Java Server Page (JSP)

- **Why?**
  - It’s tedious to generate HTML using `println()`
  - Separate presentation from processing

- **How?**
  - *Java code* embedded in *HTML documents*

---

**HelloJSP1.jsp**

```html
<!--DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN" -->
<html>
<head><title>Hello JSP 1</title></head>
<body>A JSP without J or S.</body>
</html>
```

**HelloJSP2.jsp**

```html
<!--DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN" -->
<html>
<head><title>JSP Hello World</title></head>
<body>Hello World on <%= new java.util.Date() %>.
</body>
</html>
```

---

**How Does JSP Work?**

- In package `org.apache.jsp`
- `_jspService()` handles everything
  - replaces `service()` in `HttpServlet`
  - What happened to `doGet()`, `doPost()`, ...??
- HTML text → `out.write(...)`

---

**Some Simple Observations about the JSP/Servlets**

- Look under `$TOMCAT_HOME/work/Catalina/localhost/st/context_name`
JSP Components

- HTML template text
- Code elements of Java
  - Directives
  - Comments
  - Scripting elements
  - Beans
  - Expression language
  - Custom tag libraries

Directives

- `<%@ type attr="value" ... %>`
- Three types of directives
  - page
  - include
  - taglib

Some Page Directives

```
<%@ page import="java.util.*, java.util.zip.*" %>
<%@ page contentType="text/html" %>
<%@ page session="false" %> <!-- default is true -->
<%@ page pageEncoding="Shift_JIS" %>
```

Comments

- `<%%-- Hidden Comments --%>`
- `<!-- Output (HTML) Comments -->`

JSP Scripting Elements

- JSP Expressions
- JSP Scriptlets
- JSP Declarations

JSP Expressions

- `<%= Java expression %>`
  - What’s an expression??

```
<%= expression %>
```

```java
out.write( expression ) in _jspService()
```
Example: Add.jsp

- Add two integer parameters and display the sum

Pre-defined Variables

- request, response, out
- session, application
- config, pageContext
- page

The same request, response, session etc. that are used in servlets – see the Servlet API documentation for what you can do with them.

JSP Scriptlets

- `<% Java code %>`
- All code goes inside _jspService()

Example: Add.jsp Again

- Add the code to validate the parameters

Another Scriptlet Example

```jsp
<% if (Math.random() < 0.5) { %>
<H1>Have a <I>nice</I> day!</H1>
<% } else { %>
<H1>Have a <I>lousy</I> day!</H1>
<% } %>
```

JSP Declarations

- `<%! class variables or methods %>`
- All code goes outside _jspService()
Example: RequestCounter.jsp

- Initialize a counter as a class variable
- Increment the counter each time the JSP is requested

About JSP Scripting Elements

- There is a straightforward mapping between JSP with scripting elements and servlet
- Mixing presentation and processing:
  - hard to debug
  - hard to maintain
- No clean and easy way to reuse code
- Do NOT use them