9.

Starting page: class1.jsp. Verify that the page compiles and executes. Should result in an empty browser page if successful.

```
<%@page contentType="text/html"%>
<html>
<head><title>JSP Page</title></head>
<body>

</body>
</html>
```

Next, add the taglib reference and the “connection”, and “emp” tags and execute the page. This page should result in a simple listing of all rows in the EMP table.

```
<%@ taglib uri="demoTags.jar" prefix="x" %>
<html>
<head><title>SQLTags Page</title></head>
<body>    <h2>SQLTags Page</h2>
<x:connection id="con">

    <x:emp id="e" where="order by ENAME">
        <%= e.getENAME() %>
        <br>
    </x:emp>

</x:connection>
</body>
</html>
```

Add jstl core taglib reference and the “out” tag to demonstrate an alternate syntax for JSP page output. Same page, different syntax.

```
<%@ taglib uri="demoTags.jar" prefix="x" %>
<%@ taglib uri="http://java.sun.com/jstl/core" prefix="c" %>
<html>
<head><title>SQLTags Page</title></head>
<body>    <h2>SQLTags Page</h2>
<x:connection id="con">

    <x:emp id="e" where="order by ENAME">
        <%= e.getENAME() %>
        <c:out value="{e.ENAME}"/>
        <br>
    </x:emp>

</x:connection>
</body>
</html>
```

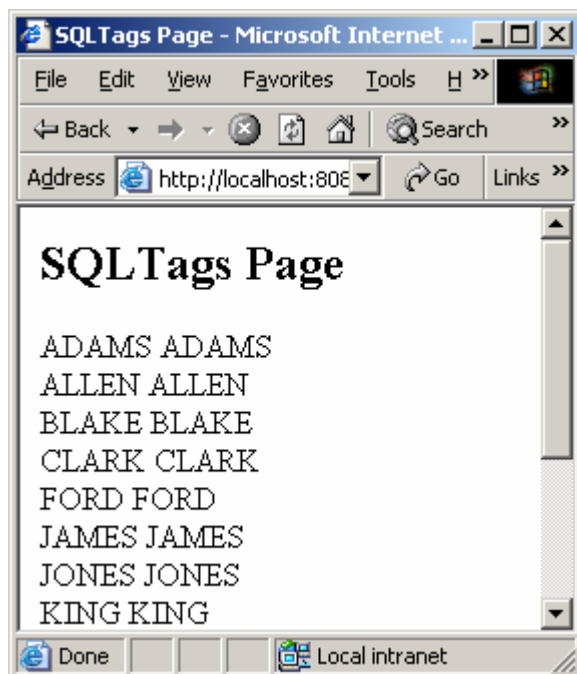
Add note on JSP 2.0 syntax: index4.jsp. JSP 2.0 requires Servlet 2.4 available in Tomcat 5.0 (not 4.0 that comes with NetBeans).

```
<%@ taglib uri="demoTags.jar" prefix="x" %>
<%@ taglib uri="http://java.sun.com/jstl/core" prefix="c" %>
<html>
<head><title>SQLTags Page</title></head>
<body>  <h2>SQLTags Page</h2>
<x:connection id="con">

    <x:emp id="e" where="order by ENAME">
        <%= e.getENAME() %>
        <c:out value="{e.ENAME}"/>
        <!-- ${e.ENAME} // JSP 2.0 syntax (no tags required!) --%>
        <br>
    </x:emp>

</x:connection>
</body>
</html>
```

The output page should look something like this:



Simple Tag plus Parent lookup

Goal: To add a parent key lookup value to the previous page. Please refer to the data model on the first page for foreign key names used in this example.

Parent table lookup demonstrates accessing parent table via foreign key name.

```
<%@ taglib uri="demoTags.jar" prefix="x" %>
<html>
<head><title>SQLTags Page</title></head>
<body>  <h2>SQLTags Page</h2>
<x:connection id="con">

  <x:emp id="e" where="order by ENAME">
    <%= e.getENAME() %> assigned to <%= e.getFK_DEPTNO().getDNAME() %>
    <br>
  </x:emp>

</x:connection>
</body>
</html>
```

Parent table lookup using jstl tag for output.

```
<%@ taglib uri="demoTags.jar" prefix="x" %>
<%@ taglib uri="http://java.sun.com/jstl/core" prefix="c" %>
<html>
<head><title>SQLTags Page</title></head>
<body>  <h2>SQLTags Page</h2>
<x:connection id="con">

  <x:emp id="e" where="order by ENAME">
    <c:out value="{e.ENAME}"/> assigned to
    <c:out value="{e.FK_DEPTNO.DNAME}"/>
    <br>
  </x:emp>

</x:connection>
</body>
</html>
```

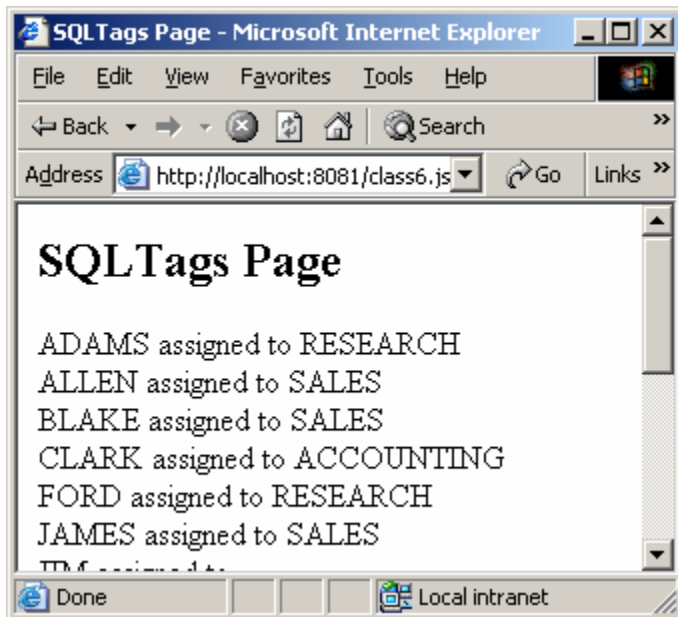
Parent table lookup using JSP 2.0 syntax for output, nice!

```
<%@ taglib uri="demoTags.jar" prefix="x" %>
<html>
<head><title>SQLTags Page</title></head>
<body>  <h2>SQLTags Page</h2>
<x:connection id="con">

  <x:emp id="e" where="order by ENAME">
    ${e.ENAME} assigned to ${e.FK_DEPTNO.DNAME}
    <br>
  </x:emp>

</x:connection>
</body>
</html>
```

The output page should look something like this:



Parent-child Nesting

Goal: To demonstrate parent-child nesting within a JSP page.

Parent-child nesting based on foreign key name. Create a new file named, class3.jsp and enter the following text.

```
<%@ taglib uri="demoTags.jar" prefix="sqltags" %>
<html>
<head><title>JSP Page</title></head>
<body>
<sqltags:connection id="connect" >

    <sqltags:dept id="d" where="order by DNAME">
        <%= d.getDNAME() %>
        <BLOCKQUOTE>
            <sqltags:emp id="e" foreignKey="FK_DEPTNO" parentName="d">
                <%= e.getENAME() %>
                <br>
            </sqltags:emp>
        </BLOCKQUOTE>
    </sqltags:dept>

</sqltags:connection>
</body>
</html>
```

Parent-child nesting based on foreign key name, alternate syntax using jstl.

```
<%@ taglib uri="demoTags.jar" prefix="sqltags" %>
<%@ taglib uri="http://java.sun.com/jstl/core" prefix="c" %>
```

```

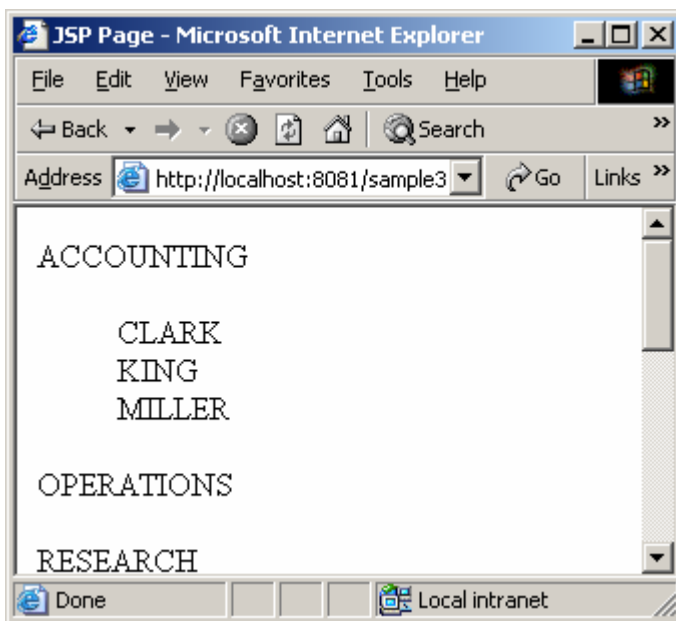
<html>
<head><title>JSP Page</title></head>
<body>
<sqltags:connection id="connect" >

    <sqltags:dept id="d" where="order by DNAME">
        <c:out value="\${d.DNAME}" />
        <BLOCKQUOTE>
            <sqltags:emp id="e" foreignKey="FK_DEPTNO" parentName="d">
                <c:out value="\${e.ENAME}" />
                <br>
            </sqltags:emp>
        </BLOCKQUOTE>
    </sqltags:dept>

</sqltags:connection>
</body>
</html>

```

The output page should look something like this:



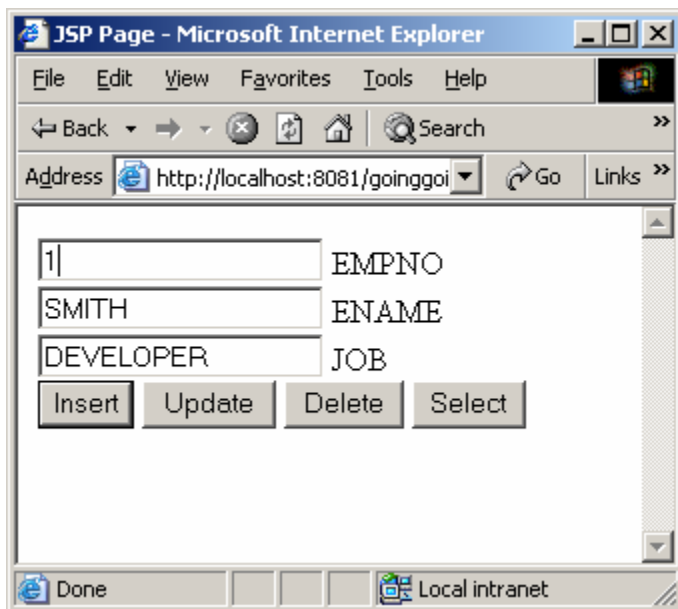
Insert/Update/Delete Page

Goal: Demonstrate the DML processing to support insert, update, and delete operations within a JSP page.

Create a new JSP page named, class4.jsp and enter the following text:

```
<%@page contentType="text/html"%>
<%@taglib prefix="x" uri="demoTags.jar" %>
<html>
<head><title>JSP Page</title></head>
<body>
<x:connection id="con">
  <x:emp id="e" properties="true" buttonName="f" >
    <x:exception name="e"/>
    <form method="post">
      <input name='EMPNO' value="<%= e.getEMPNO() %>" > EMPNO <br>
      <input name='ENAME' value="<%= e.getENAME() %>" > ENAME <br>
      <input name='JOB' value="<%= e.getJOB() %>" > JOB <br>
      <input type=submit name="f" value="Insert">
      <input type=submit name="f" value="Update">
      <input type=submit name="f" value="Delete">
      <input type=submit name="f" value="Select">
    </form>
  </x:emp>
</x:connection>
</body>
</html>
```

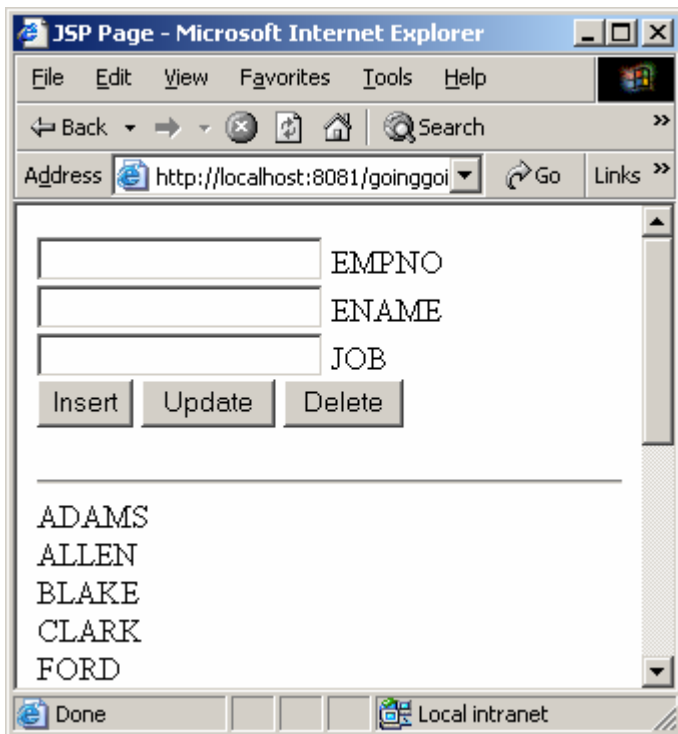
Compile and execute the new page and enter a new EMP row into the database as indicated and clicking on the “Insert” button. Once the row has been added to the database, it can be “Updated” or “Deleted” by clicking on the corresponding button.



Now, we will add an EMP list to the end of the page to help in finding rows in the table.

```
<%@page contentType="text/html"%>
<%@taglib prefix="x" uri="demoTags.jar" %>
<html>
<head><title>JSP Page</title></head>
<body>
<x:connection id="con">
  <x:emp id="e" properties="true" buttonName="f" >
    <x:exception name="e"/> <!-- report errors --%>
    <form method="post">
      <input name='EMPNO' value="<%= e.getEMPNO() %>" > EMPNO <br>
      <input name='ENAME' value="<%= e.getENAME() %>" > ENAME <br>
      <input name='JOB' value="<%= e.getJOB() %>" > JOB <br>
      <input type=submit name="f" value="Insert">
      <input type=submit name="f" value="Update">
      <input type=submit name="f" value="Delete">
    </form>
  </x:emp>
  <hr>
  <x:emp id="e2" where='order by ENAME'>
    <%= e2.getENAME() %>
    <br>
  </x:emp>
</x:connection>
</body>
</html>
```

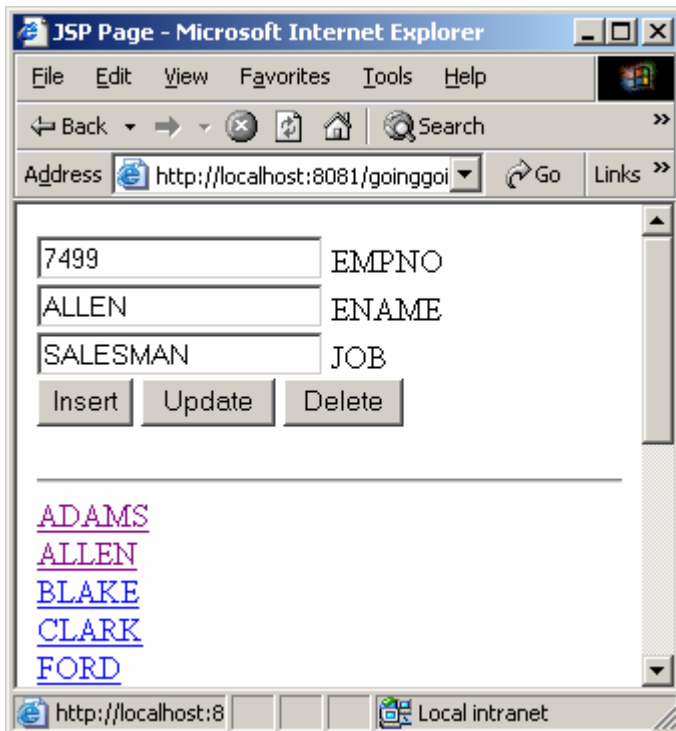
Now, execute the page and review the output in the browser.



In this last step we add an HREF to each item in the list to provide a link to the “Update” form for each employee in the database.

```
<%@page contentType="text/html"%>
<%@taglib prefix="x" uri="demoTags.jar" %>
<html>
<head><title>JSP Page</title></head>
<body>
<x:connection id="con">
  <x:emp id="e" properties="true" buttonName="f" >
    <x:exception name="e"/> <!-- report errors -->
    <form method="post" action="class4.jsp">
      <input name='EMPNO' value="<%= e.getEMPNO() %>" > EMPNO <br>
      <input name='ENAME' value="<%= e.getENAME() %>" > ENAME <br>
      <input name='JOB' value="<%= e.getJOB() %>" > JOB <br>
      <input type=submit name="f" value="Insert">
      <input type=submit name="f" value="Update">
      <input type=submit name="f" value="Delete">
    </form>
  </x:emp>
  <hr>
  <x:emp id="e2" where='order by ENAME'>
    <A HREF="./class4.jsp?EMPNO=<%=e2.getEMPNO()%>&f=select">
      <%= e2.getENAME() %></A>
    <br>
  </x:emp>
</x:connection>
</body>
</html>
```

In this page, each EMP entry can be clicked on to load the row in the “Update” page.



Demonstrations of Sample Code

Goal: Review the sample code included in the sqltags-1-0-6 web application.

Instructor led review of samples. Review paging and Array processing. No formal lab, but will discuss with those who are interested.

Done. This completes the lab!