

# 내장 객체와 액션 태그

# JSP 내장 객체

## ❖ JSP 내장 객체

- 직접 생성하는 과정없이 바로 사용할 수 있는 객체
- 변환 서블릿에서 자동으로 생성해서 전달

분류	내장 객체
입출력 관련 객체	request
	response
	out
서블릿 관련 객체	page
	config
외부 환경 정보 제공 객체	session application pageContext
예외 관련 객체	exception

# JSP 내장 객체

## ❖ JSP 내장 객체

- out 내장 객체
  - 서버에서 클라이언트로 열려있는 출력 스트림
- request 내장 객체
  - 요청과 관련된 정보를 저장 관리하는 객체

메소드	설명
getContextPath()	JSP 페이지가 속한 웹 애플리케이션의 컨텍스트 패스를 구합니다.
getMethod()	요청 방식이 GET 방식인지 POST 방식인지 알려줍니다.
getRequestURL()	요청 URL을 구합니다.
getRequestURI()	요청 URL에서 쿼리 스트링을 제외한 부분을 구합니다.
getQueryString()	요청 URL 다음에 오는 쿼리 스트링을 구합니다.
getSession(flag)	요청 관련된 세션 객체를 구합니다.
getRequestDispatcher(path)	지정 로컬 URL에 대한 RequestDispatcher 객체를 구합니다.
getRemoteHost()	요청한 호스트의 완전한 이름을 구합니다.
getRemoteAddr()	요청한 호스트의 네트워크 주소를 구합니다.

# JSP 내장 객체

## ❖ JSP 내장 객체

### ○ request 내장 객체

메소드	설명
getRemoteUser()	요청한 사용자의 이름을 구합니다.
getSession()	세션 객체를 구합니다.
getServerName()	서버의 이름을 구합니다.
getProtocol()	사용 중인 프로토콜을 알려줍니다.

### ❖ 01\_requestinfo.jsp

```
컨텍스트 패스 : <%= request.getContextPath() %> <br>  
요청방식 : <%= request.getMethod() %> <br>  
요청한 URL : <%= request.getRequestURL() %> <br>  
요청한 URI : <%= request.getRequestURI() %> <br>  
서버의 이름 : <%= request.getServerName() %> <br>  
프로토콜 : <%= request.getProtocol() %> <br>
```

# JSP 내장 객체

## ❖ JSP 내장 객체

### ○ request 내장 객체

메소드	설명
<code>getParameter(String name)</code>	지정한 이름의 파라미터를 구한다. 지정한 이름의 파라미터가 여러 개 있을 경우에는 첫 번째 파라미터의 값을 구한다.
<code>getParameterNames()</code>	모든 파라미터의 이름을 구한다.
<code>getParameterValues(String name)</code>	지정한 이름의 파라미터가 여러 개 있을 경우 사용하며 지정한 이름을 가진 파라미터의 모든 값을 <code>String[]</code> 으로 구한다.

`request.getParameter("id")`

아이디 : pinksung  
나 이 : 20  
전송

`request.getParameter("age")`

당신이 입력한 정보입니다.  
아이디 : pinksung  
나 이 : 20  
[다시](#)

## JSP 내장 객체

### ❖ 예제

파일 이름	역할
02_researchForm.jsp	설문 조사를 위한 폼 양식이다. [전송] 버튼을 클릭하면 서블릿으로 입력된 값이 전송되고 JSP 페이지(02_research.jsp)에서 입력받은 값을 받아 처리한다.
02_research.jsp	02_researchForm.jsp의 입력 양식의 내용을 읽어와 처리하는 JSP 파일이다.

## JSP 내장 객체

### ❖ 예제

설문 조사를 위한 폼 양식  
02\_researchForm.jsp

입력 양식의 내용을 읽어 와서  
처리하는 JSP  
02\_research.jsp

The first screenshot shows a web browser window titled '설문 조사' (Survey) at the URL 'localhost:8181/web-study-04/02\_researchForm.jsp'. The form contains the following fields and values:

- 이름 : 성윤정 (Name)
- 성별 : ☐ 남자 ☒ 여자 (Gender)
- 좋아하는 계절: ☐ 봄 ☒ 여름 ☐ 가을 ☒ 겨울 (Seasons)

The second screenshot shows a web browser window titled '설문 조사 결과' (Survey Results) at the URL 'localhost:8181/web-study-04/02\_research.jsp'. It displays the results of the survey:

- 이름 : 성윤정
- 성별 : 여자
- 당신이 좋아하는 계절은 여름 겨울입니다.

Arrows indicate the mapping of form data to JSP code:

- `request.getParameter("name")` maps to the name field.
- `request.getParameter("gender")` maps to the gender field.
- `request.getParameterValues("season")` maps to the season field.



# JSP 내장 객체

## ❖ 02\_researchForm.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
:
<body>
    <h2>설문 조사</h2>
    <form action="02_research.jsp" method="post">
        <table>
            <tr>
                <td>이름 :</td>
                <td><input type="text" name="name" size="20"></td>
            </tr>
            <tr>
                <td>성별 :</td>
                <td>
                    <input type="radio" name="gender" value="male"
                        checked="checked"> 남자
                    <input type="radio" name="gender"
                        value="female"> 여자
                </td>
            </tr>
        </table>
    </form>
</body>
```

# JSP 내장 객체

## ❖ 02\_researchForm.jsp

```
<tr>
  <td>좋아하는 계절:</td>
  <td>
    <input type="checkbox" name="season" value="1"> 봄
    <input type="checkbox" name="season" value="2"
      checked="checked"> 여름
    <input type="checkbox" name="season" value="3"> 가을
    <input type="checkbox" name="season" value="4"> 겨울
  </td>
</tr>
<tr align="center">
  <td><input type="submit" value="전송"></td>
  <td><input type="reset" value="취소"></td>
</tr>
</table>
</form>
</body>
</html>
```

# JSP 내장 객체

---

## ❖ 02\_research.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
:
<body>
    <h2>설문 조사 결과</h2>
    <%
        request.setCharacterEncoding("UTF-8");

        String name = request.getParameter("name");
        out.println("이름 : <b>" + name + "</b><br>");

        String gender = request.getParameter("gender");
        out.println("성별 : ");
        if (gender.equals("male")) {
            out.println("<b>남자</b><br>");
        } else {
            out.println("<b>여자</b><br>");
        }
    %>
```

# JSP 내장 객체

## ❖ 02\_research.jsp

```
String seasonArr[] = request.getParameterValues("season");
out.println("당신이 좋아하는 계절은 ");
for (String season : seasonArr) {
    int n = Integer.parseInt(season);
    switch (n) {
        case 1: out.println("<b> 봄 </b>입니다.");
                break;
        case 2: out.println("<b> 여름 </b>입니다.");
                break;
        case 3: out.println("<b> 가을 </b>입니다.");
                break;
        case 4: out.println(" 겨울 </b>입니다.");
                break;
    }
}
%>
<br>
<b><a href='javascript:history.go(-1) '>다시</a></b>
</body>
</html>
```

# JSP 내장 객체

---

## ❖ response 내장 객체

- 클라이언트에 대한 응답 처리를 하는 객체
- 리다이렉션
  - `response.sendRedirect(url)`

```
response.sendRedirect("http://www.roadbook.co.kr");
```

이동할 페이지 지정

## ❖ 03\_redirect.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Insert title here</title>
</head>
<body>
로드북 홈페이지로 이동
<%
    response.sendRedirect("http://www.roadbook.co.kr");
%>
</body>
</html>
```

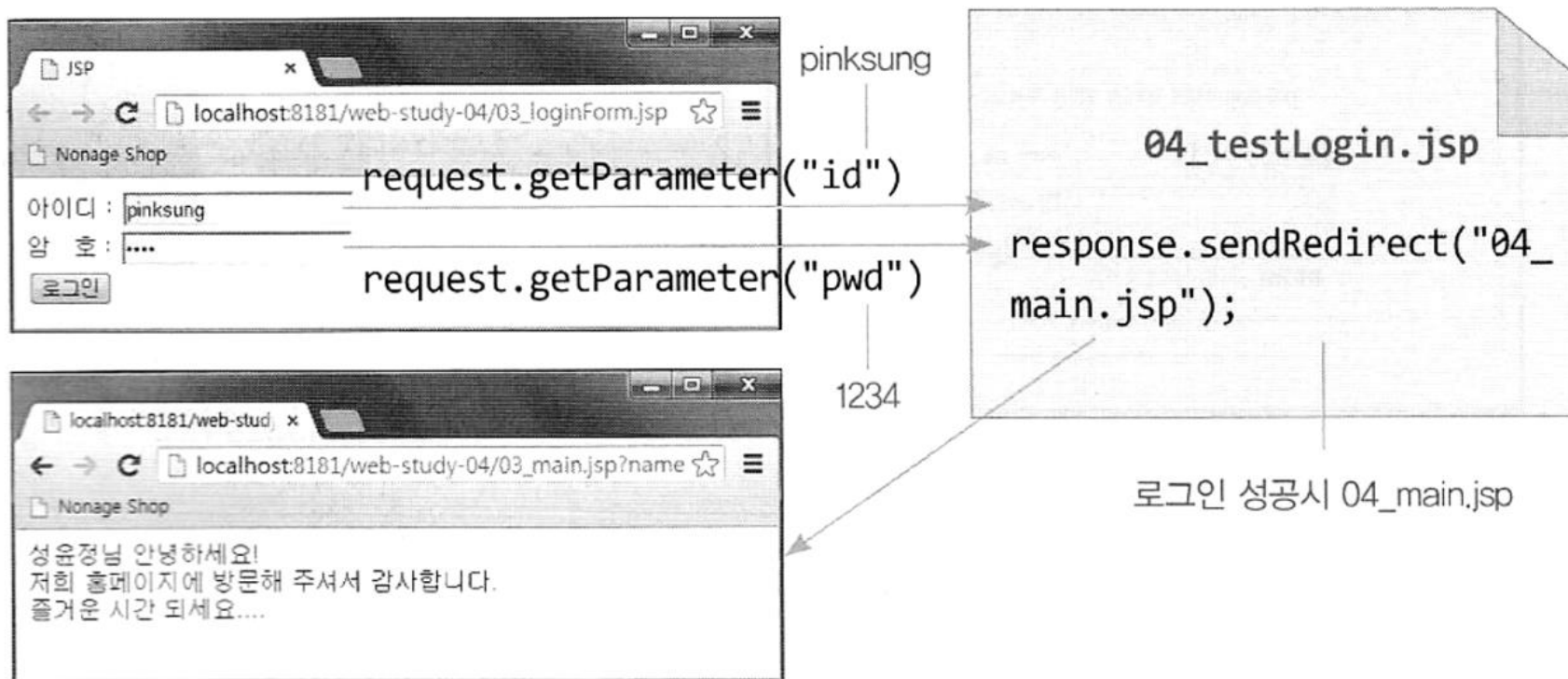
## JSP 내장 객체

### ❖ 예제

파일	설명
04_loginForm.jsp	클라이언트가 ID와 암호를 입력할 수 있는 화면을 제공하기 위한 페이지
04_testLogin.jsp	로그인 화면에서 제공한 정보를 가져와 회원 인증 절차를 거치는 페이지
04_main.jsp	로그인 인증을 받은 회원이 다양한 정보를 제공받을 수 있는 페이지 제공

# JSP 내장 객체

## ❖ 예제





### ❖ 04\_loginForm.jsp

```
<form method="post" action="04_testLogin.jsp">
  <label for="userid"> 아이디 : </label>
  <input type="text" name="id" id="userid"><br>
  <label for="userpwd"> 암호 : </label>
  <input type="password" name="pwd" id="userpwd"><br>
  <input type="submit" value="로그인">
</form>
```

## JSP 내장 객체

---

### ❖ 04\_testLogin.jsp

```
<%  
    String id="pinksung";  
    String pwd="1234";  
    String name="성윤정";  
  
    request.setCharacterEncoding("UTF-8");  
    if(id.equals(request.getParameter("id")) &&  
        pwd.equals(request.getParameter("pwd")) ){  
        response.sendRedirect("04_main.jsp?name="+  
                               URLEncoder.encode(name, "UTF-8"));  
    }  
    else{  
        response.sendRedirect("04_loginForm.jsp");  
    }  
%>
```

## JSP 내장 객체

---

### ❖ 04\_main.jsp

```
<body>
    <%= request.getParameter("name") %>님 안녕하세요!
    <br>
    저희 홈페이지에 방문해 주셔서 감사합니다.<br>
    즐거운 시간 되세요.....<br>
</body>
```

## JSP 내장 객체

---

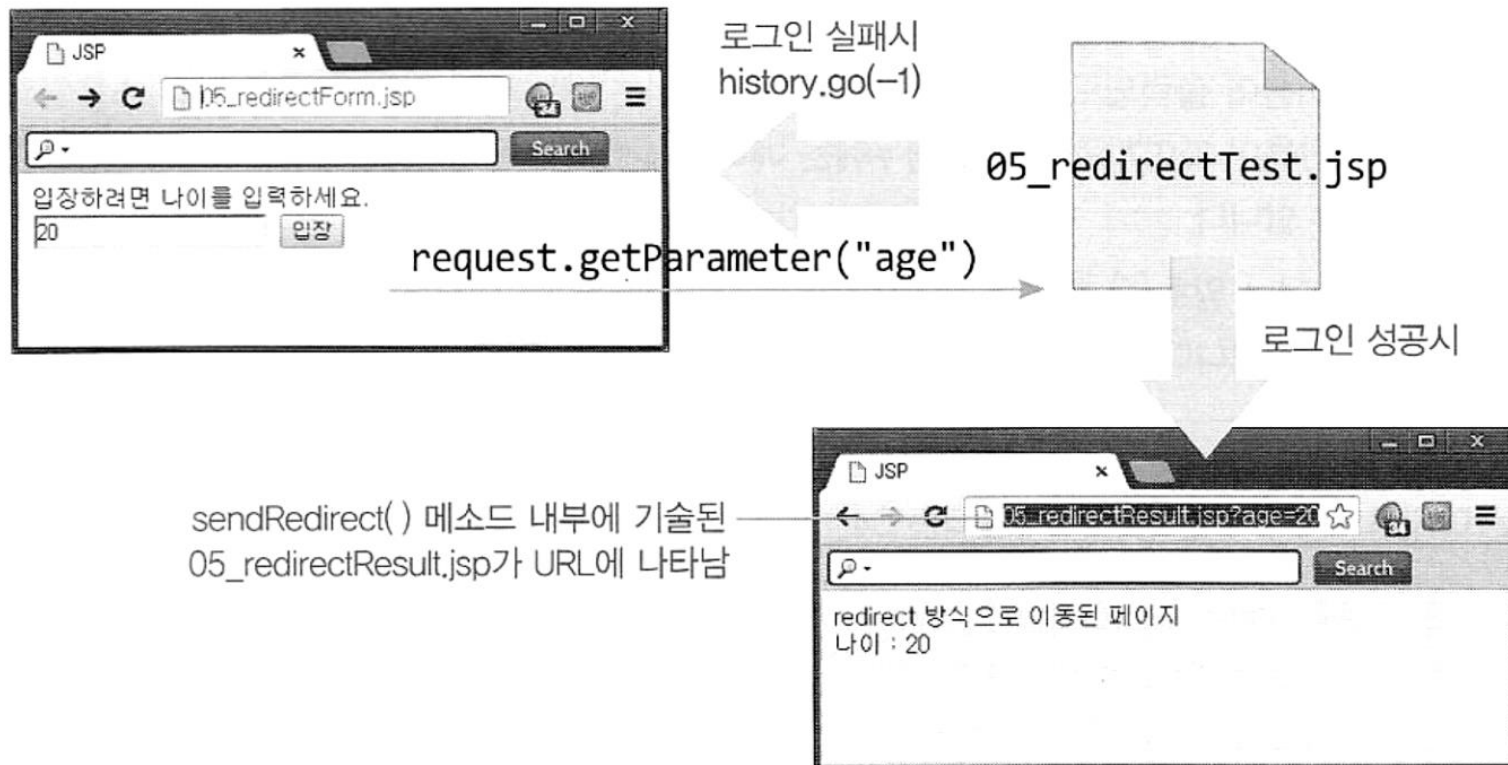
### ❖ 다른 웹페이지로의 이동

- 리다이렉트 방식
- 포워드 방식

## JSP 내장 객체

### ❖ 다른 웹페이지로의 이동

- 리다이렉트 방식
  - `response.sendRedirect()`
  - 브라우저가 url을 변경하여 재요청



# JSP 내장 객체

---

## ❖ 다른 웹페이지로의 이동

### ○ 리다이렉트 방식

- 리다이렉트된 페이지에는 새로운 request 객체가 생성
- 정보는 URL에 쿼리스트링을 통해서 전달

```
response.sendRedirect("04_main.jsp?age=" + 20 );
```

파라미터 이름    파라미터 값

## JSP 내장 객체

---

### ❖ 다른 웹페이지로의 이동

- 포워드 방식
  - RequestDispatcher의 forward() 메서드로 이동
  - 브라우저가 아닌 서버에서 페이지 이동
    - 브라우저는 요청 URL의 처리 페이지 이동을 감지 못함

```
RequestDispatcher dispatcher=request.getRequestDispatcher("05_forwardResult.jsp");  
dispatcher.forward(request, response);
```

이동할 페이지 지정

## JSP 내장 객체

### ❖ 다른 웹페이지로의 이동

#### ○ 포워드 방식





## JSP 내장 객체

### ❖ 다른 웹페이지로의 이동

#### ○ 포워드 방식

- 페이지가 이동해도 request 객체는 유지
- 정보를 request 객체에 저장하여 전달 가능

```
request.setAttribute( "age" , 20 );
```

속성 이름      속성 값

- 정보 추출

```
int age =(Integer)request.getAttribute( "age" );
```

속성 값      속성 이름

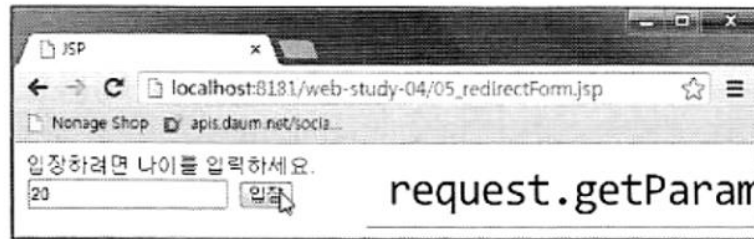
- `getAttribute()`는 Object 타입으로 리턴 → 캐스팅 필요

## JSP 내장 객체

### ❖ 예제

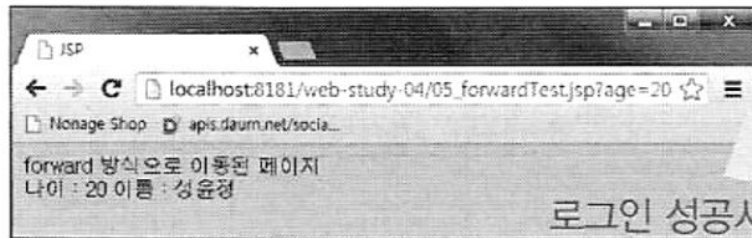
파일	설명
05_forwardForm.jsp	나이를 입력할 수 있는 화면을 제공하기 위한 페이지
05_forwardTest.jsp	입력받은 나이가 19세 미만이면 페이지 이동이 불가능하고 나이가 20세 이상이면 05_forwardResult.jsp로 이동시킴
05_forwardResult.jsp	전송한 나이와 이름 출력

## ❖ 예제



로그인 실패시

`request.getParameter("age")`



로그인 성공시

05\_forwardResult.jsp

05\_forwardTest.jsp

```
RequestDispatcher dispatcher  
=request.  
getRequestDispatcher("05_  
forwardResult.jsp");  
dispatcher.forward(request,  
response);
```

## ❖ 05\_forwardForm.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html >
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-
8">
    <title>Insert title here</title>
</head>
<body>
    입장하려면 나이를 입력하세요. <br>
    <form action="05_forwardTest.jsp">
        <input type="text" name="age">
        <input type="submit" value="입장">
    </form>
</body>
</html>
```

## JSP 내장 객체

---

### ❖ 05\_forwardTest.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
    int age=Integer.parseInt(request.getParameter("age"));
    if(age<=19){
%>
<script type="text/javascript">
    alert("19세 미만이므로 입장 불가능")
    history.go(-1)
</script>
<%
    }else{
        request.setAttribute("name", "성윤정");
        RequestDispatcher dispatcher

        =request.getRequestDispatcher("05_forwardResult.jsp");
        dispatcher.forward(request, response);
    }
%>
```

## JSP 내장 객체

### ❖ history 객체

속성	내용
length	브라우저의 history 목록에 저장된 URL의 갯수
메소드	내용
back()	한 단계 전 URL로 이동
forward()	한 단계 뒤 URL로 이동
go()	지정된 단계의 URL로 이동

## JSP 내장 객체

---

### ❖ 05\_forwardResult.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
    String age=request.getParameter("age");
    String name=(String)request.getAttribute("name");
%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Insert title here</title>
</head>
    <body bgcolor=pink>
        forward 방식으로 이동된 페이지 <br>
        나이 : <%=age %>
        이름 : <%=name %>
    </body>
</html>
```

# JSP 내장 객체

## ❖ JSP 내장객체

### ○ application

- 하나의 웹 애플리케이션을 관리하고 웹 애플리케이션 안에서의 자원 공유 제공
- 정보 저장 가능
  - `setAttribute()/getAttribute()` 제공
  - 애플리케이션 전체에서 정보 유지(공유)
- 실체는 `ServletContext` 인터페이스 객체

메소드	설명
<code>getServerInfo()</code>	컨테이너의 이름과 버전을 반환합니다.
<code>getContextPath()</code>	웹 애플리케이션의 URL 경로 중 컨텍스트 패스명을 반환합니다.
<code>getRealPath()</code>	JSP의 실제 경로를 반환한다.
<code>getMimeType(filename)</code>	지정된 파일의 MIME 타입을 반환합니다.
<code>log(message)</code>	지정된 message의 로그를 저장합니다.



## JSP 내장 객체

❖ 예제 : application의 실제 경로 알아보기

❖ 06\_application.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html >
:
<body>
<%
    String appPath = application.getContextPath();
    String filePath = application.getRealPath("06_application.jsp");
    String serverInfo = application.getServerInfo();
%>
    웹 애플리케이션의 컨텍스트 패스명<br>
    <b><%=appPath%></b><hr>
    웹 애플리케이션의 파일 경로명<br>
    <b><%=filePath%></b><br>
    컨테이너의 이름과 버전 번호<br>
    <b><%=serverInfo%></b><br>
</body>
</html>
```

## JSP 내장 객체

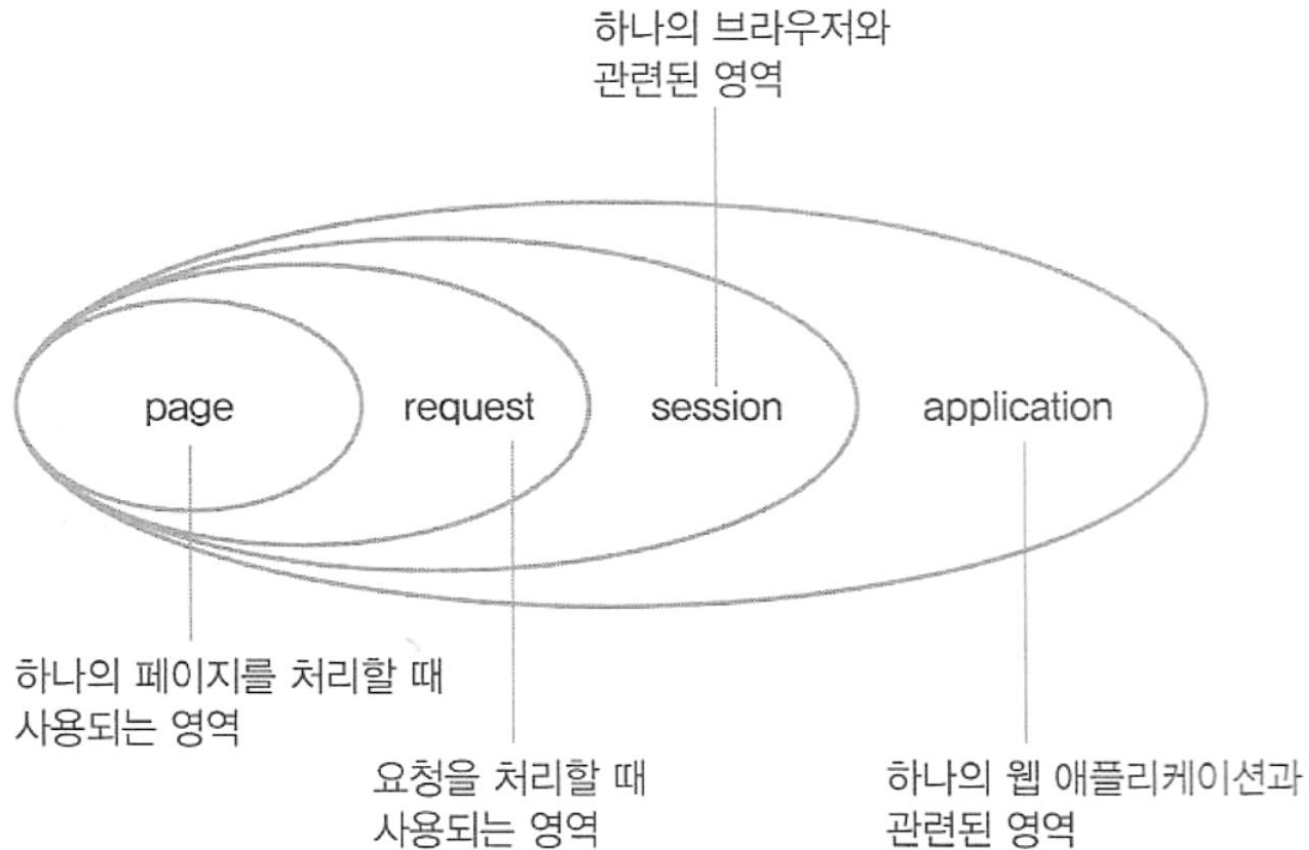
### ❖ 내장 객체의 영역(scope)

- 내장 객체의 영역은 객체의 유효기간이라고도 불림
- 영역의 종류 : page, request, session, application

영역	설명
page	하나의 JSP 페이지를 처리할 때 사용되는 영역
request	하나의 요청을 처리할 때 사용되는 영역
session	하나의 브라우저와 관련된 영역
application	하나의 웹 애플리케이션과 관련된 영역

# JSP 내장 객체

## ❖ 내장 객체의 영역(scope)



# JSP 내장 객체

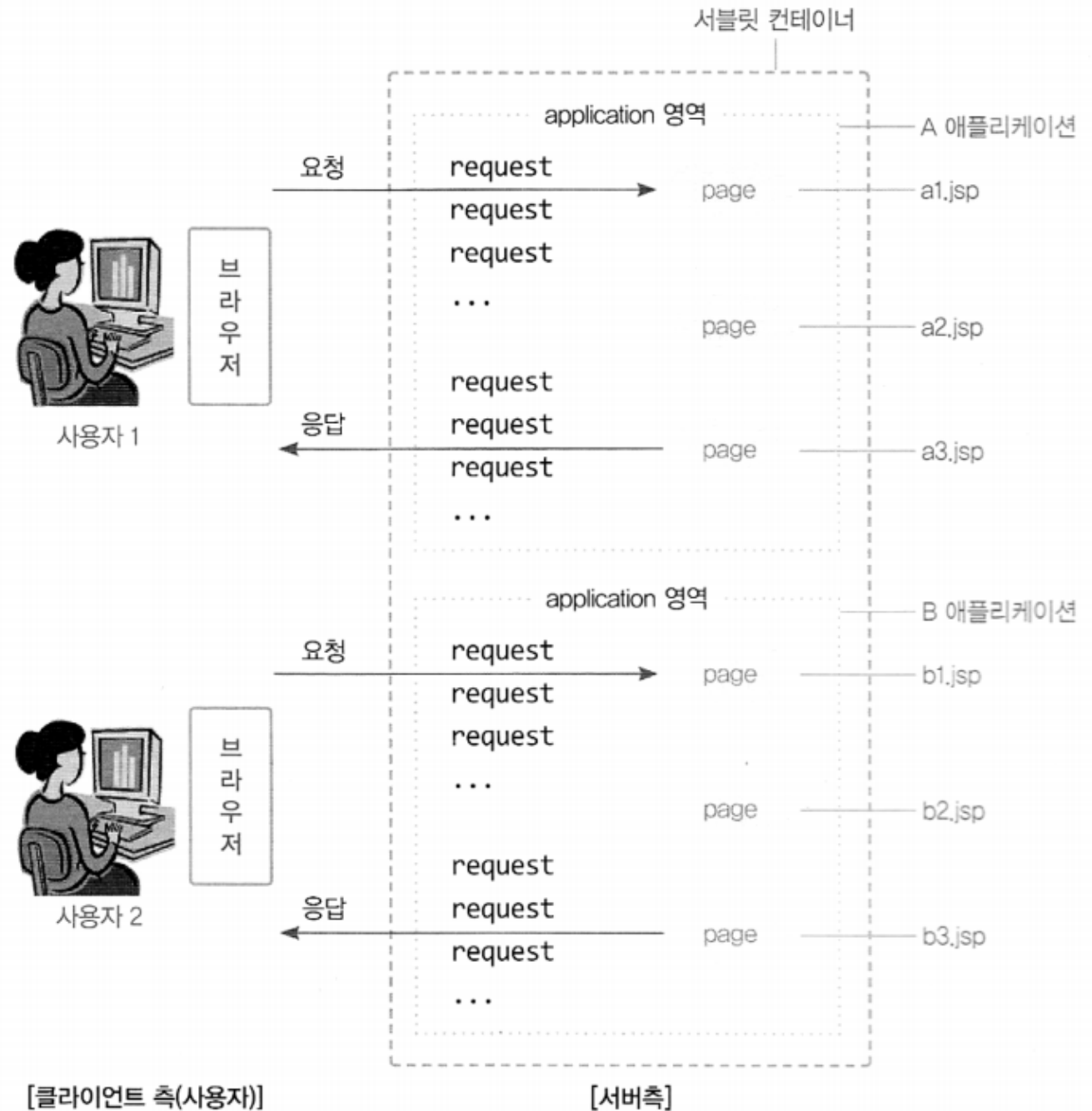
---

## ❖ 내장 객체의 영역(scope)

- page 영역
  - 한 번의 클라이언트 요청에 하나의 JSP 페이지 범위
  - `pageContext` 내장 객체에 할당
- request 영역
  - 요청과 관련된 영역
  - `request` 내장 객체에 할당
  - 요청이 바뀔때마다 새로운 객체 생성
  - `forward` 통해 페이지를 이동해도 유지
  - `redirect`로 페이지 이동시 요청이 달라지므로 다른 `request` 영역이 됨
- session 영역
  - 웹 브라우저를 닫기 전까지 페이지를 이동하더라도 사용자 정보를 서버에서 유지
  - 요청이 달라져도 정보 유지
- application 영역
  - 웹 애플리케이션과 관련된 전체 영역
  - 서버에서 웹 애플리케이션 기동에서 부터 종료될 때까지 유지
  - 전체 웹 페이지에서 공유

# JSP 내장 객체

## ❖ 내장 객체의 영역 (scope)



## JSP 내장 객체

### ❖ 내장 객체의 영역(scope)

- 관련 메서드

메소드	설명
setAttribute(name, value)	이름(name)에 값(value)을 설정합니다.
getAttribute(name)	매개 변수로 준 이름에 설정된 값을 얻어냅니다.
getAttributeNames()	현재 객체에 관련된 모든 속성의 이름을 뽑아냅니다.
removeAttribute(name)	매개 변수로 준 이름에 설정된 값을 제거합니다.

## JSP 내장 객체

### ❖ 예제

파일 이름	역할
07_firstPage.jsp	pageContext, request, session, application 객체에 새로운 속성을 추가한 후 07_secondPage.jsp 페이지로 포워딩한다. 이 페이지에서 설정한 속성 값은 브라우저에 출력할 수 없기 때문에 콘솔 창에 출력해서 확인합니다.
07_secondPage.jsp	07_firstPage.jsp에서 설정한 속성 값을 브라우저에 출력합니다.
07_thirdPage.jsp	07_secondPage.jsp의 [또 다른 페이지]를 클릭하여 리다이렉트 방식으로 요청 되는 페이지로 07_firstPage.jsp에서 설정한 속성 값을 브라우저에 출력합니다.

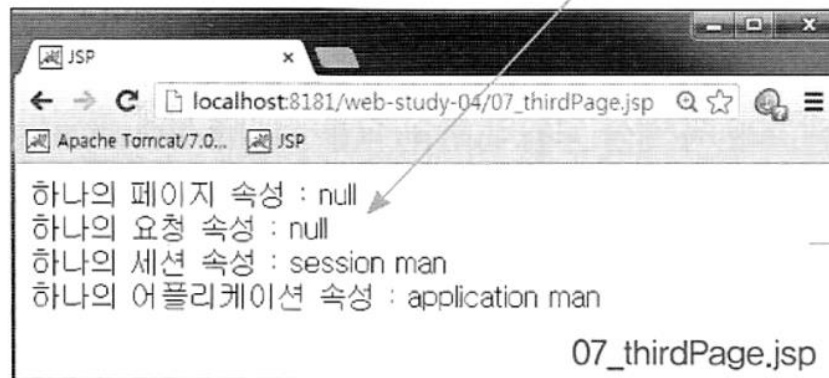
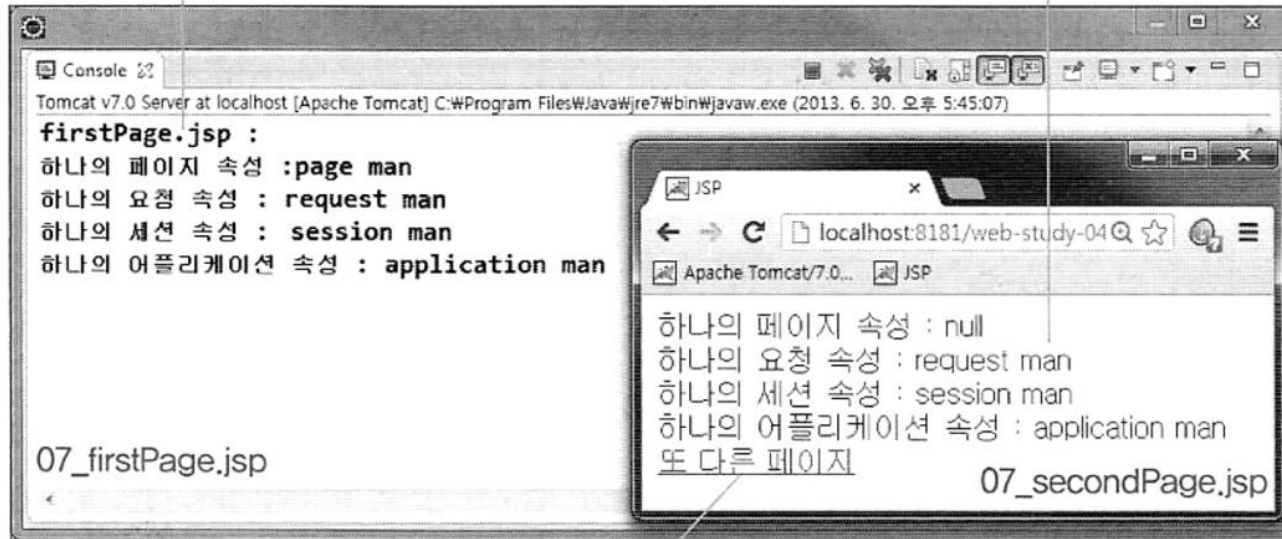


# JSP 내장 객체

## ❖ 예제

pageContext, request, session, application  
객체에 설정한 속성값을 콘솔창에 출력하여  
확인한다.

07\_firstPage.jsp에서 설정한 속성값을  
브라우저에 출력한다.



07\_firstPage.jsp에서 설정한 속성값을  
브라우저에 출력한다.



# JSP 내장 객체

## ❖ 07\_firstPage.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
    pageContext.setAttribute("name", "page man");
    request.setAttribute("name", "request man");
    session.setAttribute("name", "session man");
    application.setAttribute("name", "application man");

    System.out.println("firstPage.jsp : ");
    System.out.println("하나의 페이지 속성 : "+
        pageContext.getAttribute("name"));
    System.out.println("하나의 요청 속성 : " + request.getAttribute("name"));
    System.out.println("하나의 세션 속성 : " + session.getAttribute("name"));
    System.out.println("하나의 애플리케이션 속성 : " +
        application.getAttribute("name"));

    request.getRequestDispatcher("07_secondPage.jsp").forward(request,
                                                                    response);
%>
```

# JSP 내장 객체

## ❖ 07\_secondPage.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Insert title here</title>
</head>
<body>
    하나의 페이지 속성 : <%= pageContext.getAttribute("name") %> <br>
    하나의 요청 속성 : <%= request.getAttribute("name") %> <br>
    하나의 세션 속성 : <%= session.getAttribute("name") %> <br>
    하나의 애플리케이션 속성 : <%= application.getAttribute("name") %> <br>

    <a href="07_thirdPage.jsp"> 또 다른 페이지 </a>
</body>
</html>
```

# JSP 내장 객체

---

## ❖ 07\_thirdPage.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Insert title here</title>
</head>
<body>
    하나의 페이지 속성 : <%= pageContext.getAttribute("name") %> <br>
    하나의 요청 속성 : <%= request.getAttribute("name") %> <br>
    하나의 세션 속성 : <%= session.getAttribute("name") %> <br>
    하나의 애플리케이션 속성 : <%= application.getAttribute("name") %> <br>

    <a href="07_thirdPage.jsp"> 또 다른 페이지 </a>
</body>
</html>
```

## 액션 태그

### ❖ 액션 태그

- 액션(기능)을 수행하는 태그
  - 내장 객체를 통해 자바 코드로 작성할 수 있는 기능을 태그로 표현한 것

태그의 종류	설명
<jsp:forward>	다른 사이트로 이동할 때 사용 페이지의 흐름을 제어할 때 사용
<jsp:include>	정적 혹은 동적인 자원을 현재 페이지의 내용에 포함시킨다. 페이지를 모듈화할 때 사용
<jsp:param>	<jsp:forward>, <jsp:include>, <jsp:plugin>과 같이 사용되어 파라미터를 추가할 때 사용
<jsp:useBean>	빈(Been)을 생성하고 사용하기 위한 환경을 정의하는 액션 태그
<jsp:setProperty>	액션은 빈에서 속성 값을 할당
<jsp:getProperty>	액션은 빈에서 속성 값을 얻어올 때 사용

## 액션 태그

---

### ❖ 액션 태그

#### ○ 형식

```
<jsp:태그명 속성="값">내용</jsp:태그명>  
  <jsp:forward page="yellow02.jsp">  
    <jsp:param name = "url" value="red02.jsp"/>  
    <jsp:param name = "news" value="Happy New Year!"/>  
  </jsp:forward>
```

- 내용이 없는 경우 단일 태그 표시 가능

```
<jsp:태그명 속성="값" />  
  
<jsp:forward page="yellow.jsp" />
```

## 액션 태그

---

### ❖ <jsp:forward> 액션 태그

- 페이지 포워딩
  - page 속성에 포워딩할 url 기술

```
<jsp:forward page="relativeURLspec"/>
```

```
RequestDispatcher dispatcher =  
    getServletContext().getRequestDispatcher("url");  
dispatcher.forward(request, response);
```

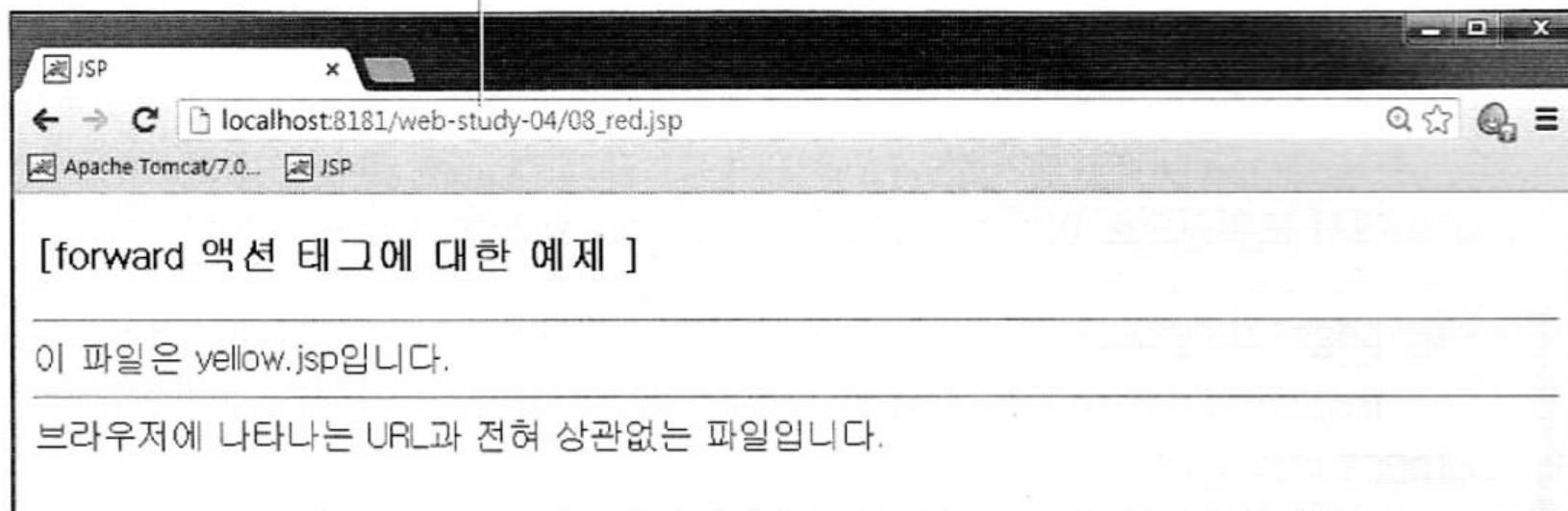
- 포워드로 이동하여도 웹 브라우저의 요청 URL은 기존 URL 유지

## 액션 태그

### ❖ 예제

파일	설명
08_red.jsp	브라우저로 로드할 파일. 웹 문서의 배경색을 빨간색으로 지정한다. 이 파일에 <jsp:forward> 태그로 yellow.jsp 파일을 요청한다.
08_yellow.jsp	red.jsp에 의해 요청되는 페이지이다. 웹 문서의 배경색을 노란색으로 지정한다.

08\_red.jsp를 요청하였지만 브라우저에 나타나 내용은 yellow.jsp 파일에 기술된 내용이다. 하지만, 주소 입력란에 나타난 url은 여전히 08\_red.jsp이다.



## 액션 태그

---

### ❖ 08\_red.jsp

```
<body bgcolor="red">  
    이 파일은 red.jsp입니다. <br>  
    브라우저에 배경색이 빨간색으로 나타날까요?<br>  
    노란색으로 나타날까요?<hr>  
    forward 액션 태그가 실행되면 이 페이지의 내용은 출력되지 않습니다.  
<br>  
    <jsp:forward page="08_yellow.jsp" />  
</body>  
</html>
```



### ❖ 08\_yellow.jsp

```
<body bgcolor="yellow">

    <h3> [forward 액션 태그에 대한 예제]</h3>
    <hr>
    이 파일은 yellow.jsp입니다.
    <br>
    <hr>
    브라우저에 나타나는 URL과 전혀 상관없는 파일입니다.

</body>
</html>
```

## 액션 태그

---

### ❖ <jsp:param> 액션 태그

- <jsp:forward> 액션태그로 이동하는 페이지에 정보를 추가
- 단독 사용 불가
  - <jsp:include>, <jsp:forward>의 하위 태그로 이용

```
<jsp:param name="파라미터명" value="파라미터값" />
```

```
<jsp:forward page="main.jsp">  
  <jsp:param name="username" value="Bae Su Ji" />  
</jsp:forward>
```

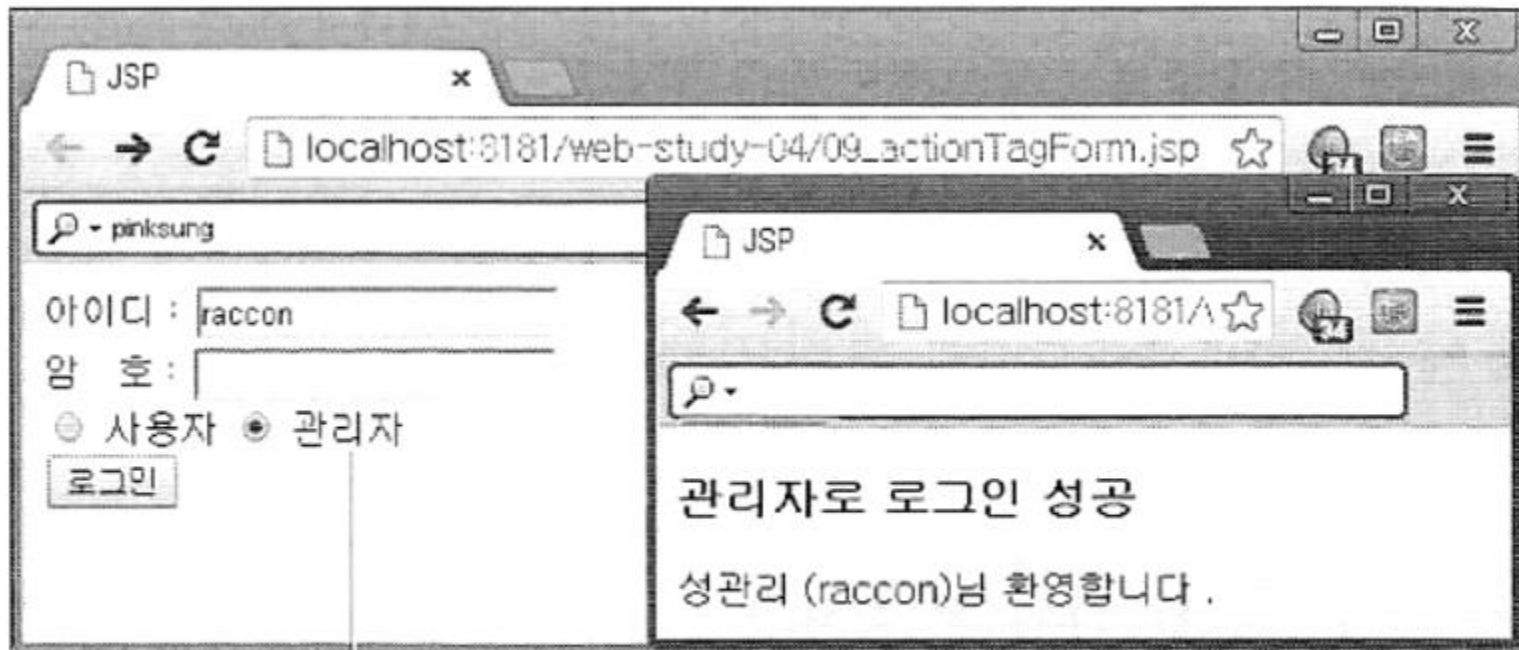
## 액션 태그

### ❖ 예제

파일	설명
09_actionTagForm.jsp	라디오 버튼을 두어 사용자로 로그인할지 관리자로 로그인할지를 선택하도록 한다.
09_actionTagTest.jsp	〈jsp:forward〉 태그와 〈jsp:param〉 태그의 관계를 이해하기 위한 간단한 JSP 파일이다. 조건에 따라 이동할 페이지를 달리 한다.
09_userMain.jsp	사용자 페이지이다.
09_managerMain.jsp	관리자 페이지이다.

## 액션 태그

### ❖ 예제



관리자로 로그인

### ❖ 09\_actionTagForm.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Insert title here</title>
</head>
<body>
    <form action="09_actionTagTest.jsp">
        아이디 : <input type="text" name="userID"><br>
        암 &nbsp;호 : <input type="password" name="userPwd"><br>
        <input type="radio" name="loginCheck" value="user"
            checked="checked"> 사용자
        <input type="radio" name="loginCheck" value="manager"> 관리자 <br>
        <input type="submit" value="로그인">
    </form>
</body>
</html>
```

### ❖ 09\_actionTagTest.jsp

```
<%@page import="java.net.URLEncoder"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
    String id="pinksung";
    String pwd="1234";
    String name="성윤정";

    request.setCharacterEncoding("UTF-8");

    if(id.equals(request.getParameter("id")) &&
        pwd.equals(request.getParameter("pwd"))) {
        response.sendRedirect("04_main.jsp?name="+
            URLEncoder.encode(name, "UTF-8"));
    }
    else{
        response.sendRedirect("04_loginForm.jsp");
    }
%>
```

## 액션 태그

### ❖ 09\_userMain.jsp

```
<%@page import="java.net.URLEncoder"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
String userID=request.getParameter("userID");
String userPwd=request.getParameter("userPwd");
String loginCheck=request.getParameter("loginCheck");
if(loginCheck.equals("user")){
%>
    <jsp:forward page="09_userMain.jsp">
        <jsp:param value='<%=URLEncoder.encode("전고객", "UTF-8")%>'
            name="userName"/>
    </jsp:forward>
<%}else{ %>
    <jsp:forward page="09_managerMain.jsp">
        <jsp:param value='<%=URLEncoder.encode("성관리", "UTF-8")%>'
            name="userName"/>
    </jsp:forward>
<%} %>
```

## 액션 태그

---

### ❖ 09\_userMain.jsp

```
<%@page import="java.net.URLDecoder"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html >
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Insert title here</title>
</head>
<body bgcolor="pink">
    <h3>사용자로 로그인 성공 </h3>
    <%=URLDecoder.decode(request.getParameter("userName"), "UTF-8")%>
    (<%=request.getParameter("userID")%>)님 환영합니다 .
</body>
</html>
```



## 액션 태그

---

### ❖ 09\_managerMain.jsp

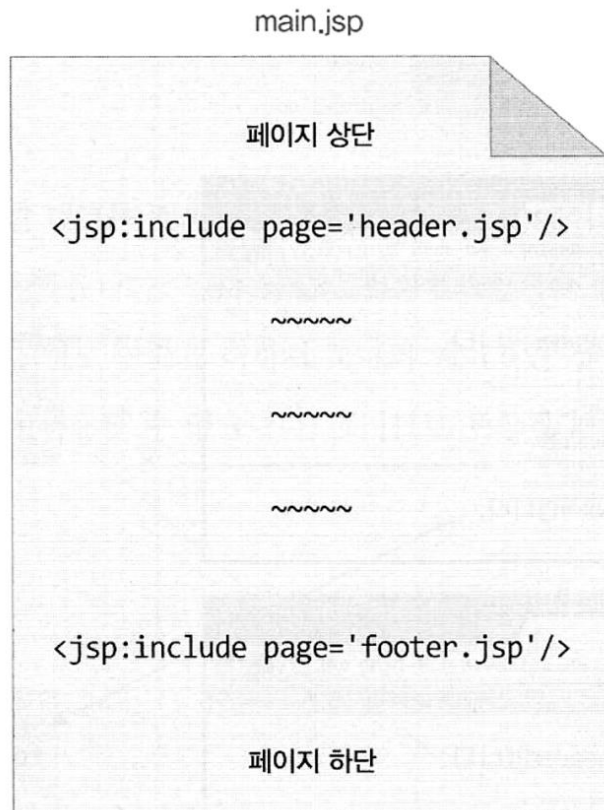
```
<%@page import="java.net.URLDecoder"%>
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-
8">
    <title>Insert title here</title>
</head>
<body bgcolor="yellow">
    <h3>관리자로 로그인 성공 </h3>
    <%=URLDecoder.decode(request.getParameter("userName"), "UTF-
8")%>
    <%=request.getParameter("userID") %>님 환영합니다 .
</body>
</html>
```

## 액션 태그

### ❖ <jsp:include> 액션 태그

- 현재 페이지에 다른 페이지의 내용을 동적으로 포함시킬 때 사용

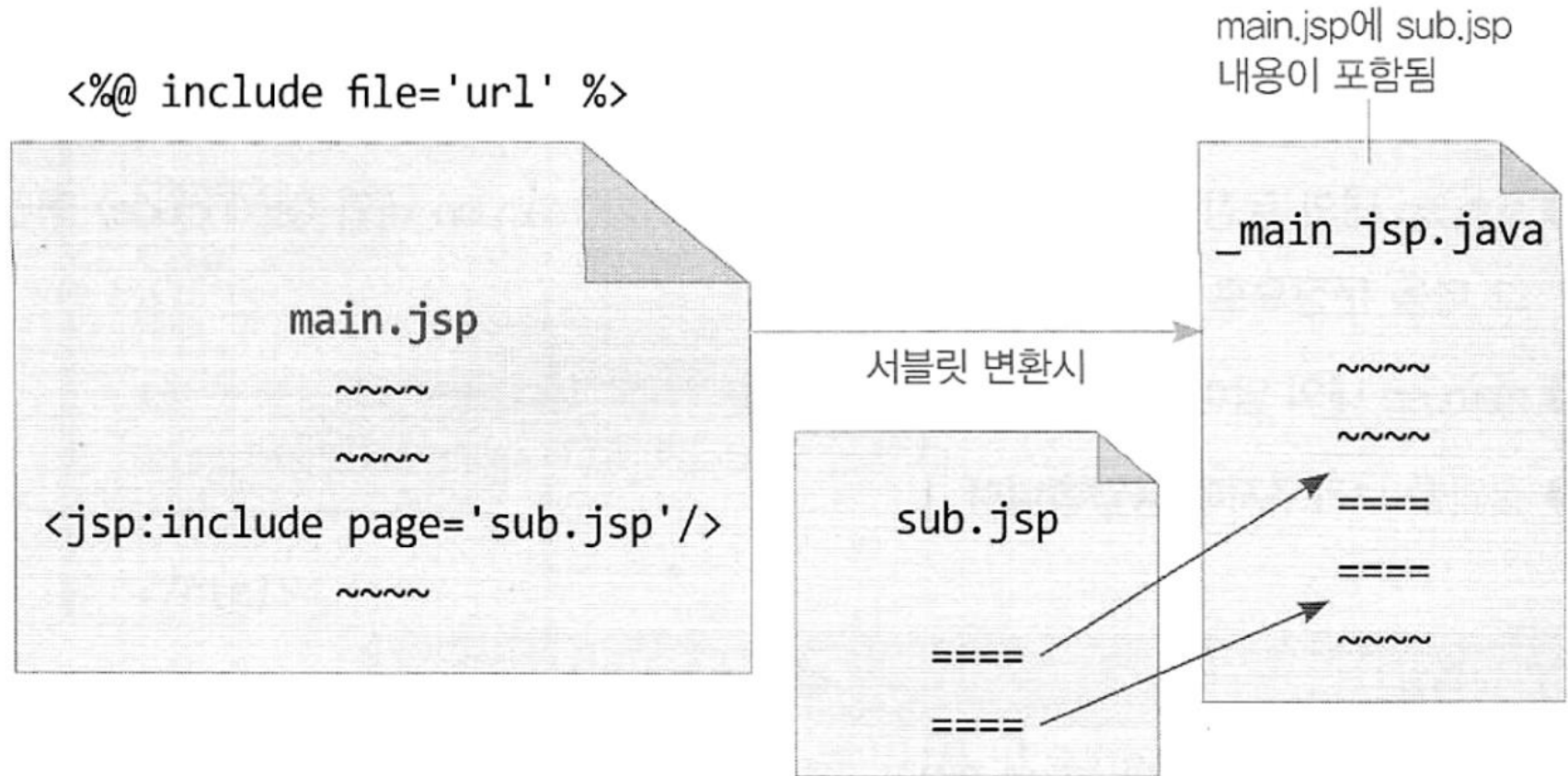
```
<jsp:include page="페이지url" flush="true/false"></jsp:include>
```



## 액션 태그

### ❖ <jsp:include> 액션 태그

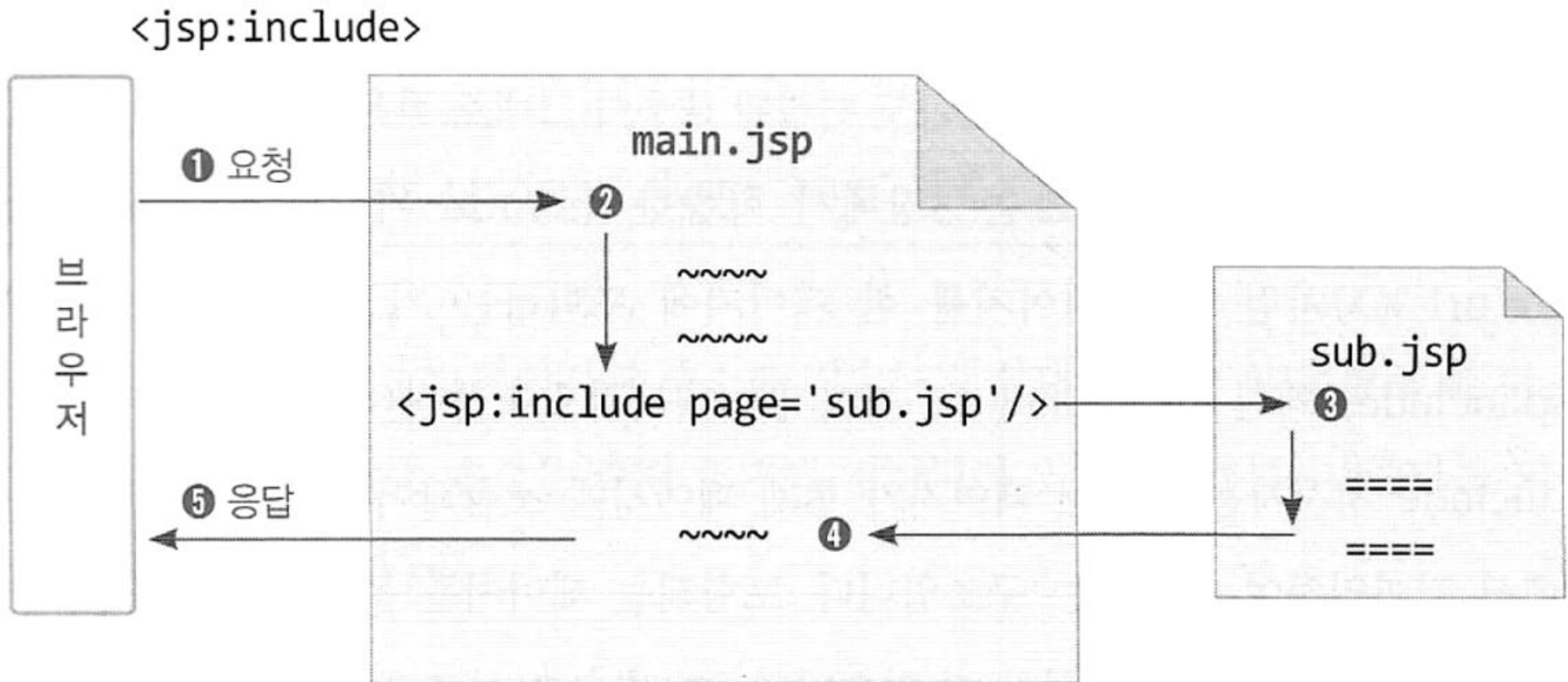
- o include 디렉티브와의 차이



## 액션 태그

### ❖ <jsp:include> 액션 태그

- include 디렉티브와의 차이



## 액션 태그

### ❖ 예제

파일	설명
10_header.jsp	화면 상단에 출력할 공통 페이지이다.
10_footer.jsp	화면 하단에 출력할 공통 페이지이다.
10_main.jsp	메인 페이지로 헤더와 풋터를 포함하고 있다.
10_sub.jsp	서브 페이지로 헤더와 풋터를 포함하고 있다.

## 액션 태그

---

### ❖ 10\_header.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"  
    pageEncoding="UTF-8"%>  
<h3> 이부분은header입니다.</h3>  
<hr>
```

### ❖ 10\_footer.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<hr>
<h4> 이부분은footer입니다.</h4>
```

## 액션 태그

---

### ❖ 10\_main.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html >
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-
8">
    <title>Insert title here</title>
</head>
<body>
    <jsp:include page="10_header.jsp"/>
    메인 화면입니다. <br>
    <a href="10_sub.jsp">서브 페이지로 이동</a>
    <jsp:include page="10_footer.jsp"/>
</body>
</html>
```



## 액션 태그

---

### ❖ 10\_sub.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html >
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-
8">
    <title>Insert title here</title>
</head>
<body>
    <jsp:include page="10_header.jsp"/>
    서버 화면입니다. <br>
    <a href="10_main.jsp">메인 페이지로 되돌아가기</a>
    <jsp:include page="10_footer.jsp"/>
</body>
</html>
```