



Servlet/JSP Connect #3

Agenda

1

Inter Servlet communication

2

Introduction to JSP

Objectives

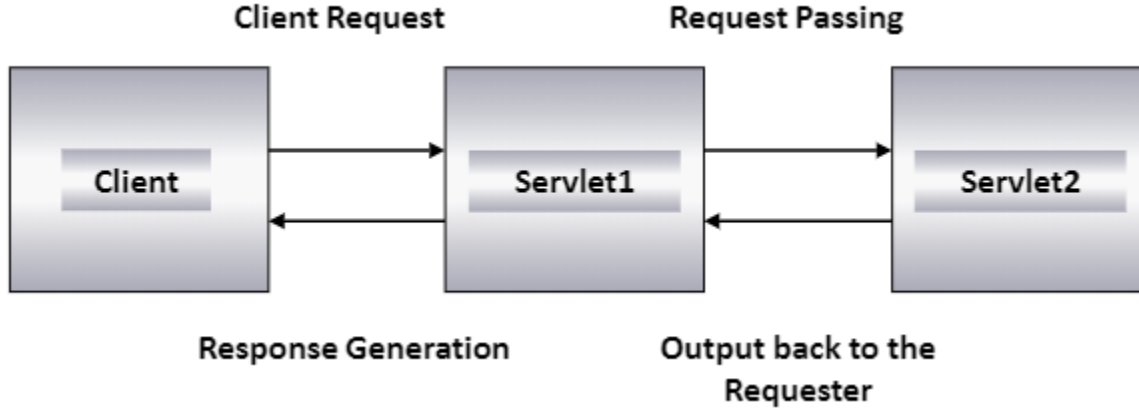
At the end of this session, you will be able to:

- Understand how inter servlet communication happens
- Understand JSP Elements - scriptlet, expression and declaration

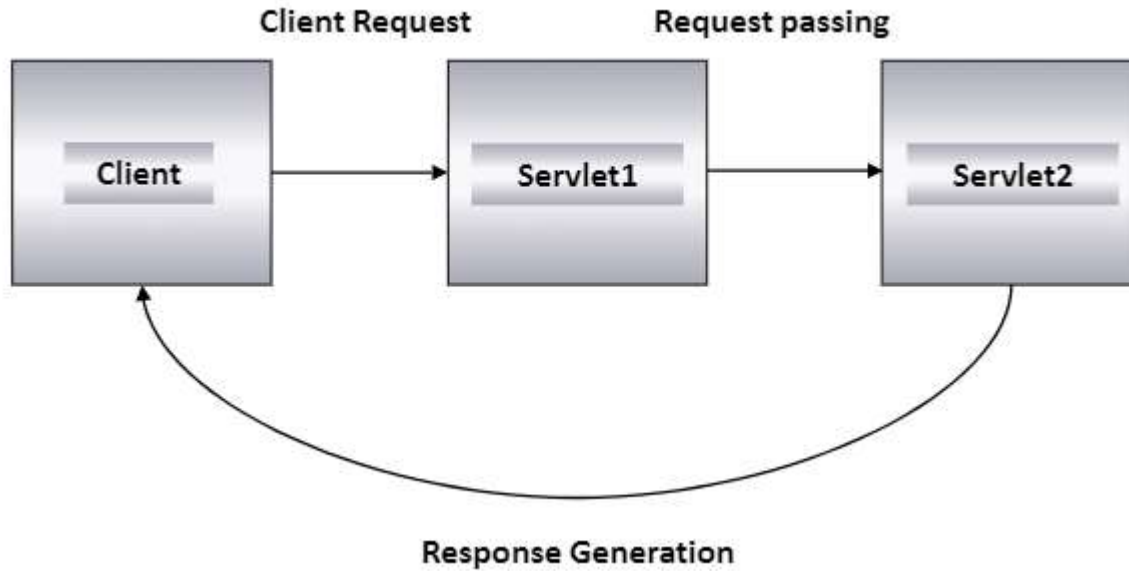
RequestDispatcher include/forward-



RequestDispatcher – include



RequestDispatcher – forward



RequestDispatcher include - Demo

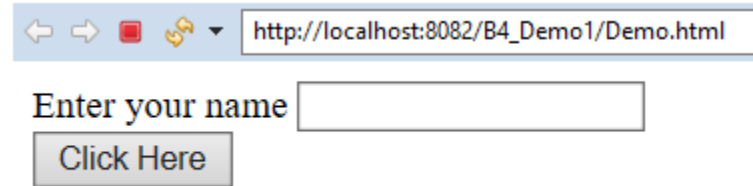


To Do List

1. Create a Demo.html page with a textbox called name and submit button
2. Create 2 Servlets
 - a) Create WelcomeServlet1 – Which can convert name into uppercase
 - b) Create WelcomeServlet2 – Which can find the length of name

Demo.html

```
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<form action="WelcomeServlet1">
Enter your name <input type="text" name="name"/><br>
<input type="submit" value="Click Here"/>
</form>
</body>
</html>
```



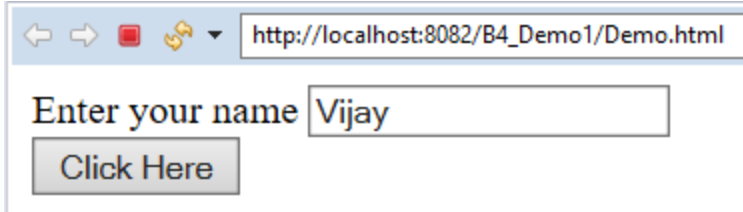
WelcomeServlet1

```
public class WelcomeServlet1 extends HttpServlet {  
  
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws  
        ServletException, IOException {  
        PrintWriter out = response.getWriter();  
        String name = request.getParameter("name");  
        name = name.toUpperCase();  
        out.print("<h1> Name : " + name + "</h1>");  
        RequestDispatcher requestDispatcher = request.getRequestDispatcher("WelcomeServlet2");  
        requestDispatcher.include(request, response);  
        out.print("<h3>Thank you</h3>");  
        out.close();  
    }  
}
```

WelcomeServlet2

```
public class WelcomeServlet2 extends HttpServlet {  
  
    protected void doGet(HttpServletRequest request, HttpServletResponse response)  
        throws ServletException, IOException {  
  
        String name = request.getParameter("name");  
        PrintWriter out = response.getWriter();  
        out.print("<h2> Length of the name : " + name.length() + "</h2>");  
  
    }  
}
```

RequestDispatcher – Demo 1



Enter your name

[Click Here](#)



Name : VIJAY

Length of the name : 5

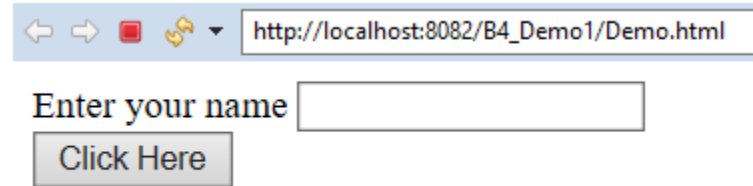
Thank you

RequestDispatcher forward-demo



Reuse Demo.html

```
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<form action="WelcomeServlet1">
Enter your name <input type="text" name="name"/><br>
<input type="submit" value="Click Here"/>
</form>
</body>
</html>
```



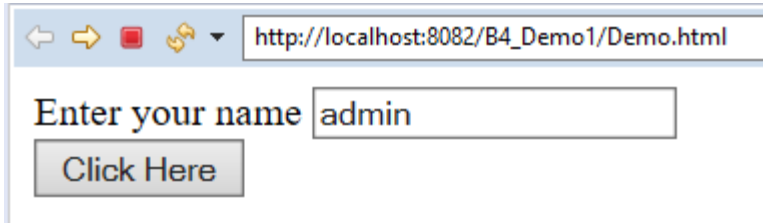
WelcomeServlet1

```
public class WelcomeServlet1 extends HttpServlet {  
  
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws  
        ServletException, IOException {  
        PrintWriter out = response.getWriter();  
        String name = request.getParameter("name");  
        name = name.toUpperCase();  
        request.setAttribute("data", name);  
        RequestDispatcher requestDispatcher = request.getRequestDispatcher("WelcomeServlet2");  
        requestDispatcher.forward(request, response);  
        out.close();  
    }  
}
```

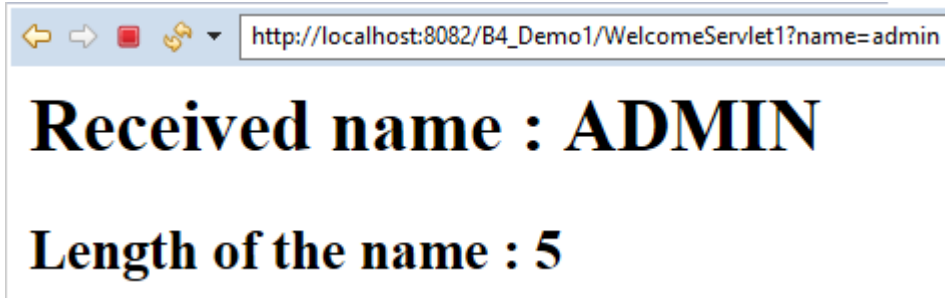
WelcomeServlet2

```
public class WelcomeServlet2 extends HttpServlet {  
  
    protected void doGet(HttpServletRequest request, HttpServletResponse response)  
        throws ServletException, IOException {  
  
        String receivedName = request.getAttribute("data").toString();  
        PrintWriter out = response.getWriter();  
        out.print("<h1> Received name : " + receivedName + "</h1>");  
        out.print("<h2> Length of the name : " + receivedName.length() + "</h2>");  
    }  
}
```


RequestDispatcher – Demo 2



A screenshot of a web browser window. The address bar shows the URL `http://localhost:8082/B4_Demo1/Demo.html`. Below the address bar, there is a text input field with the label "Enter your name" and the text "admin" entered. Below the input field is a button labeled "Click Here".



A screenshot of a web browser window. The address bar shows the URL `http://localhost:8082/B4_Demo1/WelcomeServlet1?name=admin`. The main content area displays two lines of text in a large, bold, black serif font: "Received name : ADMIN" and "Length of the name : 5".

Introduction to JSP

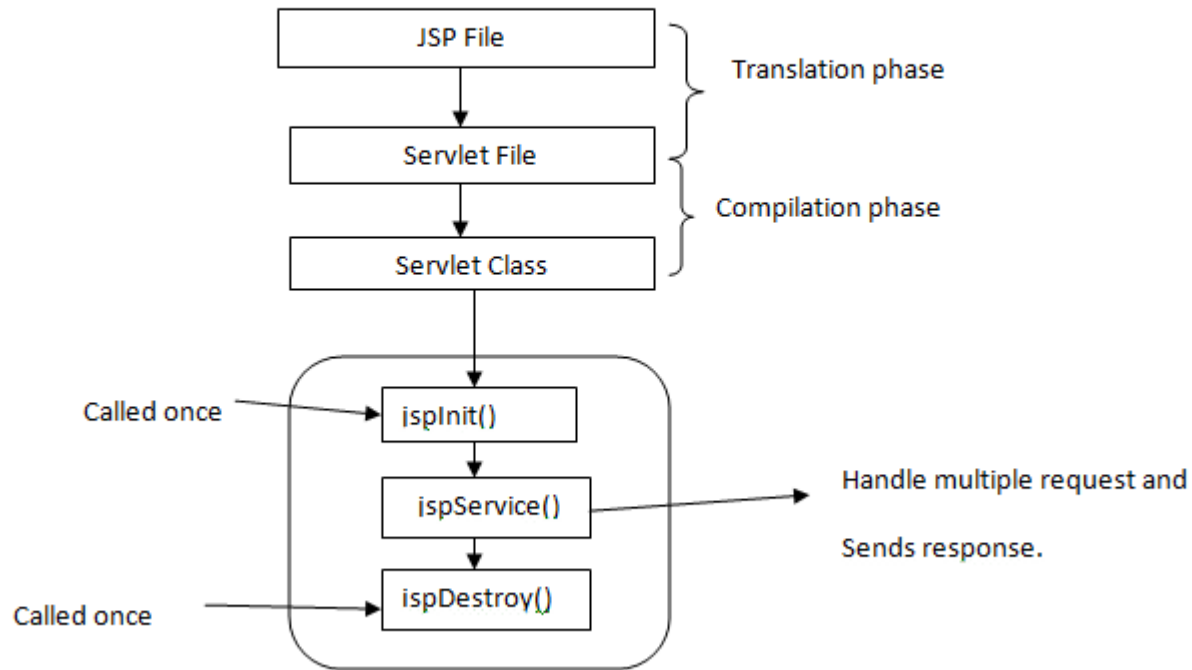


Introduction to JSP

- JSP stands for Java Server Pages
- JSP technology is used to create web applications just like Servlet technology
- JSP page consists of HTML tags and JSP tags
- JSP is used to present dynamic content.

Life cycle of JSP

JSP Life Cycle

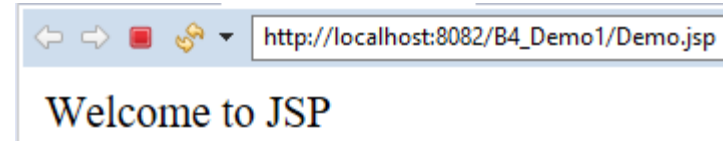


JSP Scripting elements

- The scripting elements provides the ability to include Java code in the JSP.
- There are three types of scripting elements:
 1. scriptlet tag
 2. expression tag
 3. declaration tag

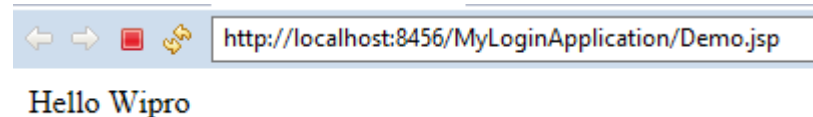
Example for scriptlet tag – Create a Demo.jsp page

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<% out.print("Welcome to JSP"); %>
</body>
</html>
```



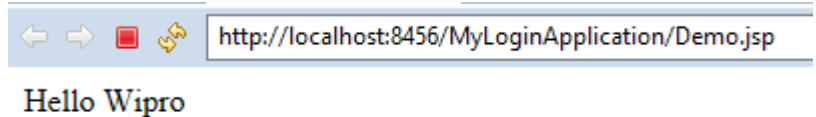
Example for declaration tag – Modify the Demo.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<%! String name = "Wipro"; %>
<% out.print("Hello " + name); %>
</body>
</html>
```



Example for expression tag – Modify the Demo.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<%! String name = "Wipro"; %>
Hello <%= name %>
</body>
</html>
```



Login application using JSP without JDBC



Login.html

```
<html>
<body>
<form action="LoginJsp.jsp" name="LoginForm">
<div>
<br>
<table>
<tr><td>User Name</td><td> <input type="text" name="uname"/></td></tr>
<tr><td>Password </td><td><input type="password" name="pwd"/></td></tr>
<tr><td colspan="2" align="center"><input type="submit" value="Login" /></td></tr>
</table>
</div>
</form>
</body>
</html>
```

LoginJsp.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head> <meta charset="ISO-8859-1"> <title>Insert title here</title>
</head>
<body>
<%
    String userName = request.getParameter("uname");
    String password = request.getParameter("pwd");
    if(userName.equals("Admin") && password.equals("Wipro@123")){
        out.print("<font color=green><b>Welcome Admin !</b></font>");
    }else{
        out.print("<font color=red><b>Invalid user name or password</b></font>");
    }
%>
</body>
</html>
```

Login application using JSP without JDBC

http://localhost:8456/MyLoginApplication/Login.html

User Name

Password

http://localhost:8456/MyLoginApplication/Login.jsp?uname=Admin&pwd=Wipro@123

Welcome Admin !

http://localhost:8456/MyLoginApplication/Login.html

User Name

Password

http://localhost:8456/MyLoginApplication/Login.jsp?uname=Hello&pwd=123

Invalid user name or password

Login application using MVC



1. Login.html

```
<form name="LoginForm" method="post" action="LoginServlet">
<div>
<br>
<table>
<tr><td>User Name</td><td> <input type="text" name="uname"/></td></tr>
<tr><td>Password </td><td><input type="password" name="pwd"/></td></tr>
<tr><td colspan="2" align="Center"><input type="Submit" value="Login" onclick="return
loginValidation()"/></td></tr>
</table>
</div>
</form>
```

2. Reuse the UserBean class – No change

```
package com.wipro.bean;
public class UserBean {
private String userName;
private String password;

public UserBean() {
}
public String getUserName() {
return userName;
}
public void setUserName(String userName) {
this.userName = userName;
}
public String getPassword() {
return password;
}
public void setPassword(String password) {
this.password = password;
}
}
```

3. Reuse the DBUtil class – No change

```
package com.wipro.util;
public class DBUtil {
    public static Connection getConnection(){
        Connection con = null;
        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            con =
            DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:orcl", "scott", "tiger");
        }
        catch(Exception e){}
        return con;
    }
}
```


4. Reuse the UserDao class – No change

```
package com.wipro.dao;
public class UserDao {
public String checkUser(UserBean userBean){
    String status = "Fail";
    try{
        String sql = "select * from user_table where username=? and password=?";
        Connection con = DBUtil.getConnection();
        PreparedStatement st = con.prepareStatement(sql);
        st.setString(1, userBean.getUserName());
        st.setString(2, userBean.getPassword());
        ResultSet rs = st.executeQuery();
        if(rs.next())
            status = "Success";
        st.close();
        con.close();
    }
    catch(Exception e){ e.printStackTrace(); }
    return status;
}
}
```

LoginServlet

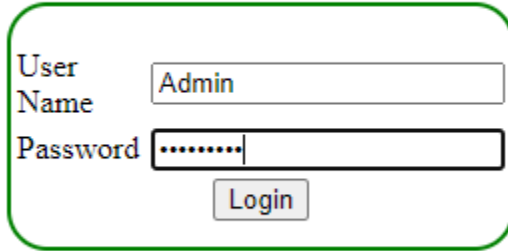
```
public class LoginServlet extends HttpServlet {  
  
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws  
        ServletException, IOException {  
        // a. Fetch the input values  
        String name = request.getParameter("uname");  
        String pwd = request.getParameter("pwd");  
  
        // b. Create an UserBean object and initialize  
        UserBean bean = new UserBean(name, pwd);  
  
        // c. create UserDao object  
        UserDao dao = new UserDao();  
  
        // d. attach the result as an attribute  
        request.setAttribute("result", dao.checkUser(bean));  
  
        // e. forward the request to Result.jsp  
        request.getRequestDispatcher("Result.jsp").forward(request, response);  
    }  
}
```

Result.jsp

```
<body>
<%
String result=(String)request.getAttribute("result");
String name=request.getParameter("uname");
if (result.equals("Success"))
{
%>
<h1 style="color:blue">Welcome <%=name %></h1>
<%
}
else
{
%>
<h1 style="color:red">Sorry <%=name %>, you are not a valid
user</h1>
<%} %>
</body>
```

Login application using MVC

← → ↻ ⓘ localhost:6789/LoginApp2/Login.html



User Name

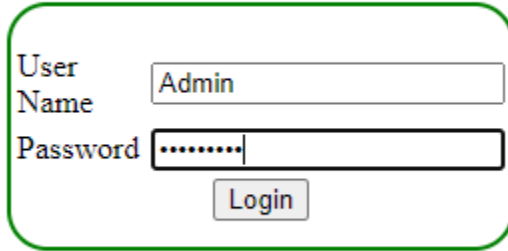
Password

← → ↻ ⓘ localhost:6789/LoginApp2/LoginServlet

Welcome Admin

Login application using MVC

← → ↻ ⓘ localhost:6789/LoginApp2/Login.html



User Name

Password

← → ↻ ⓘ localhost:6789/LoginApp2/LoginServlet

Sorry Admin, you are not a valid user

Knowledge Check



Knowledge Check

- Which of the following is the scriptlet tag in JSP?
 1. `<% %>`
 2. `<%! %>`
 3. `<%= %>`
- Which of the following is the declaration tag in JSP?
 1. `<% %>`
 2. `<%! %>`
 3. `<%= %>`

Knowledge Check

- Which of the following is the expression tag in JSP?
 1. `<% %>`
 2. `<%! %>`
 3. `<%= %>`
- JSPs are eventually compiled into Java servlets, you can do as much with JSPs as you can do with Java servlets.
 1. True
 2. False

Knowledge Check

- Which of the following tags is used to execute Java source code in JSP?
 1. Declaration tag
 2. Scriptlet tag
 3. Expression tag

Exercise



Exercise-1

- Create a HTML page to accept a name as an input (textbox) from the user
- Create a submit button with the value of “Say Hello” in that HTML page
- If the submit button is clicked, it should invoke a JSP which displays a Hello message (Hello and the username)

Exercise-2

- Create a HTML page to accept two numbers (integer) from the user
- Create a submit button with the value “Addition” in that HTML page
- If the submit button is clicked, it should invoke a JSP which should display the sum of the two numbers

Exercise-3

- Create a HTML page to accept a number(integer) as an input from the user.
- Create a submit button with the value of “Factorial” in that HTML page.
- If the submit button is clicked, it should invoke a JSP which should display the factorial of the given number



Thank you