



JSP

CSCI 201

Principles of Software Development

Jeffrey Miller, Ph.D.
jeffrey.miller@usc.edu



Outline

- JSP
- Program

JSP 3-Tier Architecture

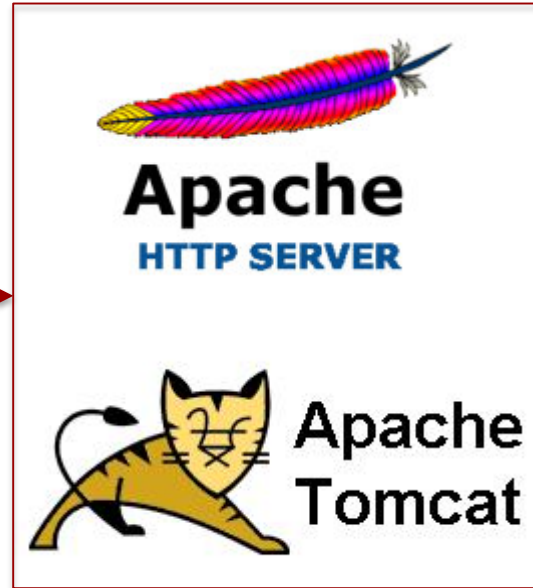


Client



Server

Web/Application Server



Database





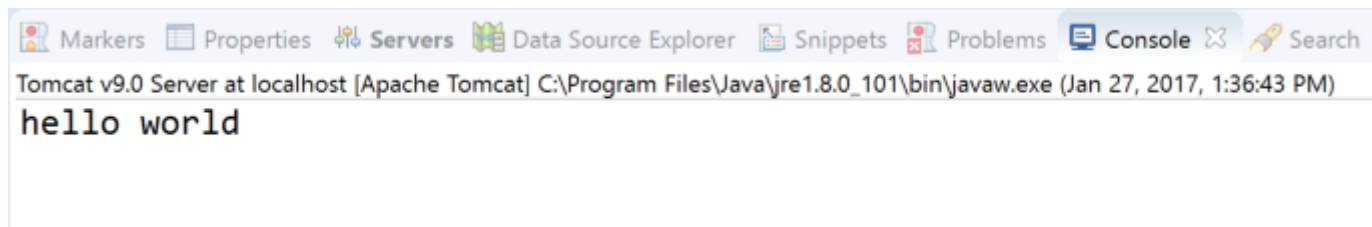
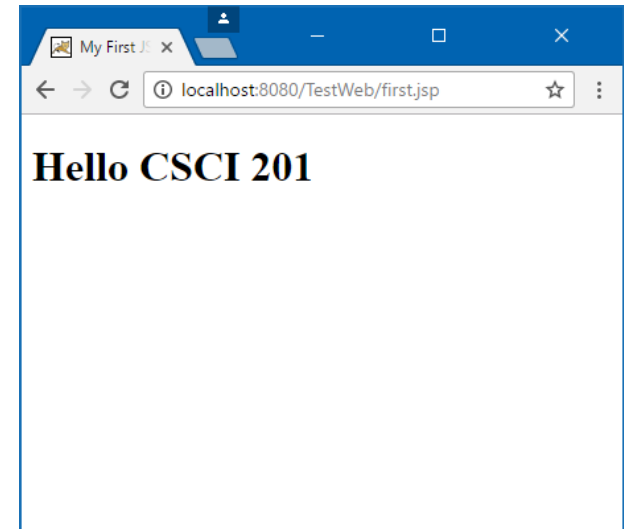
- Because embedding HTML, CSS, and JavaScript as the output of a servlet is quite tedious, Java has a way to instead embed Java in your HTML file
 - › We can do this through a Java Server Page (JSP)
- JSPs get converted to servlets the first time they are accessed, compiled, and made available dynamically by the application server
- A JSP will be a file that looks like client-side code with embedded Java within special tags
 - › Declaration `<% ! %>`
 - › Directive `<%@ %>`
 - › Expression `<%= %>`
 - › Scriptlet `<% %>`
 - › JSTL – explained in a later slide



My First JSP



```
1  <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2      pageEncoding="ISO-8859-1"%>
3  <!DOCTYPE html>
4  <html>
5      <head>
6          <title>My First JSP</title>
7      </head>
8      <body>
9          <h1>Hello CSCI 201</h1>
10         <% System.out.println("hello world"); %>
11     </body>
12 </html>
```



Generated Servlet (partial service method)



```
1  try {
2      response.setContentType("text/html; charset=ISO-8859-1");
3      pageContext = _jspxFactory.getPageContext(this, request, response, null,
4                                                  true, 8192, true);
5      _jspx_page_context = pageContext;
6      application = pageContext.getServletContext();
7      config = pageContext.getServletConfig();
8      session = pageContext.getSession();
9      out = pageContext.getOut();
10     _jspx_out = out;
11     out.write("\r\n");
12     out.write("<!DOCTYPE html>\r\n");
13     out.write("<html>\r\n");
14     out.write("    <head>\r\n");
15     out.write("        <title>My First JSP</title>\r\n");
16     out.write("    </head>\r\n");
17     out.write("    <body>\r\n");
18     out.write("        <h1>Hello CSCI 201</h1>\r\n");
19     out.write("    ");
20     System.out.println("hello world");
21     out.write("\r\n");
22     out.write("    </body>\r\n");
23     out.write("</html>\r\n");
```

Location of compiled JSP

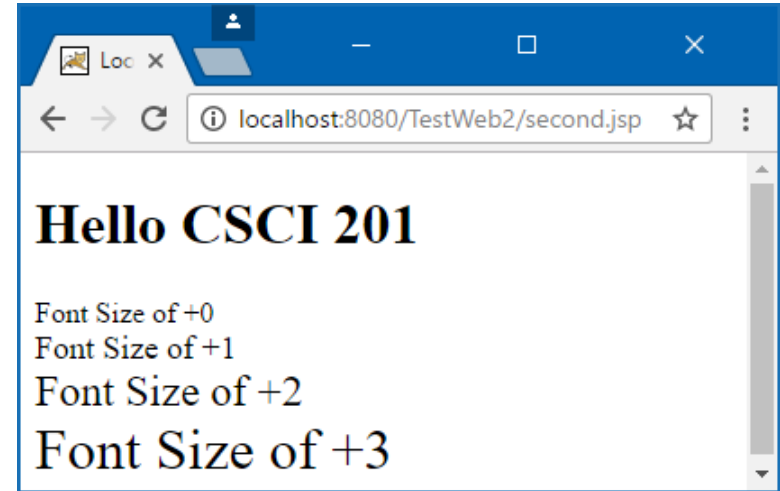
```
C:\Users\jeffadmin\workspace\
.metadata\plugins\
org.eclipse.wst.server.core\tmp1\
work\Catalina\localhost\
TestWeb2\org\apache\jsp
```



My Second JSP



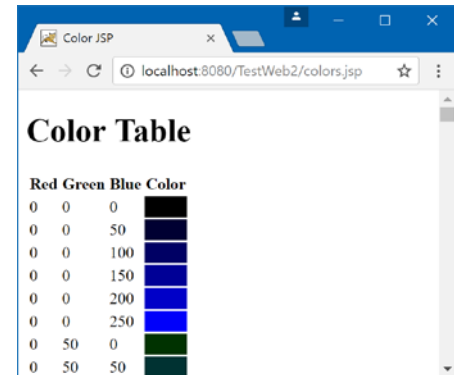
```
1  <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2     pageEncoding="ISO-8859-1"%>
3  <!DOCTYPE html>
4  <html>
5     <head>
6         <title>Loops</title>
7     </head>
8     <body>
9         <h1>Hello CSCI 201</h1>
10        <%
11            for (int i=0; i < 4; i++) {
12                <%
13                    <font size="+<%= i %>">Font Size of +<%= i %></font><br />
14                <%
15                    }
16                <%>
17            </body>
18 </html>
```



Color JSP



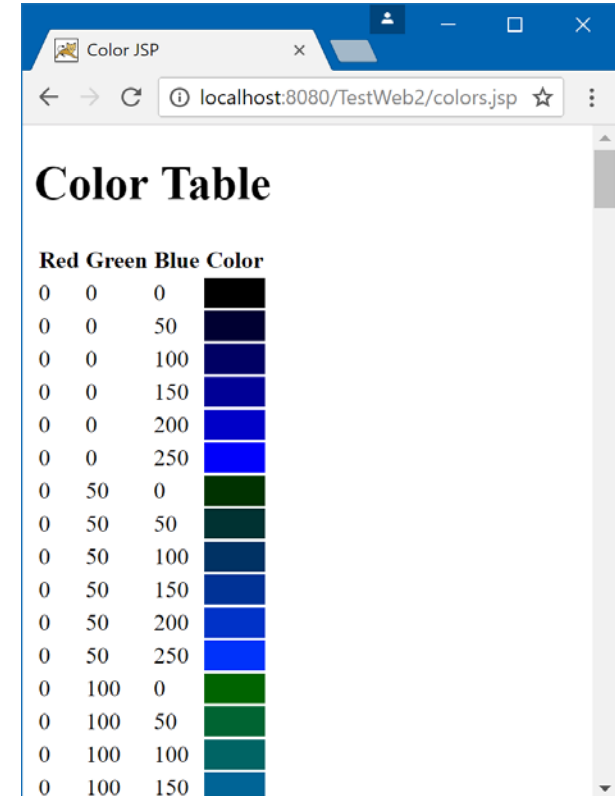
```
1  <%@ page language="java"
2  <!DOCTYPE html>
3  <html>
4    <head>
5      <title>Color JSP</title>
6    <%!
7      String getColor(int r, int g, int b) {
8        String color = "";
9        color += makeHex(r);
10       color += makeHex(g);
11       color += makeHex(b);
12       return color;
13     }
14
15     String makeHex(int c) {
16       String hexString = Integer.toHexString(c);
17       if (hexString.length() == 1) {
18         hexString = "0" + hexString;
19       }
20       return hexString;
21     }
22   %>
23 </head>
24 <body>
25   <h1>Color Table</h1>
26
27   <table>
28     <tr>
29       <th>Red</th>
30       <th>Green</th>
31       <th>Blue</th>
32       <th>Color</th>
33     </tr>
34
35     <%
36       for (int red=0; red < 255; red+=50) {
37         for (int green=0; green < 255; green+=50) {
38           for (int blue=0; blue < 255; blue+=50) {
39
40             <tr>
41               <td><%= red %></td>
42               <td><%= green %></td>
43               <td><%= blue %></td>
44
45               String color = getColor(red, green, blue);
46
47               <td style="background-color:#<%= color %>;" ></td>
48             </tr>
49           }
50         }
51       }
52     %>
53   </table>
54 </body>
55 </html>
```



Color JSP Generated HTML



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Color JSP</title>
5 </head>
6 <body>
7 <h1>Color Table</h1>
8 <table>
9 <tr><th>Red</th><th>Green</th><th>Blue</th><th>Color</th></tr>
10 <tr>
11 <td>0</td>
12 <td>0</td>
13 <td>0</td>
14 <td style="background-color:#000000;"> </td>
15 </tr>
16 <tr>
17 <td>0</td>
18 <td>0</td>
19 <td>50</td>
20 <td style="background-color:#000032;"> </td>
21 </tr>
22 <tr>
23 <td>0</td>
24 <td>0</td>
25 <td>100</td>
26 <td style="background-color:#000064;"> </td>
27 </tr>
...
1743 </table>
1744 </body>
1745 </html>
```



JSPs and HTML Forms



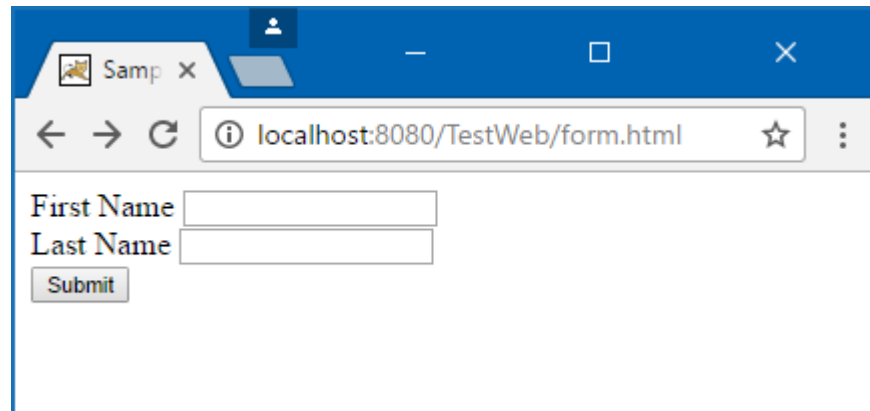
- JSPs (which get converted to servlets) can be used to process the data submitted from an HTML form through the `request` variable
- JSPs have a number of implicit variables
 - › `HttpServletRequest request`
 - › `HttpServletResponse response`
 - › `PrintWriter out`
 - › `HttpSession session`
 - › `ServletContext application`
 - › `ServletConfig config`
 - › `JSPWriter pageContext`
 - › `HttpServletRequest this`
 - › `Exception exception`



JSP Form Example



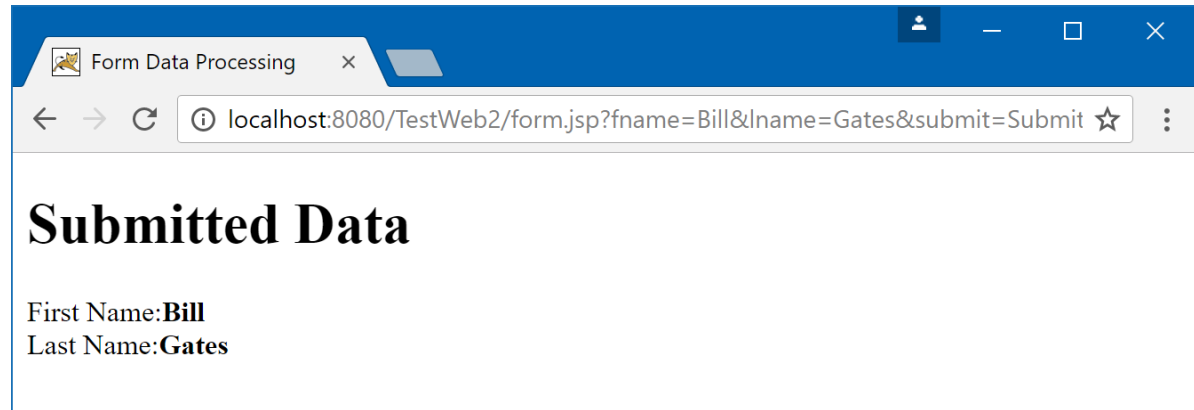
```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Sample Form</title>
5    </head>
6    <body>
7      <form name="myform" method="GET" action="FormServlet">
8        First Name <input type="text" name="fname" /><br />
9        Last Name <input type="text" name="lname" /><br />
10       <input type="submit" name="submit" value="Submit" />
11     </form>
12   </body>
13 </html>
```



JSP Form Example



```
1  <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2     pageEncoding="ISO-8859-1"%>
3  <!DOCTYPE html>
4  <html>
5     <head>
6         <title>Form Data Processing</title>
7     </head>
8     <body>
9     <%
10         String fname = request.getParameter("fname");
11         String lname = request.getParameter("lname");
12     %>
13     <h1>Submitted Data</h1>
14     First Name:<strong><%= fname %></strong><br />
15     Last Name:<strong><%= lname %></strong>
16 </body>
17 </html>
```





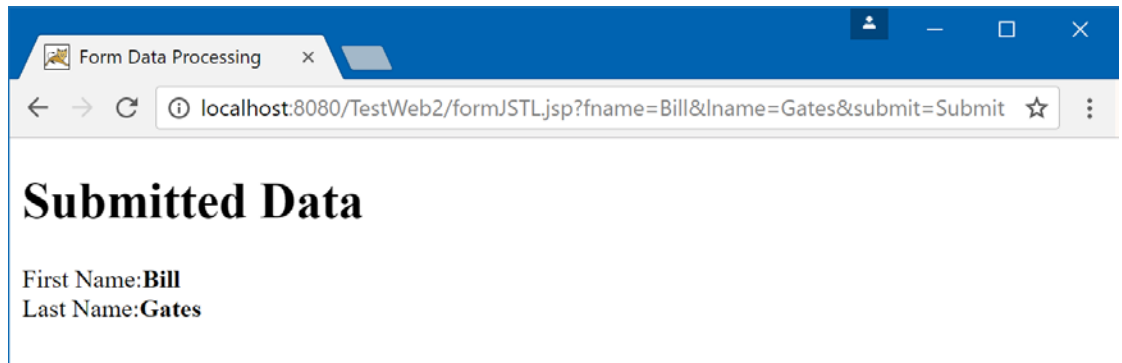
- The Java Server Pages Standard Tag Library (JSTL) is a collection of useful JSP tags that have core functionality for many JSP applications
- The idea with JSTL was to keep front-end programmers from having to learn Java while allowing them to write code with similar functionality
- There are five tag library groups in JSTL
 - › Core tags - `<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>`
 - › Formatting tags - `<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>`
 - › SQL tags - `<%@ taglib prefix="sql" uri="http://java.sun.com/jsp/jstl/sql" %>`
 - › XML tags - `<%@ taglib prefix="x" uri="http://java.sun.com/jsp/jstl/xml" %>`
 - › JSTL functions - `<%@ taglib prefix="fn" uri="http://java.sun.com/jsp/jstl/functions" %>`
- To use JSTL, you must place `jstl-1.2.jar` in the lib directory of your Tomcat installation, then restart Tomcat



JSTL Example



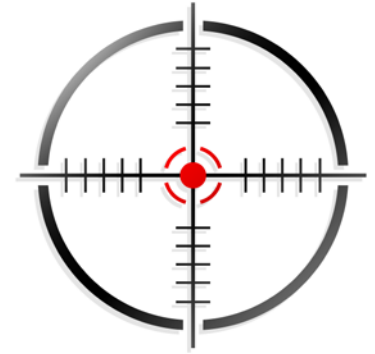
```
1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2     pageEncoding="ISO-8859-1"%>
3 <%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
4 <!DOCTYPE html>
5 <html>
6     <head>
7         <title>Form Data Processing</title>
8     </head>
9     <body>
10        <c:set var="fname" value="${param.fname}" />
11        <c:set var="lname" value="${param.lname}" />
12        <h1>Submitted Data</h1>
13        First Name:<strong><c:out value="${fname}" /></strong><br />
14        Last Name:<strong><c:out value="${lname}" /></strong>
15    </body>
16 </html>
```



JSP Scope



- Different variables can have different scope in JSPs
- There are four different scopes in JSPs
 - › Application
 - › Session
 - › Page
 - › Request
- The following variable would remain in scope as long as the session is valid



```
<c:set var="fname" value="{param.fname}" scope="session" />
```



- For more information on JSPs
 - › Go to <http://docs.oracle.com/javaee/5/tutorial/doc/bnagy.html>
 - › Go through one of the many JSP tutorials online
- For more information on JSTL
 - › Go to <http://docs.oracle.com/javaee/5/jstl/1.1/docs/tlddocs/>
 - › Go to <https://jstl.java.net/>





Outline

- JSP
- Program

Program



- Modify the program you created during the Servlet lecture to have the HTML form submit to a JSP instead.

School Form

Email: jeffrey.miller@usc.edu
Password:
Birthday: 01/01/2017
New Student?: Yes No
College: USC
Major: Computer Science
Favorite Color:
Terms and Conditions: I agree.

Submitted Data

Field	Value
Email	jeffrey.miller@usc.edu
Password	csci201
Birthday	2017-01-01
New Student	yes
College	USC
Major	CS
Color	#ffff00
Terms and Conditions	yes