

Lab 2 – Scripting Elements

Import the Lab02_ScriptingElements_STARTER file and rename it to Lab02_ScriptingElements

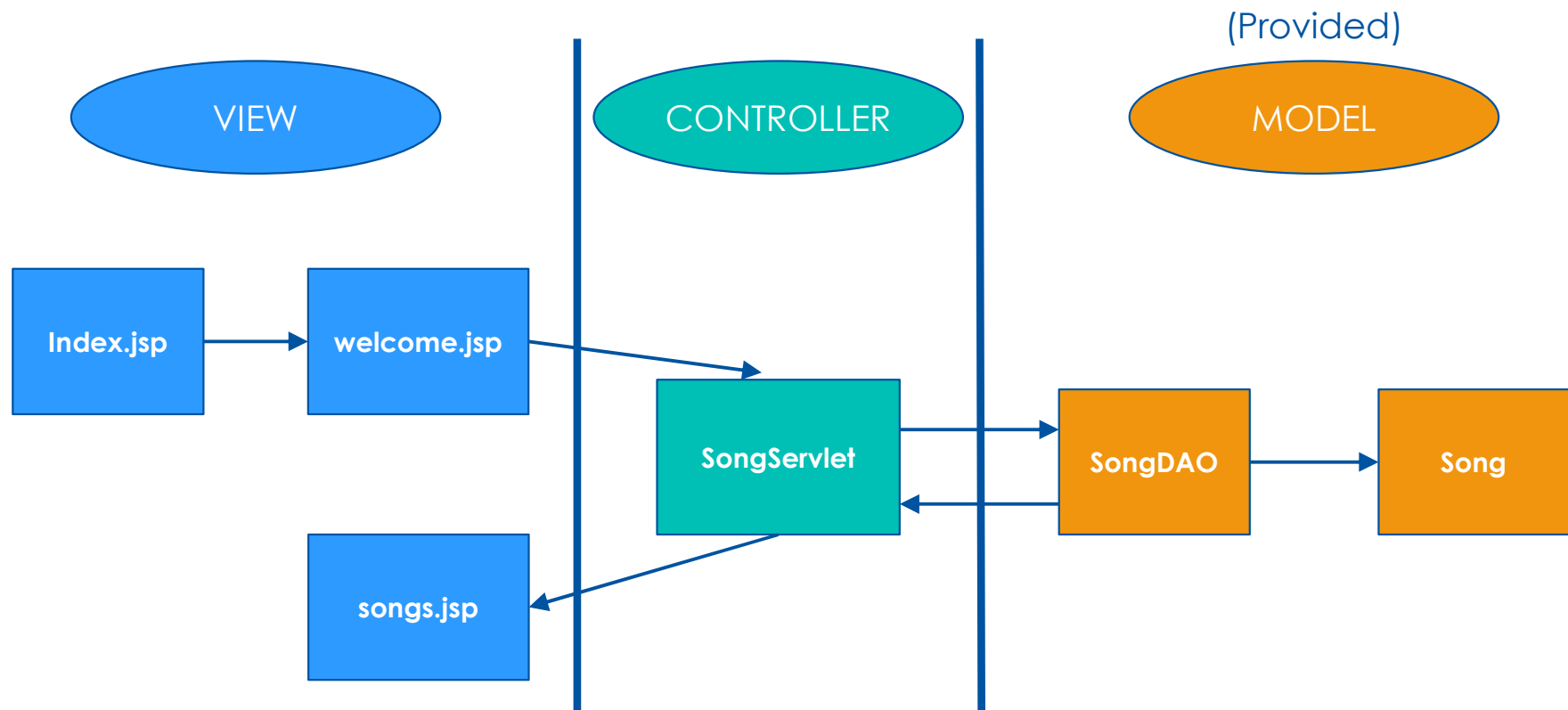
- This is a maven project with the Tomcat dependency already added

- Examine the Song and SongDAO files in the com.music.model package
 - Song is a simple POJO (Plain Old Java Object)
 - SongDAO contains our fake data and access methods

A partially complete songs.jsp file is provided which we'll complete later

Lab 2 – Scripting Elements

Application Overview



Lab 2 – Scripting Elements

- Right click on WebContent > New > JSP File
- Create an index.jsp page and add the code below

```
...  
<title>Welcome</title>  
</head>  
<body>  
    <h3>Welcome to Simple Song Service</h3>  
    <br>  
    <hr>  
    <h3>Please enter your user name</h3>  
    <form action="welcome.jsp">  
        <label for="username">User Name:</label>  
        <input type="text" name="username"> <br/><br/>  
        <input type="submit" value="Next">  
    </form>  
</body>
```

Lab 2 – Scripting Elements

- Next, create a welcome.jsp file
 - Set the title to “Simple Song Service”
 - Add an Expression to get the user name from the request and welcome them
 - Add a link to invoke the get method of SongServlet (which we’ll write next)

```
...  
<title>Simple Song Site</title>  
</head>  
<body>  
  <h3>Welcome to Simple Song Service</h3><br><hr>  
  <p><%= "Welcome " + request.getParameter("username") %></p>  
  <a href="SongServlet">Show All Songs</a>  
</body>
```

Lab 2 – Scripting Elements

- You should be able to run the application on the server and see the following output
 - Note the “Show All Songs” link doesn’t work yet

index.jsp



A screenshot of a web browser displaying the 'index.jsp' page. The address bar shows 'localhost:8081/Lab02_ScriptingElements/index.jsp'. The page content includes a heading 'Welcome to Simple Song Service', a horizontal line, and a prompt 'Please enter your user name'. Below this is a form with the label 'User Name:' followed by a text input field. At the bottom left of the form is a button labeled 'Next'.

welcome.jsp



A screenshot of a web browser displaying the 'welcome.jsp' page. The address bar shows 'localhost:8081/Lab02_ScriptingElements/welcome.jsp?username=Bardi+C'. The page content includes a heading 'Welcome to Simple Song Service', a horizontal line, and the text 'Welcome Bardi C'. Below this is a blue, underlined link labeled 'Show All Songs'.

Lab 2 – Scripting Elements

- Create a new Servlet to serve as our controller
 - Name it SongServlet
 - Put it in the `com.music.controller` package
 - Override the `doGet` method to do the following:
 - Create a new `SongDAO` and call the `findAllSongs` method
 - Set a session attribute with the key “Songs”. The value will be the list returned from `findAllSongs`
 - Forward the request to `songs.jsp` (which we’ll write next)

```
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    List<Song> allSongs = new SongDAO().findAllSongs();
    HttpSession session = request.getSession();
    session.setAttribute("Songs", allSongs);
    request.getRequestDispatcher("/songs.jsp").forward(request,response);
}
```

Lab 2 – Scripting Elements

- Open the provided songs.jsp file
- Add a Scriptlet (think `<% %>`), that does the following:
 - Gets the list of songs from session attribute set in our SongServlet doGet method
 - Loop through the list and print out each song (including id, title, and artist)

```
<%  
    List<Song> songs = (List<Song>)session.getAttribute("Songs");  
    for(Song _song : songs){  
        out.print(_song);  
        out.print("<br/>");  
    }  
%>
```

Lab 2 – Scripting Elements

Run the application. Your output should look something like this



```
id=101 title=Baby Love artist=The Supremes  
id=102 title=Pancho and Lefty artist=Townes Van Zandt  
id=103 title=Truth Hurts artist=Lizzo  
id=104 title=Take It Easy artist=The Eagles  
id=105 title=Your're So Vain artist=Carly Simon
```


Lab 2 – Scripting Elements

Let's add a declaration (`<%! %>`) and one more expression (`<%= %>`)

- Work in your welcome.jsp file
- Declare a private variable for storing a discount code and a private method to get the discount code
- Add a message and an expression that calls the get discount code method

```
...  
<%!private String discountCode = "VIPCUST10";  
private String getDiscountCode() {  
    return discountCode;  
}%>  
<div>  
    Use this code to get an additional 10% discount:  
    <h4><%=getDiscountCode()%></h4>  
</div>  
<a href="SongServlet">Show All Songs</a>
```

Lab 2 – Scripting Elements

Run the application again, starting at index.jsp

This time, your welcome.jsp should look like the solution below

