

## **JSP Part 2**

# Agenda

---

- JSP vs Servlet
- LifeCycle of Servlet
- JSP Elements
- JSP Page directive
- Directives vs Action tags

# Review of last session on JSP

---

- JSP vs Servlet
- Model1 vs Model 2 (MVC) architecture
- JSP Elements
  - Declaration
  - Expression
  - Scriptlets

# JSP vs Servlet

---

Servlet = HTML in a Java class

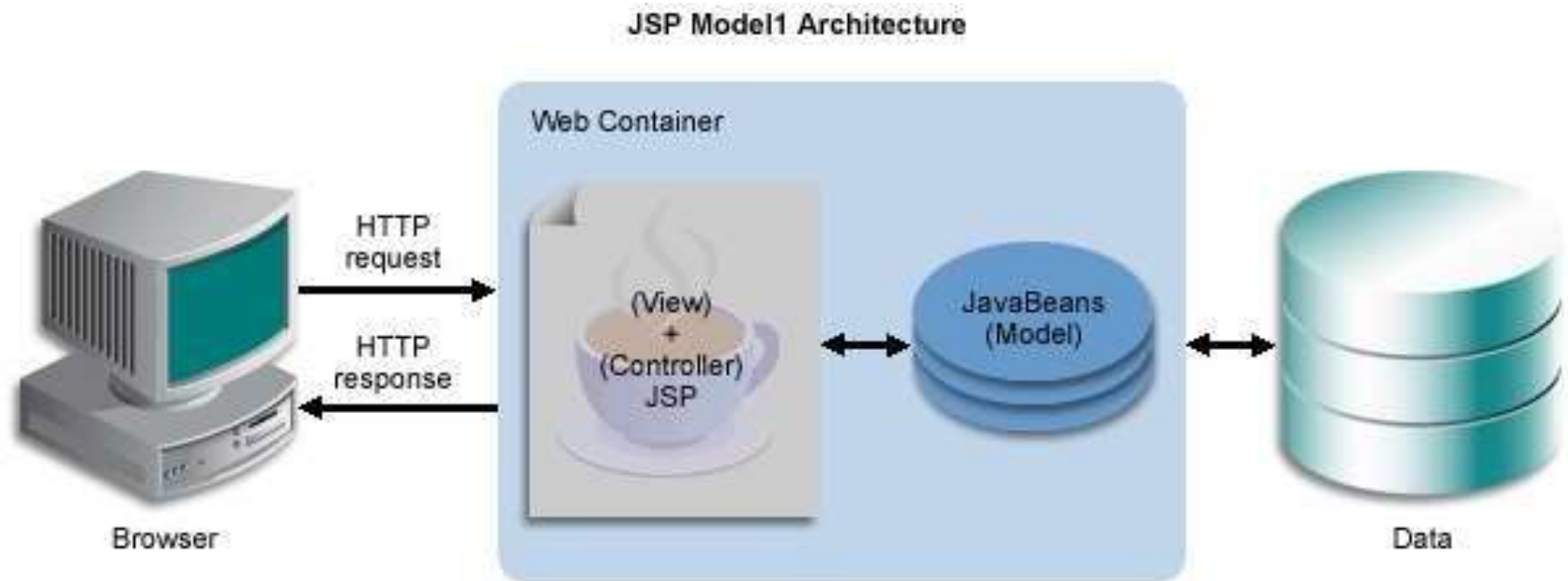
```
out.println("<h1> Hello World! </h1>");
```

JSP = Java in an HTML page

```
<%= request.getParameter("title") %>
```

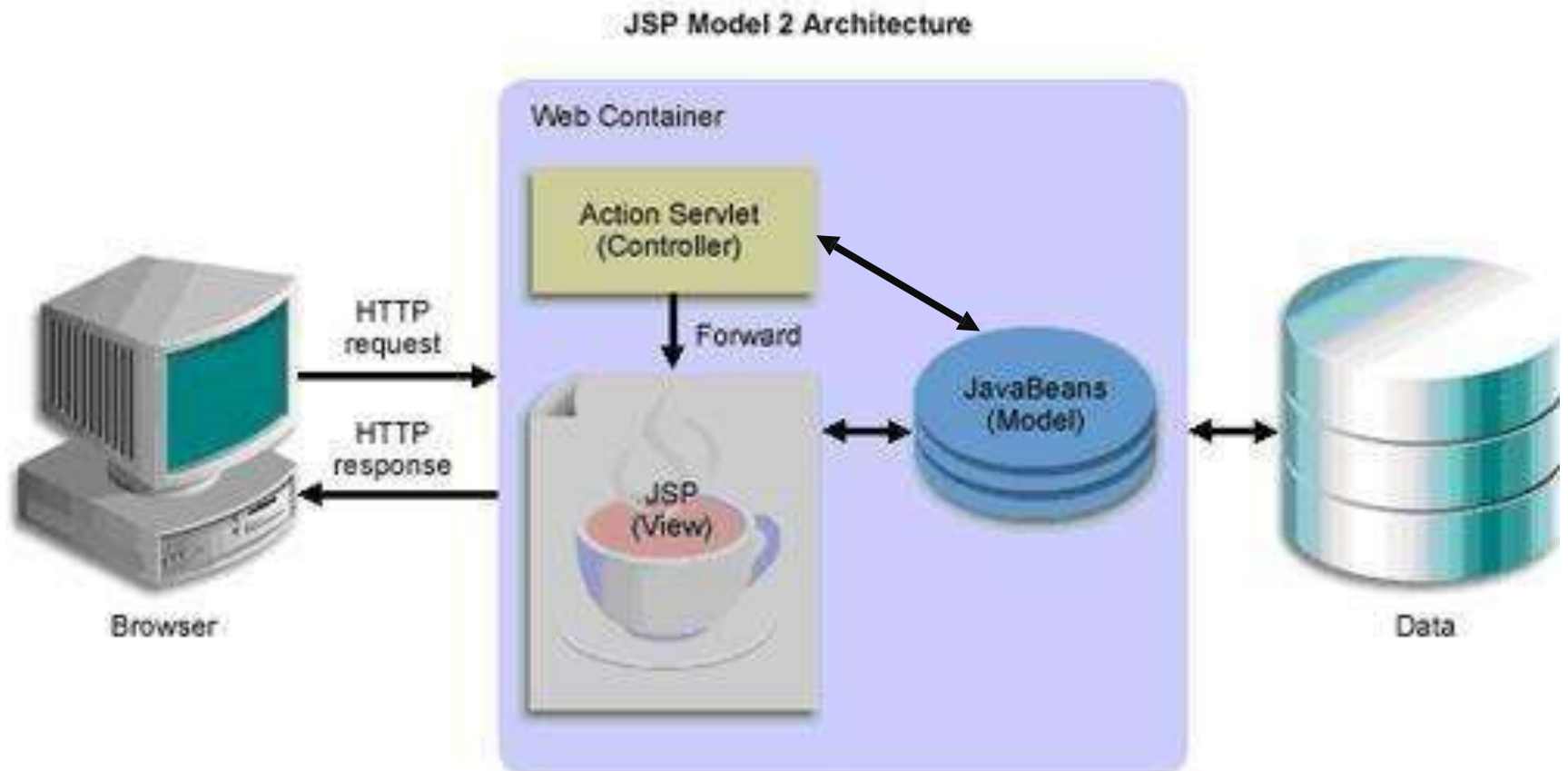
# Model 1 Architecture

---

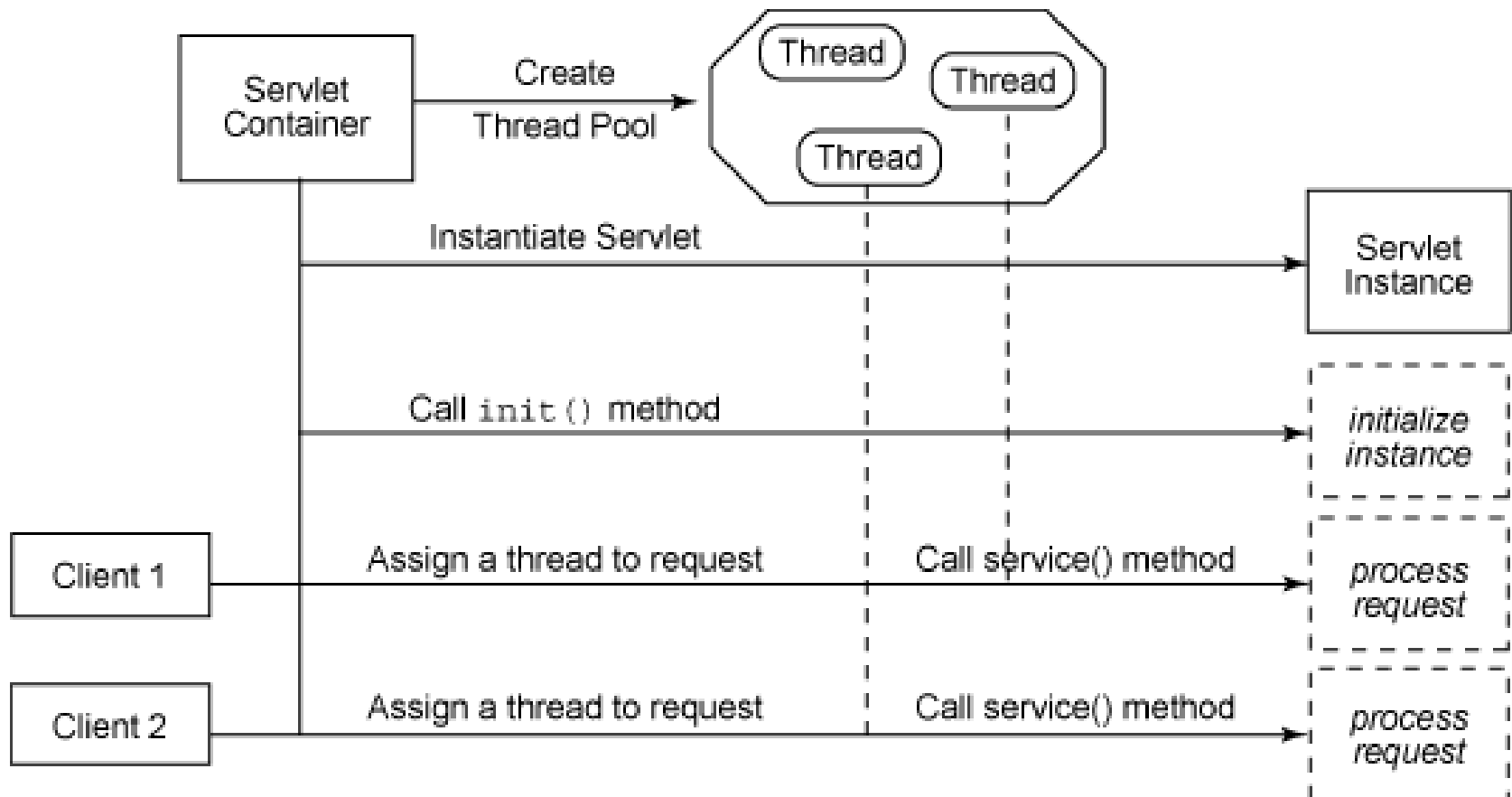


# Model 2 Architecture

---



# LifeCycle of Servlet



# JSP Elements

---

- Declarations      `<%! code %>`
- Expressions      `<%= expression %>`
- Scriptlets      `<% code %>`



# **Directives - page, include, taglib**

# JSP Page Directive

---

- Directives are messages to the JSP container and do not produce output into the current output stream

- Syntax:**

```
<%@ directive attribute="value" %>
```

```
<%@ directive attribute1="value1"
```

```
attribute1="value2" ... %>
```

There are three types of directives:

1. **page**
2. **include**
3. **taglib**

XML form:

```
<jsp:directive.directiveType attribute="value" />
```

# Page Directive

---

- Defines attributes that apply to an entire JSP page.

<%@ page

[ language="java" ]

[ extends="package.class" ]

[ import="{package.class | package.\*}, ..." ]

[ session="true|false" ]

[ buffer="none|8kb|sizekb" ]

[ autoFlush="true|false" ]

[ isThreadSafe="true|false" ]

[ info="text" ]

[ errorPage="relativeURL" ]

[ contentType="mimeType [ ;charset=characterSet ]" ]

isErrorPage="true|false" ]

%>

# Page directive example..

---

```
<%@ page import="java.util.*" %>
```

```
<HTML>
```

```
<BODY>
```

```
<%
```

```
    System.out.println( "Evaluating date now" );
```

```
    Date date = new Date();
```

```
%>
```

```
Hello! The time is now <%= date %>
```

```
</BODY>
```

```
</HTML>
```

# Include Directive

---

- Includes a static file in a JSP file, parsing the file's JSP elements.
  - Syntax

**<%@ include file="relativeURL" %>**

**The <%@ include %> directive inserts a file of text or code in a JSP file at translation time, when the JSP file is compiled.**

**<%@ include %> process is static. A static include means that the text of the included file is added to the JSP file.**

**You may include a JSP or HTML file like this. E.g. for headers and footers.**

# Include directive example...

---

- index.jsp

```
<html>
```

```
<head>
```

```
<title>Main JSP Page</title>
```

```
</head>
```

```
<body>
```

```
  <%@ include file="file1.jsp" %>
```

Main JSP Page: Content between two include directives.

```
  <%@ include file="file2.jsp" %>
```

```
</body>
```

```
</html>
```

- file1.jsp

```
<p align="center">
```

This is my File1.jsp and I will include it in index.jsp using include directive

```
</p>
```

- file2.jsp

```
<p align="center">
```

This is File2.jsp

```
</p>
```

# Taglib Directive

---

- Defines a tag library and prefix for the custom tags used in the JSP page.

```
<%@ taglib uri="URIToTagLibrary" prefix="tagPrefix" %>
```

```
<%@ taglib uri="http://thathost/tags" prefix="public" %>
```

```
<public:loop>
```

```
</public:loop>
```

The **<%@ taglib %>** directive declares that the JSP file uses custom tags, names the tag library that defines them, and specifies their tag prefix.

**We will look into this further when we look at frameworks like Struts and Spring.**

# **JSP Action Tags: include, forward, useBean**



# Directives vs Action tags

---

- Directives are used during translation phase while actions are used during request processing phase.
- Unlike Directives, Actions are re-evaluated each time the page is accessed.

# <jsp:include>

---

```
<html>
<head>
<title>Demo of JSP include Action Tag</title>
</head>
<body>
<h3>JSP page: Demo Include</h3>
<jsp:include page="sample.jsp" />
</body>
</html>
```

# <jsp:include> with parameters

---

## index.jsp

```
<html>
<head>
<title>JSP Include example with parameters</title>
</head>
<body>
<h2>This is index.jsp Page</h2>
<jsp:include page="display.jsp">
<jsp:param name="userid" value="Pawan" />
<jsp:param name="password" value="Pawan" />
<jsp:param name="name" value="Pawan Kumar" />
<jsp:param name="age" value="27" />
</jsp:include>
</body>
</html>
```

## display.jsp

```
<html>
<head>
<title>Display Page</title>
</head>
<body>
<h2>Hello this is a display.jsp Page</h2>
UserID: <%=request.getParameter("userid") %><br>
Password is: <%=request.getParameter("password") %><br>
User Name: <%=request.getParameter("name") %><br>
Age: <%=request.getParameter("age") %>
</body>
</html>
```

# <jsp:forward> without parameters

---

- **first.jsp**

```
<html>
<head>
<title>Demo of JSP Forward Action Tag</title>
</head>
<body>
<h3>JSP page: Demo forward</h3>
<jsp:forward page="second.jsp" />
</body>
</html>
```

# <jsp:forward> with parameters

---

## index.jsp

```
<html>
<head>
<title>JSP Include example with parameters</title>
</head>
<body>
<h2>This is index.jsp Page</h2>
<jsp:forward page="display.jsp">
<jsp:param name="userid" value="Pawan" />
<jsp:param name="password" value="Pawan" />
<jsp:param name="name" value="Pawan Kumar" />
<jsp:param name="age" value="27" />
</jsp:forward>
</body>
</html>
```

## display.jsp

```
<html>
<head>
<title>Display Page</title>
</head>
<body>
<h2>Hello this is a display.jsp Page</h2>
UserID: <%=request.getParameter("userid") %><br>
Password is: <%=request.getParameter("password") %><br>
User Name: <%=request.getParameter("name") %><br>
Age: <%=request.getParameter("age") %>
</body>
</html>
```

## <jsp:useBean> : Bean class

---

- First write a bean Details.java with username, age, password as properties.

```
public class Details {  
    public Details() {  
    }  
    private String username;  
    private int age;  
    private String password;  
  
    // Write getter and setter methods here...  
  
}
```

## <jsp:useBean> : index.jsp

---

```
<html>
```

```
<head><title>
```

```
useBean, getProperty and setProperty example
```

```
</title></head>
```

```
<form action="userdetails.jsp" method="post">
```

```
User Name: <input type="text" name="username"><br>
```

```
User Password: <input type="password" name="password"><br>
```

```
User Age: <input type="text" name="age"><br>
```

```
<input type="submit" value="register">
```

```
</form>
```

```
</html>
```

## <jsp:useBean>: userDetails.jsp

---

```
<jsp:useBean id="userinfo" class="mypackage.Details">  
</jsp:useBean>
```

```
<jsp:setProperty property="*" name="userinfo"/>
```

You have entered below details:<br>

```
<jsp:getProperty property="username" name="userinfo"/><br>
```

```
<jsp:getProperty property="password" name="userinfo"/><br>
```

```
<jsp:getProperty property="age" name="userinfo" /><br>
```



# Conclusion

---

JavaServer Pages (JSP) lets you separate the dynamic part of your pages from the static HTML.

1. One can simply write the regular HTML in the normal manner, using whatever Web-page-building tools you normally use.
2. One can enclose then the code for the dynamic parts in special tags, most of which

start with `"<%"`

and end with `"%>"`