

JSP (Java Server Pages) - Complete Guide

Chapter 1 : 1. What is JSP?

JSP (Java Server Pages) is a Java-based technology used to create dynamic web pages.

- Combines HTML with Java code to generate web pages dynamically.
- Runs on the server, generating HTML content to be sent to the client browser.
- Easier to write and maintain compared to pure servlets.

Chapter 2 : 2. Why Use JSP?

- To produce dynamic content based on user requests, database data, session data, etc.
- Integrates well with Java EE ecosystem.
- Allows embedding Java code directly into HTML.

Chapter 3 : 3. JSP Page Structure

HTML part

```
<html>
<head><title>Sample JSP</title></head>
<body>
  <h1>Welcome to my site!</h1>
</body>
</html>
```

Java Code Scriptlet

```
<%
    String username = "CrazyBoy";
    out.println("Hello, " + username + "!");
%>
```

Expression (Outputs Java expression result)

Hello, <%= username %>!

JSP (Java Server Pages) - Complete Guide

Chapter 4 : 4. JSP Lifecycle (Brief)

1. User requests JSP page.
2. Server compiles JSP to a servlet (if not already compiled).
3. Servlet executes and produces HTML.
4. HTML sent to client browser.

Chapter 5 : 5. Key JSP Elements and Syntax

Syntax	Purpose	Example
<% code %>	Scriptlet (Java code block)	<% out.println("Hi"); %>
<%= expr %>	Expression (output result)	<%= new Date() %>
<%! decl %>	Declaration (variables/methods)	<%! int count = 0; %>
<%@ directive %>	Page directive	<%@ page language="java" %>
<jsp:include>	Include another resource	<jsp:include page="header.jsp" />

Chapter 6 : 6. JSP Implicit Objects

- request - HTTP request object
- response - HTTP response object
- session - Session object for current user
- application - Servlet context object
- out - PrintWriter to send output
- config - Servlet config object
- pageContext - JSP page context
- page - Current page (this)
- exception - Exception object (for error pages)

Chapter 7 : 7. Example: Simple JSP Page with Parameter

```
<%@ page contentType="text/html; charset=UTF-8" language="java" %>
```

JSP (Java Server Pages) - Complete Guide

```
<html>

<head><title>My JSP Example</title></head>

<body>

<%

    String name = request.getParameter("name");

    if (name == null || name.isEmpty()) {        name

    = "Guest";

    }

%>

<h1>Welcome, <%= name %>!</h1>

</body>

</html>
```

- Access with <http://yourserver/page.jsp?name=CrazyBoy>
- Output: "Welcome, CrazyBoy!"

Chapter 8 : 8. Session Handling in JSP

```
<%

    String user = (String) session.getAttribute("username");

    if (user == null) {        response.sendRedirect("login.jsp");

    } else {        out.println("Welcome

back, " + user);

    }

%>
```

Chapter 9 : 9. Advantages of JSP

- Easy to learn for Java developers.
- Seamless integration with Java code.
- Generates dynamic content efficiently.

JSP (Java Server Pages) - Complete Guide

- Supports MVC architecture with Servlets and frameworks.

Chapter 10 : 10. Disadvantages of JSP

- Mixing Java code with HTML can make code messy.
- Modern frameworks like JSF, Thymeleaf are often preferred.
- Debugging scriptlets can be harder.

Chapter 11 : 11. Summary

- JSP allows you to embed Java in HTML for dynamic web content.
- It's compiled into servlets by the server.
- Supports all standard web programming needs: sessions, request params, output, includes.
- Should be used with clean MVC patterns for best results.