

Java Server Pages

CSE 487/587
Jan 31, 2005

References:

<http://java.sun.com/products/jsp/>
<http://www.jsptut.com>
Vijay Arthanari's presentation on JSP

Introduction

- JSP technology allows creation of web content with both dynamic and static content
- It provides the dynamic capabilities of Java Servlets technology but are easier to write. Most containers that support JSP will convert the JSP to a servlet class during deployment
- JSP supports a defined set of standard tags and user-defined custom tags

When to use Servlets

- Use servlets to implement services – servlet can perform whatever service it provides (templating, security, personalization, application control) then select and forward the request for display to a presentation component (JSP page)
- Use servlets as a web tier controller, which determines how to handle a request and chooses the next view to display.
- Avoid writing servlets that print mostly static text

When to use JSP

- Use JSP pages for Data Presentation
- Not appropriate for creating content with highly variable structure or for controlling request routing.
- Use JSP pages to generate XML
- Use JSP pages to generate unstructured textual content such as ASCII text, fixed-width or delimited data, and even PostScript.
- Ideal for assembling textual data from multiple sources

What is a JSP page?

- A JSP page is a text document with two types of text:
 - Static template data expressed in any text-based format, such as HTML, SVG, WML and XML
 - JSP elements: Special markup for including other text or executing embedded logic which construct dynamic content.
- A JSP page services requests as a Servlet. The life cycle and many of the capabilities of JSP pages (in particular the dynamic aspects) are determined by Java Servlet technology.

JSP Elements

- There are basically three different forms of JSP elements:
 - Directives: Instructions that control the behavior of the JSP page compiler; evaluated at page compilation time.
 - Scripting elements: Blocks of Java code embedded in the JSP page between the delimiters `<%` and `%>`.
 - Custom tags: Programmer-defined markup tags to generate dynamic content when the page is served.

JSP Elements

- Page directive: Defines attributes that apply to a JSP page. Example,
 - `<%@ page import="java.util.*, java.sql.*" %>`
- Include: Includes another JSP file within the current one
 - `<%@ include file="relativeURL" %>`
- Expression: Contains an expression that would get written onto the output page (HTML)
 - `<%= var1 + var2 %>`
- Scriptlet: Contains a code fragment
 - `<% Java code %>`
- Complete JSP syntax detailed at:
<http://java.sun.com/products/jsp/syntax/2.0/card20.pdf>

Example: HelloJSP (index.jsp)

```
<%@ page import="java.sql.*" %>
<!-- This is a comment in JSP. It will not be shown to the browser at all -->
<!-- This is a comment in HTML. This will be shown at the browser.
So I might as well put my name here. :-)>

Written by: Mohit Vora (mhvora@buffalo.edu)
Teaching Assistant
Dept. of Computer Science & Engrg.
CSE 4587, Spring 2005 -->

<html>
<head>
<title>A JSP Example</title>
<link rel="stylesheet" type="text/css" href="format.css"/>
</head>
<body>
<!--Using JSP to display records from the database/>
<%
try {
Class.forName("oracle.jdbc.driver.OracleDriver");
}
catch (ClassNotFoundException e) {
e.printStackTrace();
out.println("Database Driver cannot be loaded. Please check back later.");
}

int alternator = 0;
try {
Connection conn = DriverManager.getConnection(
"jdbc:oracle:thin:mvora/2004@orasserve.cse.buffalo.edu:1521:csedb");
Statement stmt = conn.createStatement();
```


