Data Display Lab

Overview: In this lab, you will learn about how to write java/JSP code that accesses your database. You will add three pages to your web application: a page that displays all your users, a page that displays all the records from your "other" table, and a page that displays all the records from your associative table.

Word Document Requirements:

After reading the "JSP Data Display Tutorial" and downloading/studying the associated sample code, generate the following error messages from the "using classes" java/JSP code:

- Missing Database Driver
- Database Unreachable
- Database Not Authorized
- Syntax error in Sql Statement
- Error Extracting Data from Result Set (bad column name)
- Error Extracting Data from Result Set (wrong data type)

For each of the above types of errors, add the following information into your Word or wordpad document:

1. The type of error (as listed above).
2. Copy/paste the error message you got (either from the JSP page or from the GlassFish log).
3. Describe how/where you would normally expect to fix the problem.

Lab Requirements:

- To facilitate grading, your three new pages shall be named users.jsp, assoc.jsp, and other.jsp, but the links in your nav bar shall be named whatever is appropriate for your web application (aligning with the names of the tables in your database).
- other.jsp:
  - This page shall show all the records and all of the fields of your "other" table.
  - The columns shall be presented in an order that makes the page useful and appealing.
  - Records shall be ordered by whatever column you chose to put first (and possibly have a secondary sort on the second column, if appropriate).
- users.jsp:
  - This page shall show all records of your user table joined with your user_role table and all the fields (except user_role_id).
  - The columns shall be presented in a useful and appealing order. Combine fields where appropriate.
  - Records shall be ordered by the 1st and possibly 2nd column.
- assoc.jsp:
  - This page shall show all records of your associative table joined with your user table and your "other" table.
  - This page shall show all fields of your associative table (except its PK and FK fields) and at least one non-key field of your user and at least one non-key field of your other table. The columns shall be presented in a useful and appealing order (sorted by 1st and possibly 2nd column). Combine fields where appropriate.
- All three of the "data display" JSP pages shall contain an HTML table to align the data.
  - Each column in the HTML table shall provide a column heading in a <th> tag.
  - Any column that displays a number (by itself) shall be right justified (the header and the data).
  - Any column that displays a date (by itself) shall be centered (the header and the data).

- Combine fields into one cell, where appropriate. For example, you might have a single column labeled "Address" that has [[street address, then a new line, then city, state, and zip on a second line]].
  - If any of your tables have URL fields, incorporate these into links. For example, suppose your table has one column that holds a teamName and another column that holds a teamWebSiteURL, the HTML that you create for one of the columns might look like this:

    <td> <a href='[teamWebSiteURL]'> [teamName] </a> </td>
  - If any of your tables have links to images, create HTML like this:

    <td> <img src= '[imageURL]'/> </td>
- Tables in your database were required to have certain null-able non-character fields (like date, int, decimal). Each table was to have at least one record with all values populated and at least one record where all null-able columns have null for their value. Your code has to be able to gracefully handle all of these data types, including null values. If you run your result set columns through the FormatUtils methods, things should be OK.
- Make sure to always have "user friendly" db error messages. For now, this just means "Database unavailable - please try later or contact your administrator. Database error message is: ..." It is important to show the actual technical error message (after the user friendly one) so that people can resolve problems experienced when code is in production.
- Each of the "data display" JSP pages shall have the same look and feel as your home page. OR (if your home page layout has very little room for text), your site can have two similar layouts – one for your home page and a second layout (with a large content area) for the rest of your pages (e.g., the data display pages, the labs page).
- Your labs page shall have a blog at the top that describes the work you did this week. This blog shall link to all the "data display" JSP pages that you create this week.
- All JSP pages in your web app (including new pages from this lab as well as pages from previous labs) shall:
  - have JSP include statements (for HTML code reuse)
  - contain nothing but well formed HTML and CSS (View Source from firefox – no red syntax errors).
- All the links in your web application shall work properly.

Design specifications:

- All data display JSP pages shall use the "WITH CLASSES" approach as demonstrated in the sample code.
  - Your web app shall have a dbUtils package (with classes DbConn and FormatUtils)
  - Your web app shall have a view package (with classes that are named according to the names of your database tables, e.g., if you have a table named product, your class should be named ProductView).
- All code shall use good programming style as specified in the class labs page under the section entitled "Requirements for All Labs AND Project".

Homework submission:

- Publish and test your web application.
- Attach (into the blackboard assignment), the document described above, along with a zip file of your complete web application project folder.

Suggested Approach:

- Specified in the labs page (where you found this document).