



# 페이지 모듈화와 요청 흐름 제어

# Contents

---

- ❖ 1. include 디렉티브
- ❖ 2. <jsp:include> 액션 태그와 모듈화
- ❖ 3. <jsp:param> 액션 태그의 기능과 사용법
- ❖ 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

# 1. include 디렉티브(<%@ include ... %>)

---

## ❖ include 디렉티브의 처리 방식과 사용법

- include 디렉티브는 지정한 페이지를 현재 위치에 포함
- 사용방법

```
<%@ include file ="포함파일" %>
```

# 1. include 디렉티브(<%@ include ... %>)

## ❖ include 디렉티브의 처리 방식과 사용법

- include 디렉티브는 지정한 페이지를 현재 위치에 포함

- 처리 방식

main.jsp

코드1

<%@ include file="sub.jsp" %>

코드2

sub.jsp

코드sub1

main.jsp의 ----- 디렉티브 위치에  
포함할 페이지의 코드를 삽입시킨 결과를  
자바코드로 작성한다.

최종 자바코드

코드1

코드 sub1

코드2

자바코드를 서블릿 클래스로 컴파일한다.

서블릿 클래스

# 1. include 디렉티브

## ❖ include 디렉티브 예제

### ▪ includeMain.jsp

```
1 <body>
2   <h1>includeMain.jsp 페이지</h1>
3   <%
4     int number = 10;
5   %
6   <%@ include file="includeSub.jsp" %>
7   <h3>공통 변수 dataFolder = "<%=dataFolder %>"</h3>
8
9 </body>
```

includeMain.jsp에서 실행

### ▪ includeSub.jsp

```
1 <body>
2   <h3>includeSub.jsp에서 읽은 number : <%=number %></h3>
3   <p>
4   <%
5   %
6     String dataFolder = "c:\\\\data";
7   %
8 </body>
9 </html>
10
11
```

# 1. include 디렉티브

## ❖ include 디렉티브의 활용 및 사용 목적

- 모든 JSP 페이지에서 사용하는 변수 지정
- 저작권 표시와 같이 모든 페이지에서 중복되는 간단한 문장

포함되는 파일 : 변수를 선언(commonVariable.jspf)

```
<%
    String memberID = (String)session.getAttribute("MEMBERID")
    File tempFolder = (File)application.getAttribute("TEMPTOLDER");
%>
```

포함하는 파일 : include 디렉티브로 변수 선언 코드 삽입

```
<%@ include file="commonVariable.jspf"%>
...
<%= memberID %>
...
<%
    Writer fw = new FileWriter(tempFolder);
    ...
%>
... 
```

# 액션 태그

---

## ❖ 액션 태그의 개요

- 서버나 클라이언트에게 어떤 행동을 하도록 명령하는 태그
  - JSP 페이지에서 페이지와 페이지 사이를 제어하거나,
  - 다른 페이지의 실행 결과 내용을 현재 페이지에 포함하거나,
  - 자바 빈즈(JavaBeans) 등의 다양한 기능을 제공
- 
- XML 형식 <jsp: ... > ~~ </jsp: ...>를 사용

# 액션 태그

## ❖ 액션 태그의 종류

액션 태그	형식	설명
forward	<jsp:forward ... />	다른 페이지로의 이동과 같은 페이지 흐름을 제어합니다.
include	<jsp:include ... />	외부 페이지의 내용을 포함하거나 페이지를 모듈화합니다.
useBean	<jsp:useBean ... />	JSP 페이지에 자바빈즈를 설정합니다.
setProperty	<jsp:setProperty ... />	자바빈즈의 프로퍼티 값을 설정합니다.
getProperty	<jsp:getProperty ... />	자바빈즈의 프로퍼티 값을 얻어옵니다.
param	<jsp:param ... />	<jsp:forward>, <jsp:include>, <jsp:plugin> 태그에 인자를 추가합니다.
plugin	<jsp:plugin ... />	웹 브라우저에 자바 애플릿을 실행합니다. 자바 플러그인에 대한 OBJECT 또는 EMBED 태그를 만드는 브라우저별 코드를 생성합니다.
element	<jsp:element ... />	동적 XML 요소를 설정합니다.
attribute	<jsp:attribute ... />	동적으로 정의된 XML 요소의 속성을 설정합니다.
body	<jsp:body ... />	동적으로 정의된 XML 요소의 몸체를 설정합니다.
text	<jsp:text ... />	JSP 페이지 및 문서에서 템플릿 텍스트를 작성합니다.

# 액션 태그

## ❖ 액션 태그의 종류

액션 태그	형식	설명
forward	<code>&lt;jsp:forward ... /&gt;</code>	다른 페이지로의 이동과 같은 페이지 흐름을 제어합니다.
include	<code>&lt;jsp:include ... /&gt;</code>	외부 페이지의 내용을 포함하거나 페이지를 모듈화합니다.
useBean	<code>&lt;jsp:useBean ... /&gt;</code>	JSP 페이지에 자바빈즈를 설정합니다.
setProperty	<code>&lt;jsp:setProperty ... /&gt;</code>	자바빈즈의 프로퍼티 값을 설정합니다.
getProperty	<code>&lt;jsp:getProperty ... /&gt;</code>	자바빈즈의 프로퍼티 값을 얻어옵니다.
param	<code>&lt;jsp:param ... /&gt;</code>	<code>&lt;jsp:forward&gt;</code> , <code>&lt;jsp:include&gt;</code> , <code>&lt;sp:plugin&gt;</code> 태그에 인자를 추가합니다.
plugin	<code>&lt;jsp:plugin ... /&gt;</code>	웹 브라우저에 자바 애플릿을 실행합니다. 자바 플러그인에 대한 OBJECT 또는 EMBED 태그를 만드는 브라우저별 코드를 생성합니다.
element	<code>&lt;jsp:element ... /&gt;</code>	동적 XML 요소를 설정합니다.
attribute	<code>&lt;jsp:attribute ... /&gt;</code>	동적으로 정의된 XML 요소의 속성을 설정합니다.
body	<code>&lt;jsp:body ... /&gt;</code>	동적으로 정의된 XML 요소의 몸체를 설정합니다.
text	<code>&lt;jsp:text ... /&gt;</code>	JSP 페이지 및 문서에서 템플릿 텍스트를 작성합니다.

## 2. <jsp:include> 액션 태그와 모듈화

### ❖ <jsp:include> 액션 태그를 이용한 공통 영역 작성

#### ■ 일반적인 웹 사이트 페이지 구성

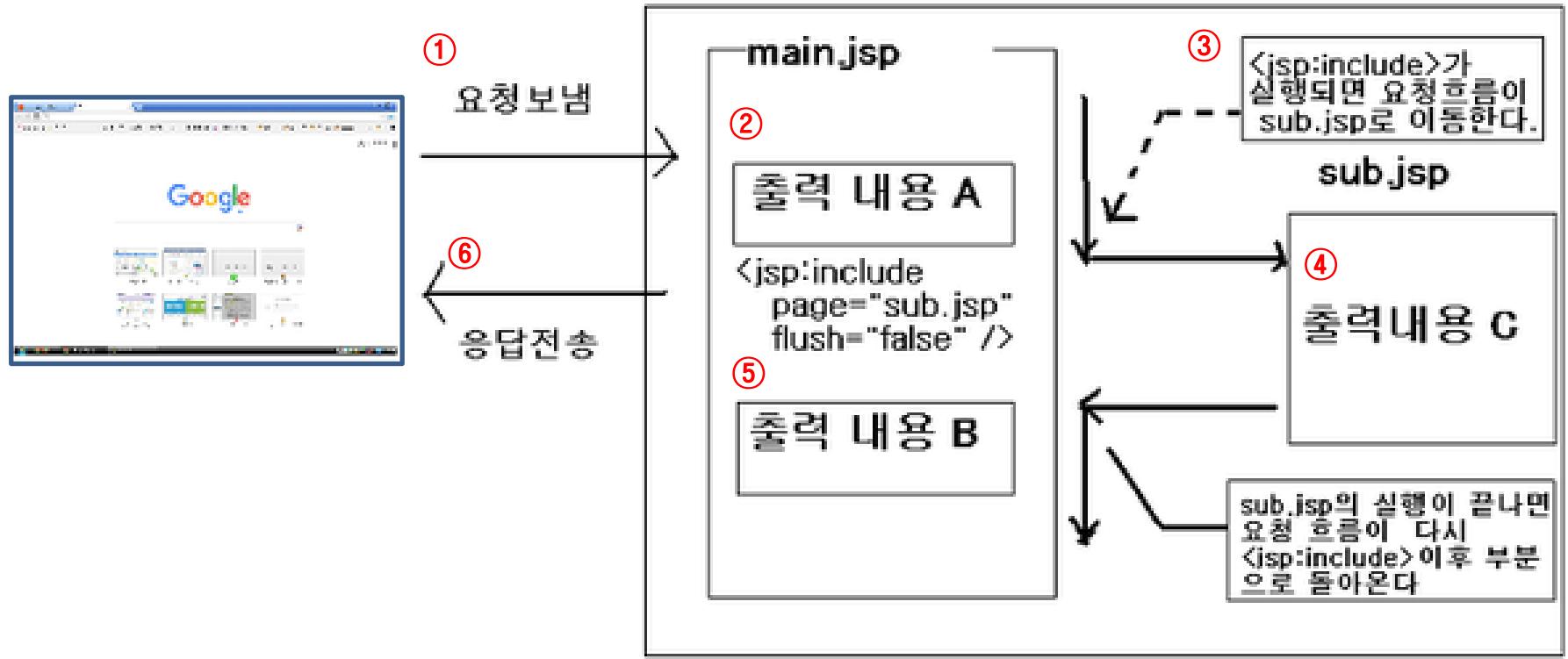
- 상단 메뉴
- 좌측 메뉴
- 하단 영역(footer)

#### ■ 화면 구성 요소의 코드 중복 문제를 없앨 때 사용

➤ <jsp:include> 액션 태그

## 2. <jsp:include> 액션 태그와 모듈화

### ❖ <jsp:include> 액션 태그 동작 방식



## 2. <jsp:include> 액션 태그 예제

### ❖ main.jsp

```
main.jsp ✘
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4<html>
5<head>
6 <meta charset="UTF-8">
7 <title>Insert title here</title>
8 </head>
9<body>
10 main.jsp에서 생성한 내용입니다
11 <jsp:include page="sub.jsp"                      ></jsp:include>
12
13 include 이후의 내용입니다.
14 </body>
15 </html>
16
```

## 2. <jsp:include> 액션 태그 예제

### ❖ main.jsp

```
main.jsp ✘
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4<html>
5<head>
6 <meta charset="UTF-8">
7 <title>Insert title here</title>
8 </head>
9<body>
10 main.jsp에서 생성한 내용입니다
11 <jsp:include page="sub.jsp"                      ></jsp:include>
12
13 include 이후의 내용입니다.
14 </body>
15 </html>
16
```

포함할 JSP 페이지의 경로를 지정한다

## 2. <jsp:include> 액션 태그 예제

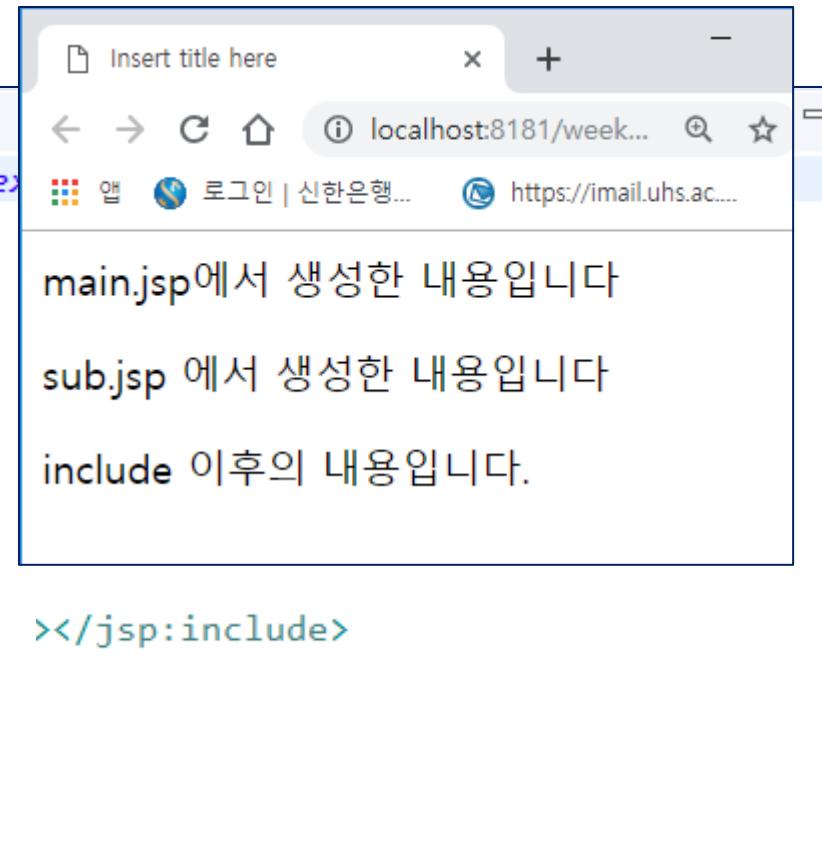
### ❖ sub.jsp

```
sub.jsp
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4<html>
5<head>
6 <meta charset="UTF-8">
7 <title>Insert title here</title>
8 </head>
9<body>
10<p>
11 sub.jsp 에서 생성한 내용입니다
12 </p>
13 </body>
14 </html>
15
```

## 2. <jsp:include> 액션 태그 예제

### ❖ main.jsp

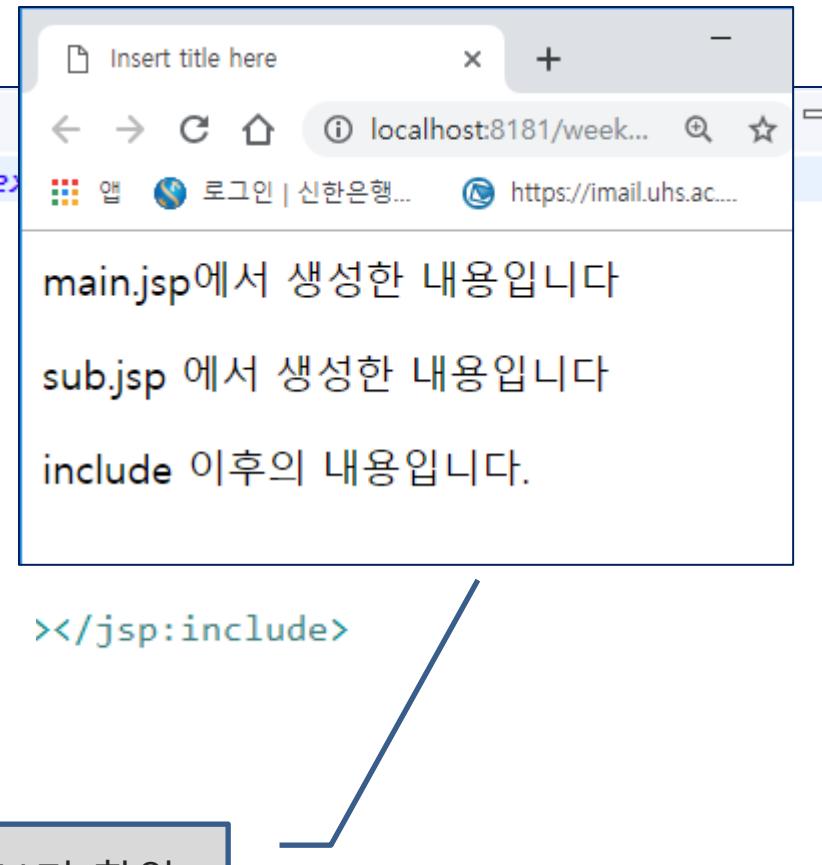
```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"%>
2 <!DOCTYPE html>
3 <html>
4 <head>
5 <meta charset="UTF-8">
6 <title>Insert title here</title>
7 </head>
8 <body>
9 main.jsp에서 생성한 내용입니다
10 <jsp:include page="sub.jsp"
11 </body>
12 </html>
13 include 이후의 내용입니다.
14 </body>
15 </html>
16
```



## 2. <jsp:include> 액션 태그 예제

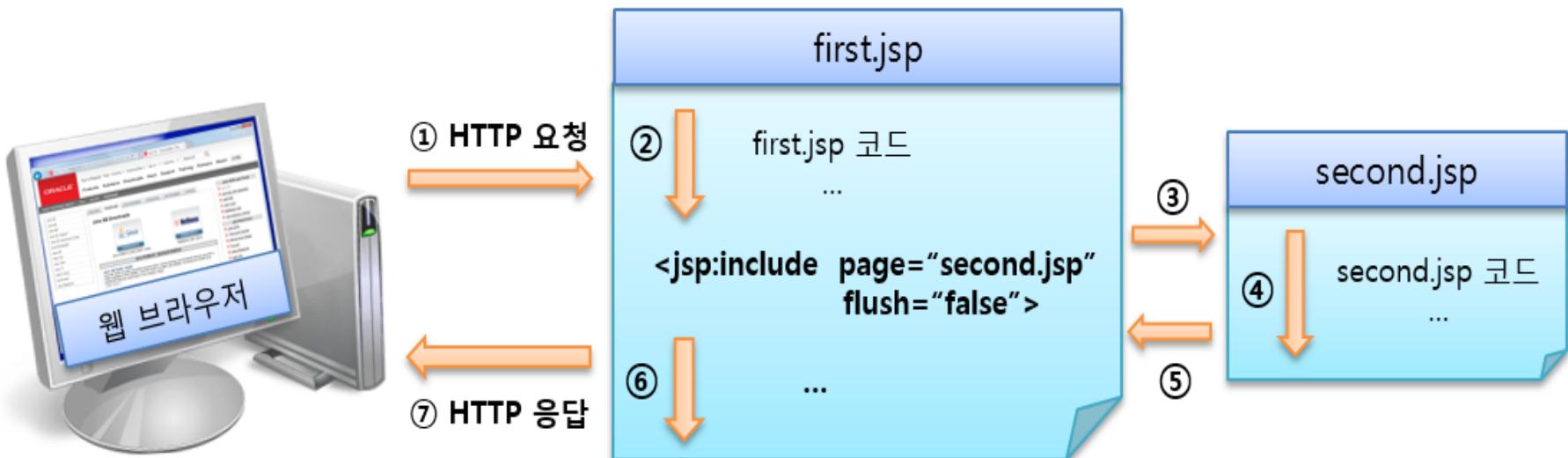
### ❖ main.jsp

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"%>
2 <!DOCTYPE html>
3 <html>
4 <head>
5 <meta charset="UTF-8">
6 <title>Insert title here</title>
7 </head>
8 <body>
9 main.jsp에서 생성한 내용입니다
10 <jsp:include page="sub.jsp" />
11 include 이후의 내용입니다.
12 </body>
13 </html>
14
15
16
```



## 2. <jsp:include> 액션 태그

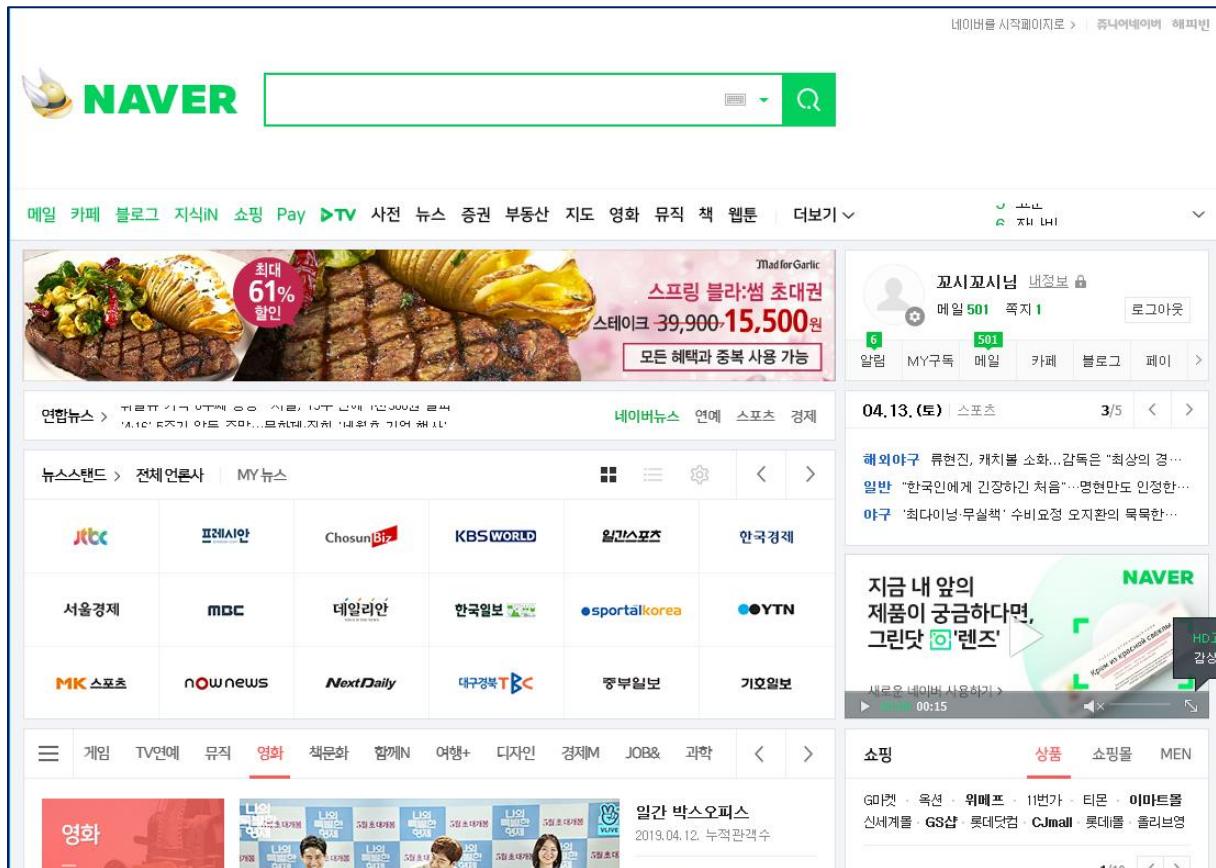
### ❖ include 액션 태그의 처리 과정



## 2. <jsp:include> 액션 태그와 모듈화

### ❖ <jsp:include> 액션 태그를 이용한 중복 영역 처리

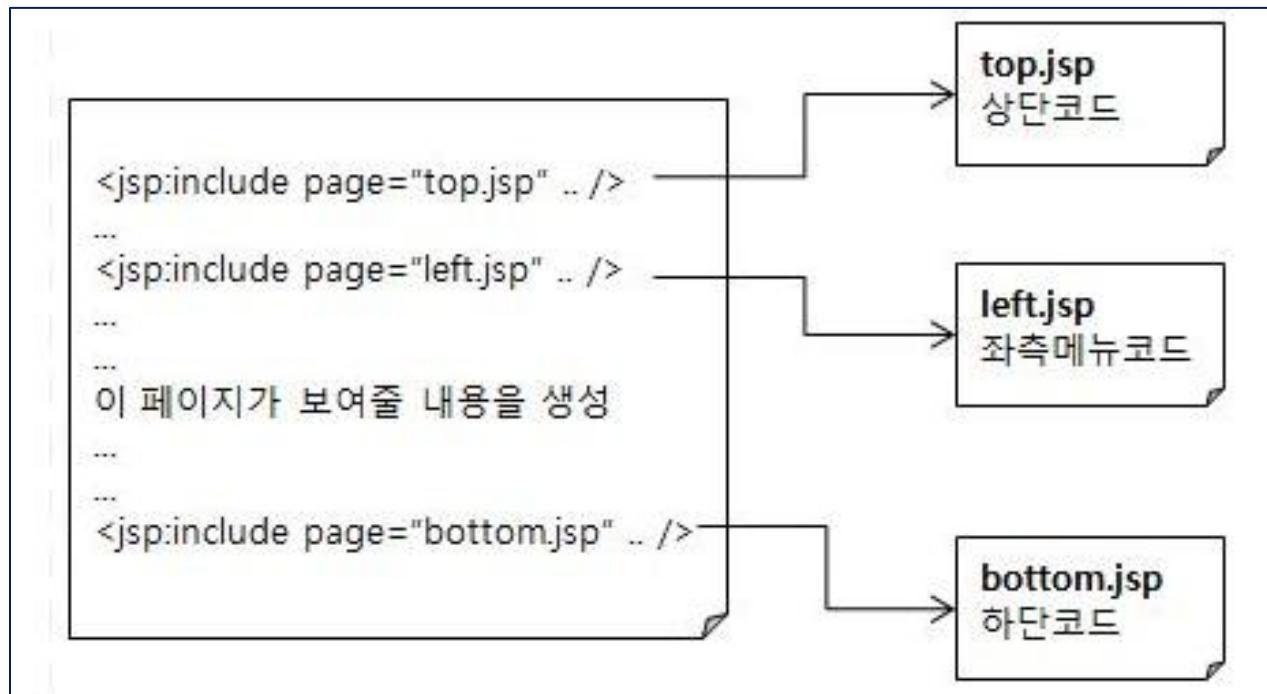
- 일반적인 웹 사이트는 상단메뉴, 좌측메뉴, 중앙내용, 하단 메뉴 등의 요소로 구성
- 상단, 하단 메뉴 – 모든 페이지에서 동일
- 좌측 메뉴, 중앙 내용 – 페이지에 따라 달라진다



## 2. <jsp:include> 액션 태그와 모듈화

### ❖ <jsp:include> 액션 태그를 이용한 중복 영역 처리

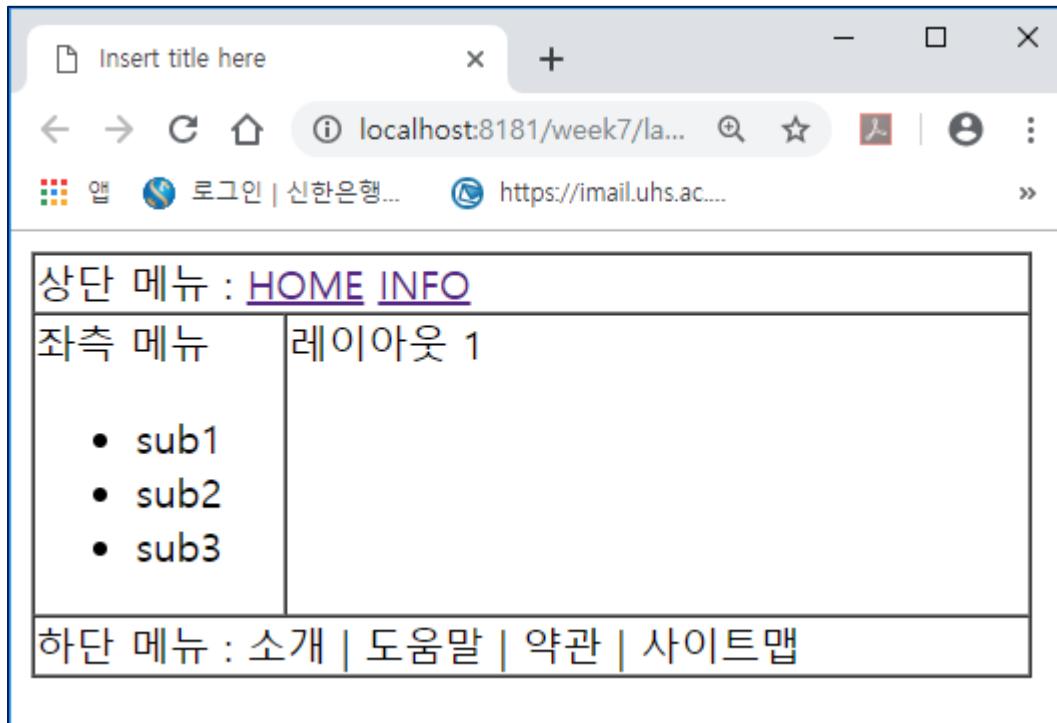
- 일반적인 웹 사이트는 상단메뉴, 좌측메뉴, 중앙내용, 하단 메뉴 등의 요소로 구성
- 상단, 하단 메뉴 – 모든 페이지에서 동일
- 좌측 메뉴, 중앙 내용 – 페이지에 따라 달라진다



## 2. <jsp:include> 액션 태그와 모듈화

### ❖ <jsp:include> 액션 태그를 이용한 중복 영역 처리

- 일반적인 웹 사이트는 상단메뉴, 좌측메뉴, 중앙내용, 하단 메뉴 등의 요소로 구성
- 상단, 하단 메뉴 – 모든 페이지에서 동일
- 좌측 메뉴, 중앙 내용 – 페이지에 따라 달라진다



## 2. <jsp:include> 액션 태그와 모듈화

### ❖ <jsp:include> 액션 태그를 이용한 중복 영역 처리

- 일반적인 웹 사이트는 상단메뉴, 좌측메뉴, 중앙내용, 하단 메뉴 등의 요소로 구성
- 상단, 하단 메뉴 – 모든 페이지에서 동일
- 좌측 메뉴, 중앙 내용 – 페이지에 따라 달라진다

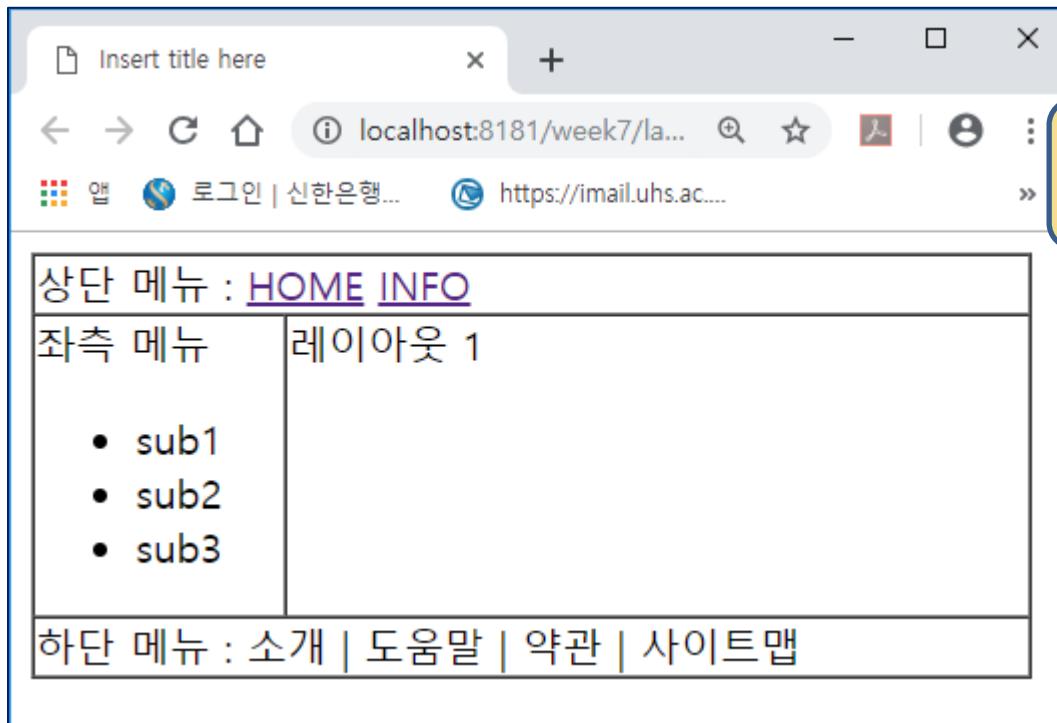


table 구성 : 3행 2열 기본

## 2. <jsp:include> 액션 태그와 모듈화

### ❖ <jsp:include> 액션 태그를 이용한 중복 영역 처리(layout.jsp)

```
layout.jsp X top.jsp left.jsp bottom.jsp
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3 <!DOCTYPE html><html><head><meta charset="UTF-8">
4 <title>Insert title here</title></head>
5 <body>
6     <table width="400" border="1" cellpadding="0" cellspacing="0">
7         <tr><td colspan="2">
8             <jsp:include page=".top.jsp" flush="false"/>
9             </td>
10            </tr>
11            <tr><td width="100" valign="top">
12                <jsp:include page=".left.jsp" flush="false"/>
13                </td>
14                <td width="300" valign="top">
15                    <!-- 내용 부분 : 시작 -->
16                    레이아웃 1<br><br><br>
17                    <!-- 내용 부분 : 끝 -->
18                </td>
19            </tr>
20            <tr><td colspan="2">
21                <jsp:include page=".bottom.jsp" flush="false"/>
22                </td>
23            </tr>
24        </table>
25    </body></html>
```

<jsp:include page="info.jsp" /> 추가

## 2. <jsp:include> 액션 태그와 모듈화

### ❖ top.jsp / left.jsp / bottom.jsp

```
layout.jsp      top.jsp ✘  left.jsp ✘  bottom.jsp
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3<!DOCTYPE html><html><head><meta charset="UTF-8">
4 <title>Insert title here</title></head>
5<body>
6 상단 메뉴 : <a href="">HOME</a>&ampnbsp<a href="">INFO</a>
```

```
layout.jsp      top.jsp ✘  left.jsp ✘  bottom.jsp
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3<!DOCTYPE html><html><head><meta charset="UTF-8">
4 <title>Insert title here</title></head>
5<body>
6 좌측 메뉴<br>
7<ul>
8     <li>sub1</li>
9     <li>sub2</li>
10    <li>sub3</li>
11</ul>
```

```
layout.jsp      top.jsp ✘  left.jsp ✘  bottom.jsp
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3<!DOCTYPE html><html><head><meta charset="UTF-8">
4 <title>Insert title here</title></head>
5<body>
6 하단 메뉴 : 소개 | 도움말 | 약관 | 사이트맵
7 </body></html>
```

## 2. <jsp:include> 액션 태그와 모듈화

### ❖ <jsp:include> 액션 태그를 이용한 중복 영역 처리(layout.jsp)

The screenshot shows a Java IDE interface with several tabs: layout.jsp, top.jsp, left.jsp, and bottom.jsp. The layout.jsp tab is active, displaying the following JSP code:

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"%>
2 <!DOCTYPE html><html><head><meta ...
3 <title>Insert title here</title>
4 <body>
5     <table width="400" border="1">
6         <tr><td colspan="2">
7             <jsp:include page="top.jsp" />
8         </td>
9     </tr>
10    <tr><td width="100" valign="top">
11        <jsp:include page="left.jsp" />
12    </td>
13    <td width="300" valign="top">
14        <!-- 내용 부분 : 시작 -->
15        레이아웃 1<br><br>
16        <!-- 내용 부분 : 끝 -->
17    </td>
18 </tr>
19 <tr><td colspan="2">
20     <jsp:include page="/bottom.jsp" flush="false"/>
21 </td>
22 </tr>
23 </table>
24 </body></html>
```

To the right of the code editor, a browser window displays the rendered output of the JSP. The browser's address bar shows "localhost:8181/week7/layout.jsp". The rendered content is:

상단 메뉴 : [HOME](#) [INFO](#)

좌측 메뉴	레이아웃 1
<ul style="list-style-type: none"><li>sub1</li><li>sub2</li><li>sub3</li></ul>	

하단 메뉴 : [소개](#) | [도움말](#) | [약관](#) | [사이트맵](#)

The browser window also includes standard navigation buttons (back, forward, search, etc.) and a status bar with the URL "https://imail.uhs.ac....".

### 3. Param 액션 태그의 기능과 사용법

#### ❖ Param 액션 태그

- 현재 JSP 페이지에서 다른 페이지에 정보를 전달하는 태그
- 단독으로 사용할 수 없으며 <jsp:include>나 <jsp:forward> 태그 내부에 사용
- 다른 페이지에 여러 개의 정보를 전송해야 할 때는 다중의 param 액션 태그를 사용

#### ❖ <jsp:param>으로 파라미터 추가하기

- <jsp:param> 액션 태그는 <jsp:include> 태그의 자식 태그로 추가된다

```
1  <% String type="typeA";%>
2  <jsp:include page="...">
3    <jsp:param name="name" value="홍길동"/> <%--value에 값을 직접 입력 --%>
4    <jsp:param name="type" value=<%=type %>/> <%--표현식으로 값 입력 --%>
5  </jsp:include>
```

### 3. Param 액션 태그의 기능과 사용법

#### ❖ Param 액션 태그 사용 예(info.jsp)

```
<body>
    <table border="1" width="100%" cellpadding="1" cellspacing="1">
        <tr>
            <td>모델번호 </td>
            <td>MQ7E2KH/A </td>
        </tr>
        <tr>
            <td>가격 </td>
            <td>1,500,000원 </td>
        </tr>
    </table>
    <br>
    <jsp:include page="infoSub.jsp">
        <jsp:param value="iPhone" name="type"/>
    </jsp:include>
</body>
```

모델번호	MQ7E2KH/A
가격	1,500,000원

### 3. Param 액션 태그의 기능과 사용법

#### ❖ Param 액션 태그 사용 예(infoSub.jsp)

```
<body>
<%
    String type = request.getParameter("type");
%>


|    |                                                                                                                               |
|----|-------------------------------------------------------------------------------------------------------------------------------|
| 타입 | <b> <%=type %> </b>                                                                                                           |
| 특징 | <%         if (type.equals("iPhone"))             out.println("수려한 디자인");         else             out.println("뛰어난 활용성"); %> |


</body>
```

타입	iPhone
특징	수려한 디자인

## 2. <jsp:include> 액션 태그와 모듈화

### ❖ <jsp:include> 액션 태그를 이용한 중복

```
layout.jsp X top.jsp left.jsp bottom.jsp
1 <%@ page language="java" contentType="text/html"
2      pageEncoding="UTF-8"%>
3<!DOCTYPE html><html><head><meta charset="UTF-8"
4 <title>Insert title here</title></head>
5<body>
6    <table width="400" border="1" cellpadding=
7        <tr><td colspan="2">
8            <jsp:include page="/top.jsp" flush=
9            </td>
10       </tr>
11       <tr><td width="100" valign="top">
12           <jsp:include page="/left.jsp" flush=
13           </td>
14           <td width="300" valign="top">
15               <!-- 내용 부분 : 시작 -->
16               레이아웃 1<br><br><br>
17               <!-- 내용 부분 : 끝 -->
18           </td>
19       </tr>
20       <tr><td colspan="2">
21           <jsp:include page="/bottom.jsp" f
22           </td>
23       </tr>
24   </table>
25 </body></html>
```

상단 메뉴 : HOME INFO

좌측 메뉴	레이아웃 1
• sub1	
• sub2	
• sub3	
모델번호	MQ7E2KH/A
가격	1,000,000원
타입	iPhone
특징	수려한 디자인

하단 메뉴 : 소개 | 도움말 | 약관 | 사이트맵

<jsp:include page="info.jsp" /> 추가

# <jsp:include> 액션 태그 & include 디렉티브

## ❖ <jsp:include> 액션 태그와 include 디렉티브의 차이

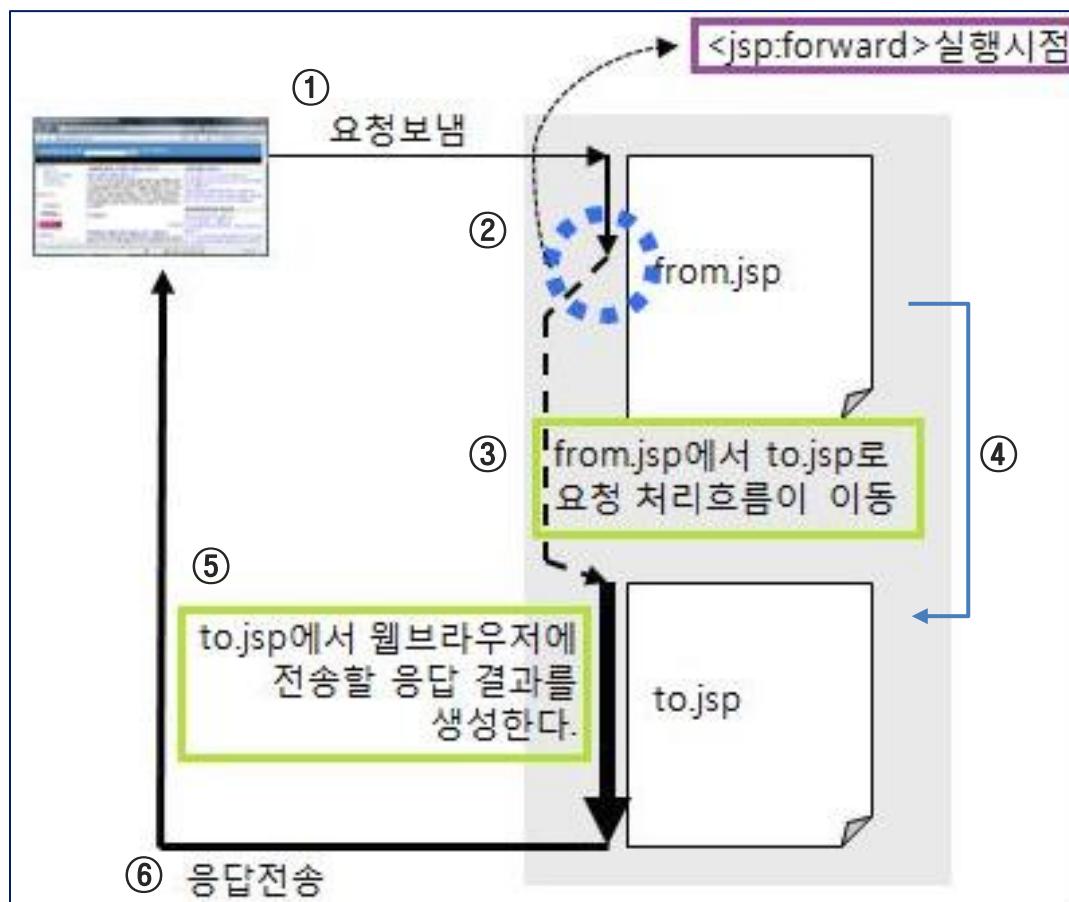
비교 항목	<jsp:include>	include 디렉티브
처리 시간	요청 시간에 처리	JSP 파일을 자바 소스로 변환할 때 처리
기능	별도의 파일로 요청 처리 흐름을 이동	현재 파일에 삽입시킴
데이터 전달 방법	request 기본 객체나 <jsp:param>을 이용한 파라미터 전달	페이지 내의 변수를 선언한 후, 변수에 값 저장
용도	화면의 레이아웃의 일부분을 모듈화할 때 주로 사용한다	다수의 JSP 페이지에서 공통으로 사용되는 변수를 지정하는 코드나 저작권과 같은 문장을 포함한다
사용법	<jsp:include page="top.jsp" />	<%@ include file="dateTime.jsp">

## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ <jsp:forward> 액션 태그를 이용한 JSP 페이지 이동

- 하나의 JSP 페이지에서 다른 JSP 페이지로 요청 처리를 전달할 때 사용

### ❖ <jsp:forward> 액션 태그의 요청 흐름

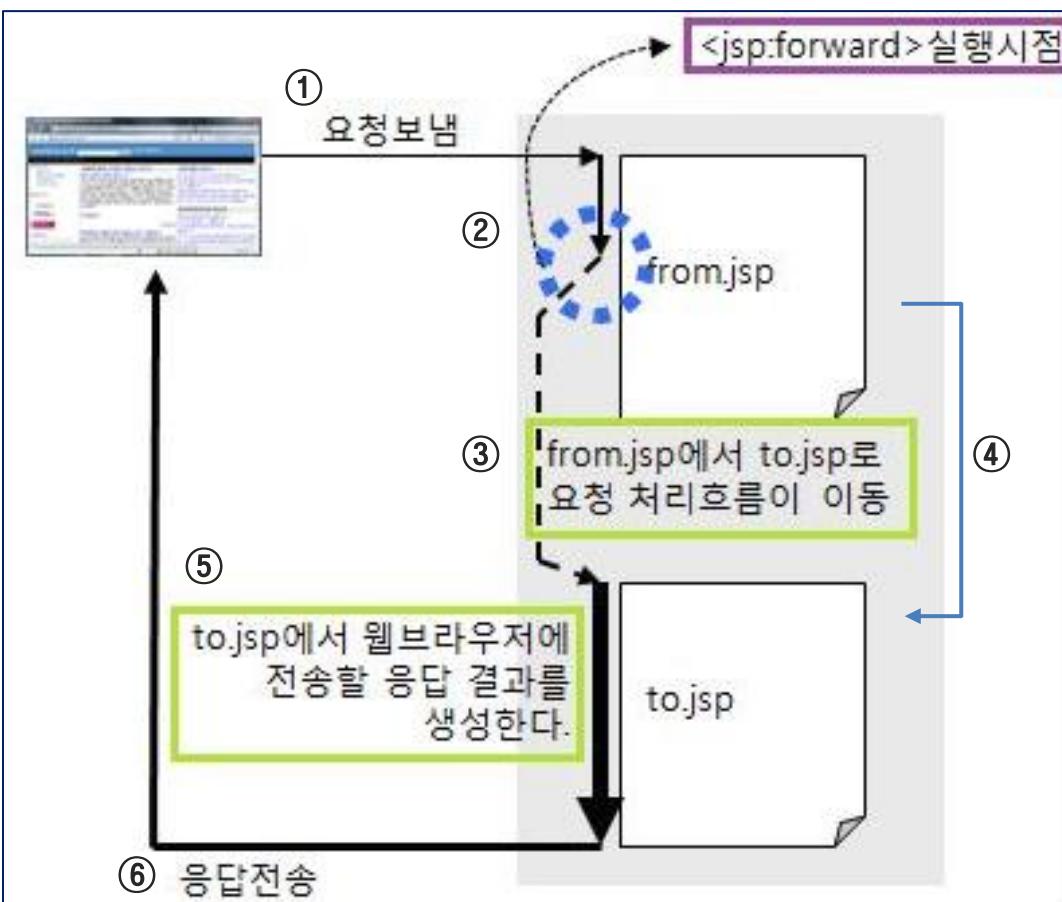


## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ <jsp:forward> 액션 태그를 이용한 JSP 페이지 이동

- 하나의 JSP 페이지에서 다른 JSP 페이지로 요청 처리를 전달할 때 사용

### ❖ <jsp:forward> 액션 태그의 요청 흐름



- ① 웹 브라우저 요청을 from.jsp에 전달한다.
- ② from.jsp는 <jsp:forward> 액션 태그를 실행한다.
- ③ <jsp:forward> 액션 태그를 실행하면 요청 흐름이 to.jsp로 이동한다.
- ④ 요청 흐름이 이동할 때 from.jsp에서 사용한 request 기본 객체와 response 기본 객체가 to.jsp로 전달된다.
- ⑤ to.jsp가 응답 결과를 생성한다.
- ⑥ to.jsp가 생성한 결과가 웹 브라우저에 전달된다.

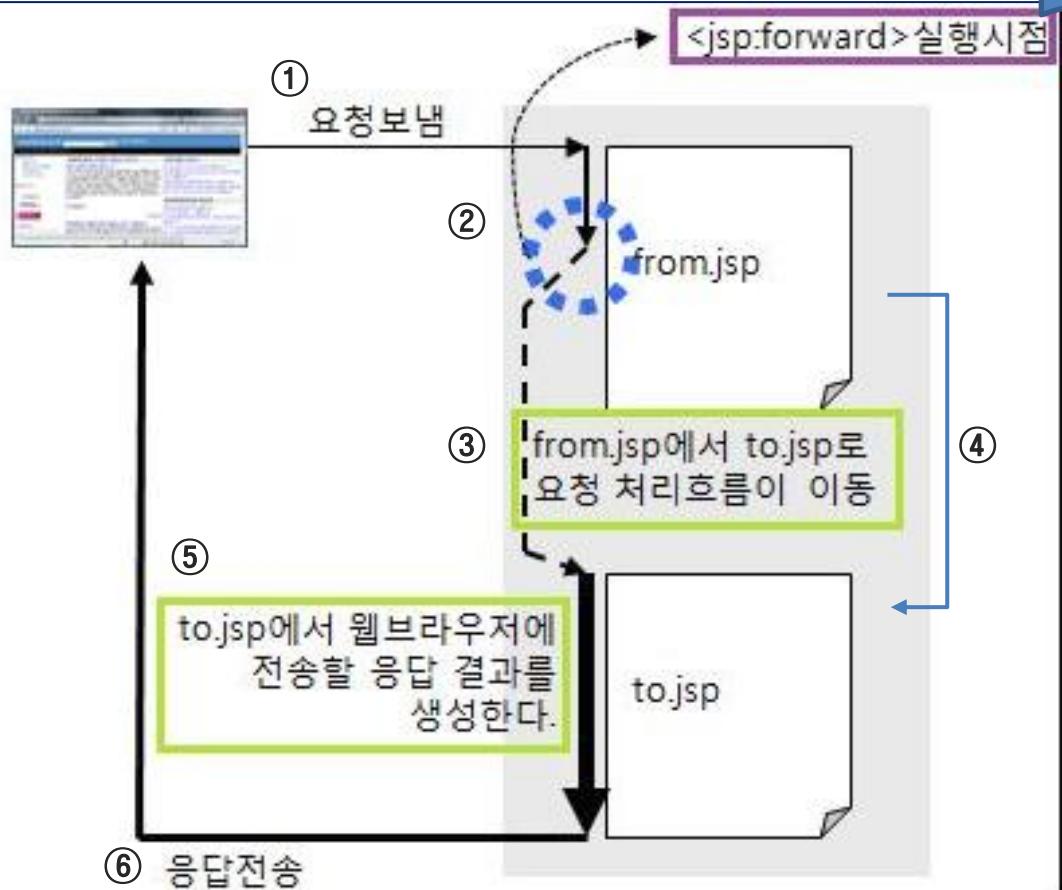
## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ <jsp:forward> 액션 태그를 이용한 JSP 페이지 이동

- 하나의 JSP 페이지에서 다른 JSP 페이지로 요청 처리를 전달할 때 사용

### ❖ <jsp:forward> 액션 태그의 요청

MVC (Model-View-Controller)  
기반 웹어플리케이션에서 사용



① 웹 브라우저 요청을 from.jsp에 전달한다.

② from.jsp는 <jsp:forward> 액션 태그를 실행한다.

③ <jsp:forward> 액션 태그를 실행하면 요청 흐름이 to.jsp로 이동한다.

④ 요청 흐름이 이동할 때 from.jsp에서 사용한 request 기본 객체와 response 기본 객체가 to.jsp로 전달된다.

⑤ to.jsp가 응답 결과를 생성한다.

⑥ to.jsp가 생성한 결과가 웹 브라우저에 전달된다.

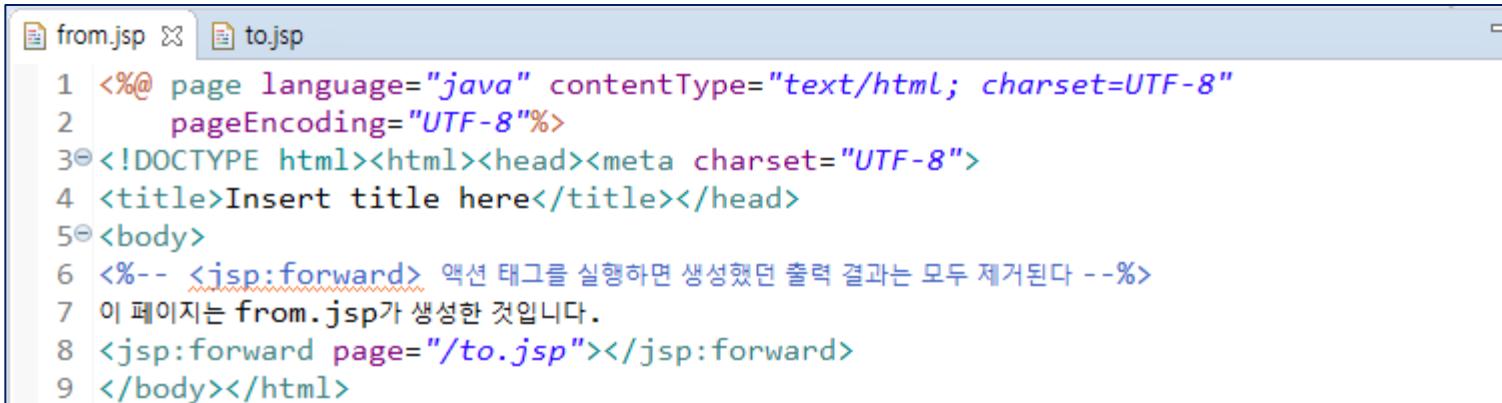
## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ <jsp:forward> 액션 태그의 사용법

```
1 <jsp:forward page="이동할페이지"/>
```

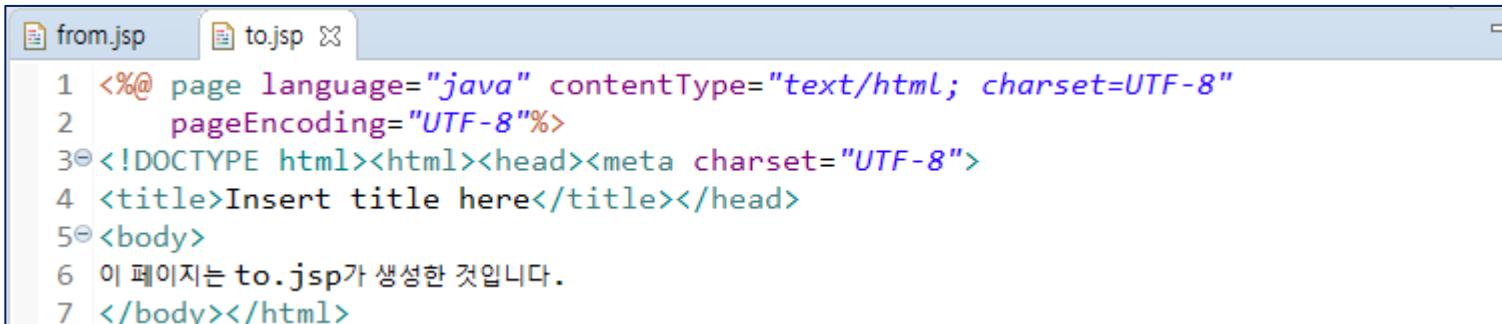
### ❖ <jsp:forward> 액션 태그 예제

#### ▪ from.jsp



```
from.jsp ✘ to.jsp
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3 <!DOCTYPE html><html><head><meta charset="UTF-8">
4 <title>Insert title here</title></head>
5 <body>
6 <%-- <jsp:forward> 액션 태그를 실행하면 생성했던 출력 결과는 모두 제거된다 --%>
7 이 페이지는 from.jsp가 생성한 것입니다.
8 <jsp:forward page="/to.jsp"></jsp:forward>
9 </body></html>
```

#### ▪ to.jsp



```
from.jsp ✘ to.jsp
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3 <!DOCTYPE html><html><head><meta charset="UTF-8">
4 <title>Insert title here</title></head>
5 <body>
6 이 페이지는 to.jsp가 생성한 것입니다.
7 </body></html>
```

# 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

## ❖ <jsp:forward> 액션

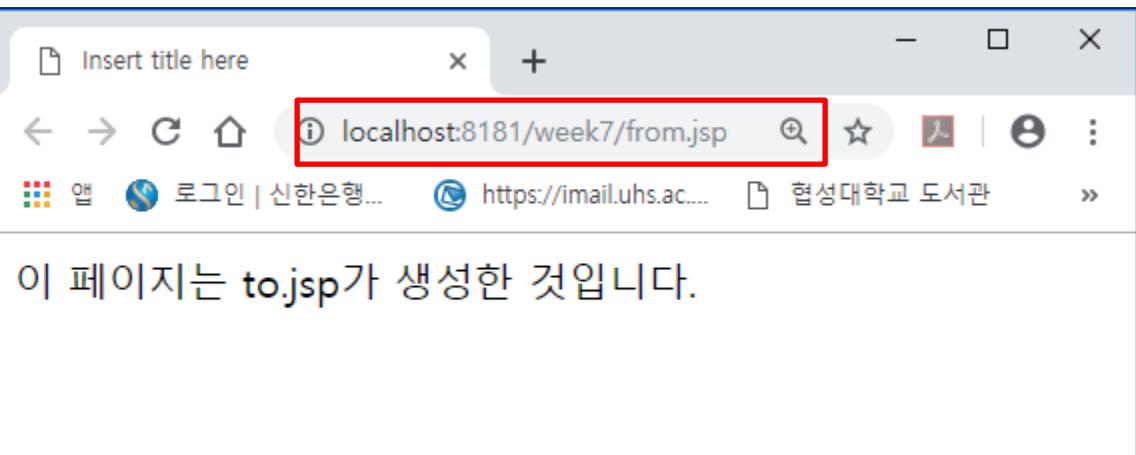
```
1 <jsp:forward page="이동할파
```

## ❖ <jsp:forward> 액션

### ▪ from.jsp

```
from.jsp ✘ to.jsp
```

```
1 <%@ page language="java"
2     pageEncoding="UTF-8"%>
3<!DOCTYPE html><html><head><meta charset="UTF-8">
4 <title>Insert title here</title></head>
5<body>
6 <%-- <jsp:forward> 액션 태그를 실행하면 생성했던 출력 결과는 모두 제거된다 --%>
7 이 페이지는 from.jsp가 생성한 것입니다.
8 <jsp:forward page="/to.jsp"></jsp:forward>
9 </body></html>
```



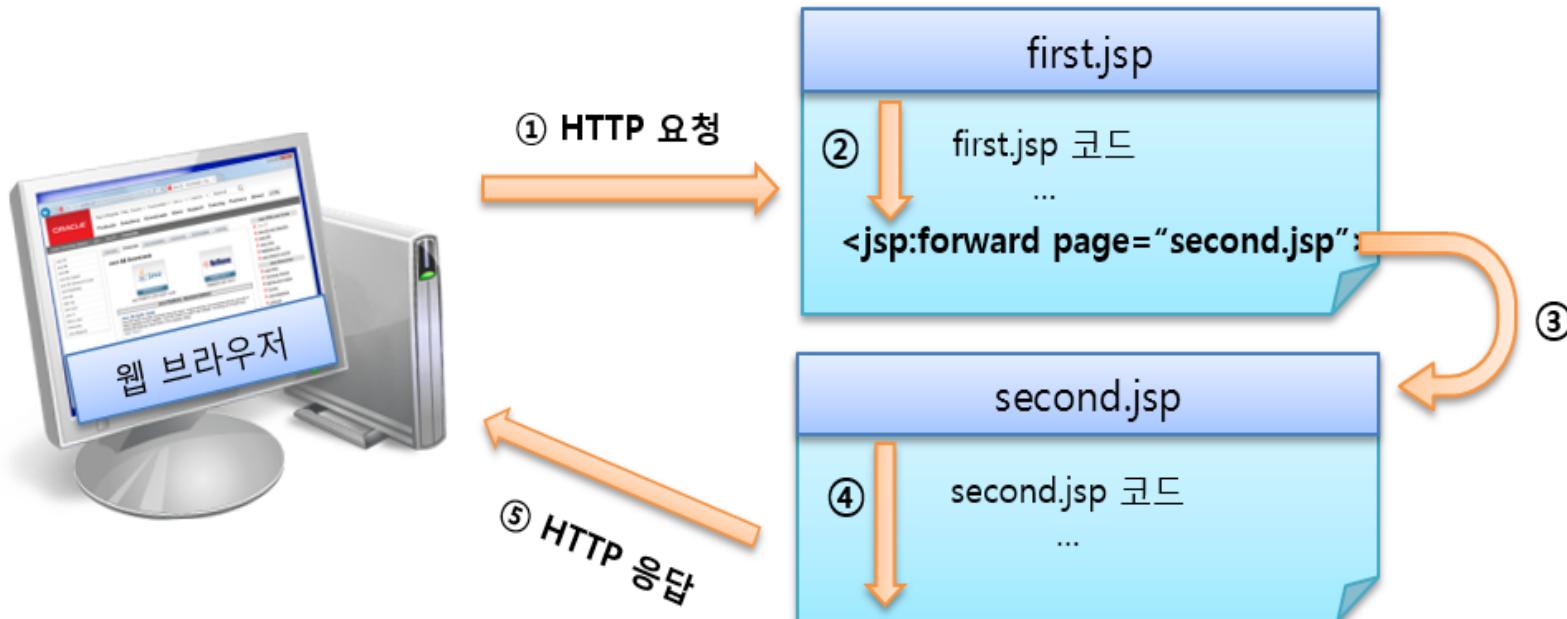
### ▪ to.jsp

```
from.jsp ✘ to.jsp ✘
```

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3<!DOCTYPE html><html><head><meta charset="UTF-8">
4 <title>Insert title here</title></head>
5<body>
6 이 페이지는 to.jsp가 생성한 것입니다.
7 </body></html>
```

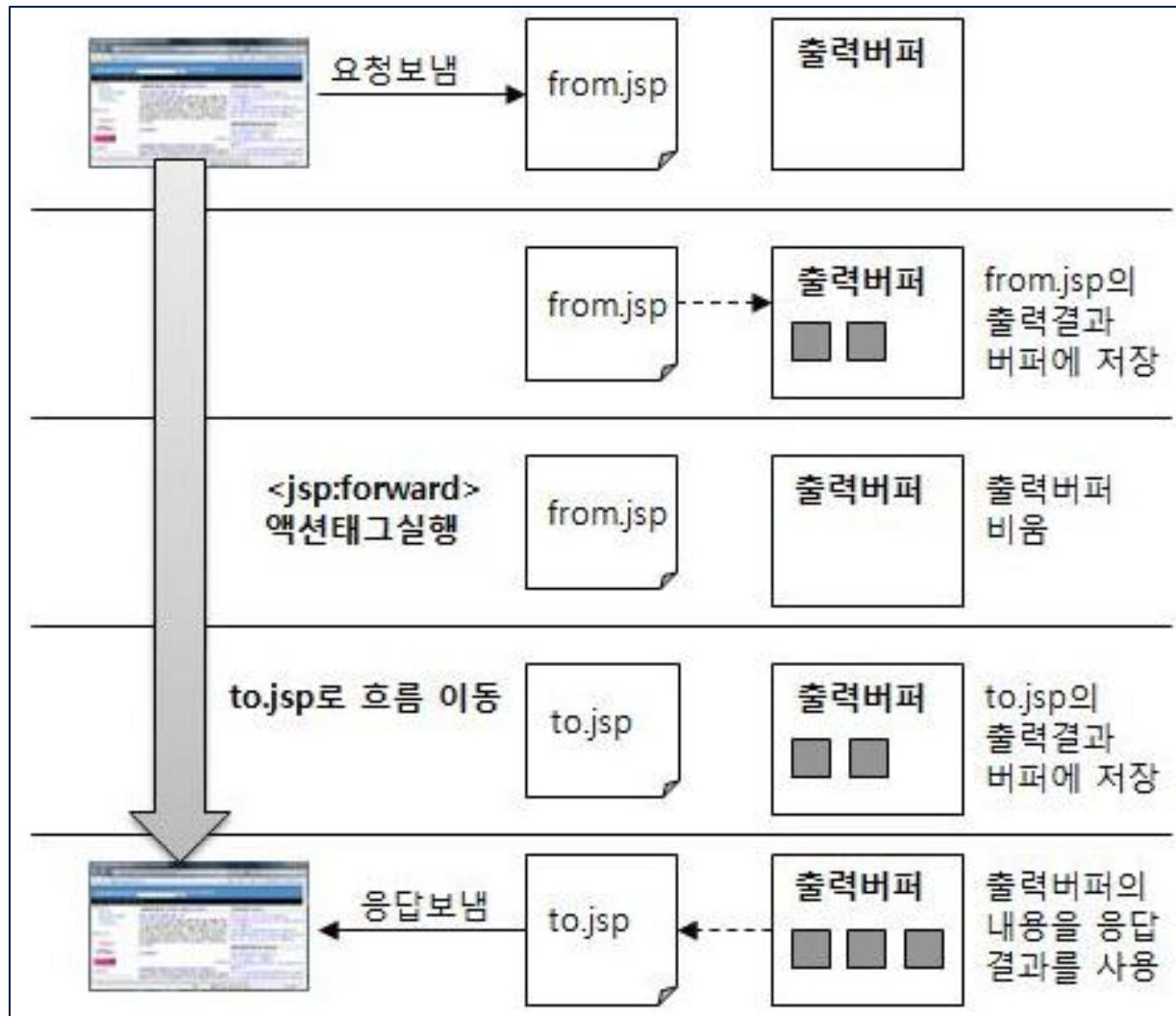
## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

- ❖ <jsp:forward> 액션 태그의 페이지 흐름 처리 과정



## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ <jsp:forward> 액션 태그와 출력 버퍼와의 관계



## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ forward 액션태그에서 페이지 사이에 값 전달하는 방법

- <jsp:param> 액션 태그 이용
- 기본 객체(request)의 속성 이용

## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ forward 액션태그에서 페이지 사이에 값 전달하는 방법

#### ▪ <jsp:param> 액션 태그 이용

```
1 <jsp:forward page="moveTo.jsp">
2   <jsp:param name="first" value="BK"/>
3   <jsp:param name="last" value="Choi"/>
4 </jsp:forward>
```

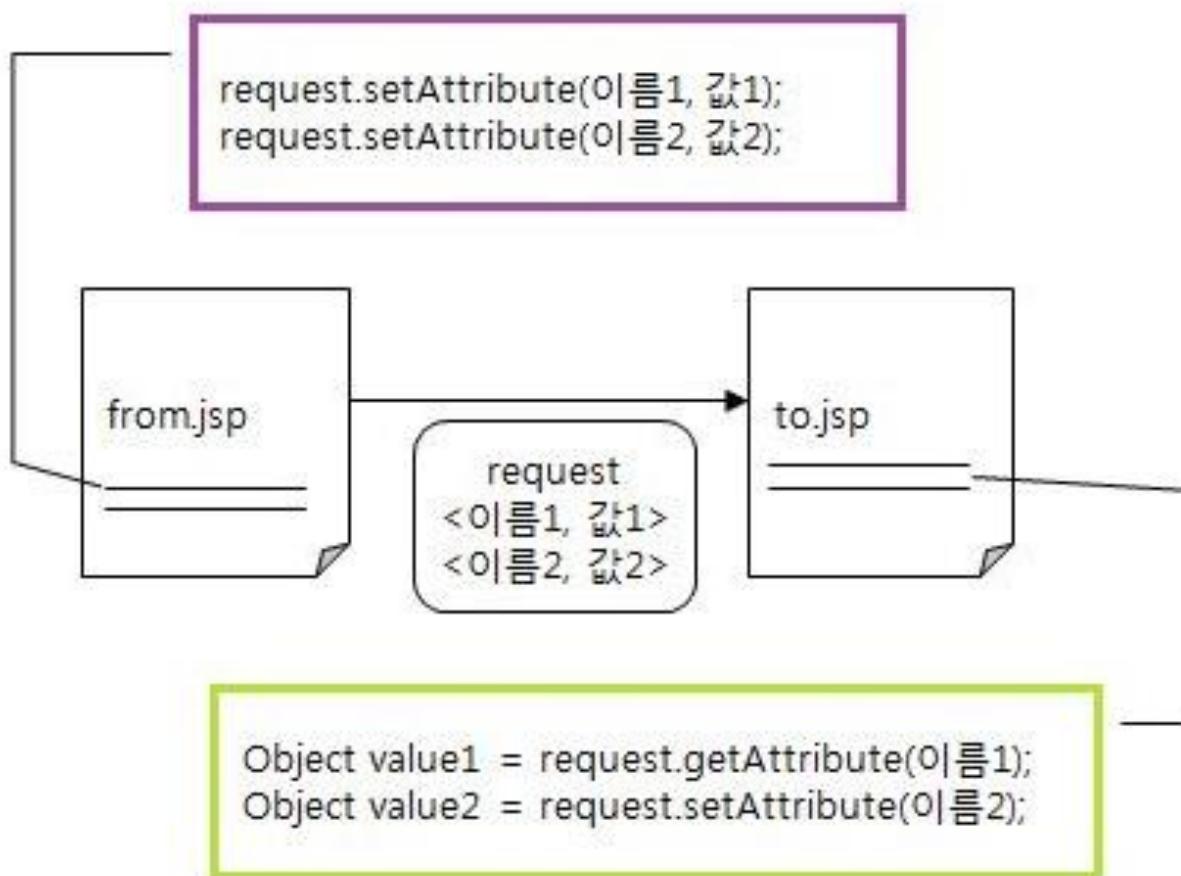


String 타입의 값만 전달

## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ forward 액션태그에서 페이지 사이에 값 전달하는 방법

- 기본 객체(request)의 속성 이용 : `setAttribute()`, `getAttribute()`



## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ forward 액션태그에서 페이지 사이에 값 전달하는 방법

#### ■ <jsp:param> 액션 태그 이용

```
1 <jsp:forward page="moveTo.jsp">
2   <jsp:param name="first" value="BK"/>
3   <jsp:param name="last" value="Choi"/>
4 </jsp:forward>
```



String 타입의 값만 전달

#### ■ 실습

- from.jsp 에서 to.jsp로 이동하기
- 액션태그를 이용하여 값 전달하기
  - 아이디
  - 패스워드
  - 이름
  - 주소
  - 이메일

## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ <jsp:param> 액션 태그 이용(from.jsp)

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html><html><head><meta charset="UTF-8">
<title>Insert title here</title></head>
<body>
<%-- <jsp:forward> 액션 태그를 실행하면 생성했던 출력 결과는 모두 제거된다 --%>
이 페이지는 from.jsp가 생성한 것입니다.

<jsp:forward page="/to.jsp">
    <jsp:param value="admin" name="id"/>
    <jsp:param value="adminpw" name="password"/>
    <jsp:param value="홍길동" name="name"/>
    <jsp:param value="서울" name="addr"/>
    <jsp:param value="hgd@naver.com" name="email"/>
</jsp:forward>
</body></html>
```

## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ <jsp:param> 액션 태그 이용(to.jsp)

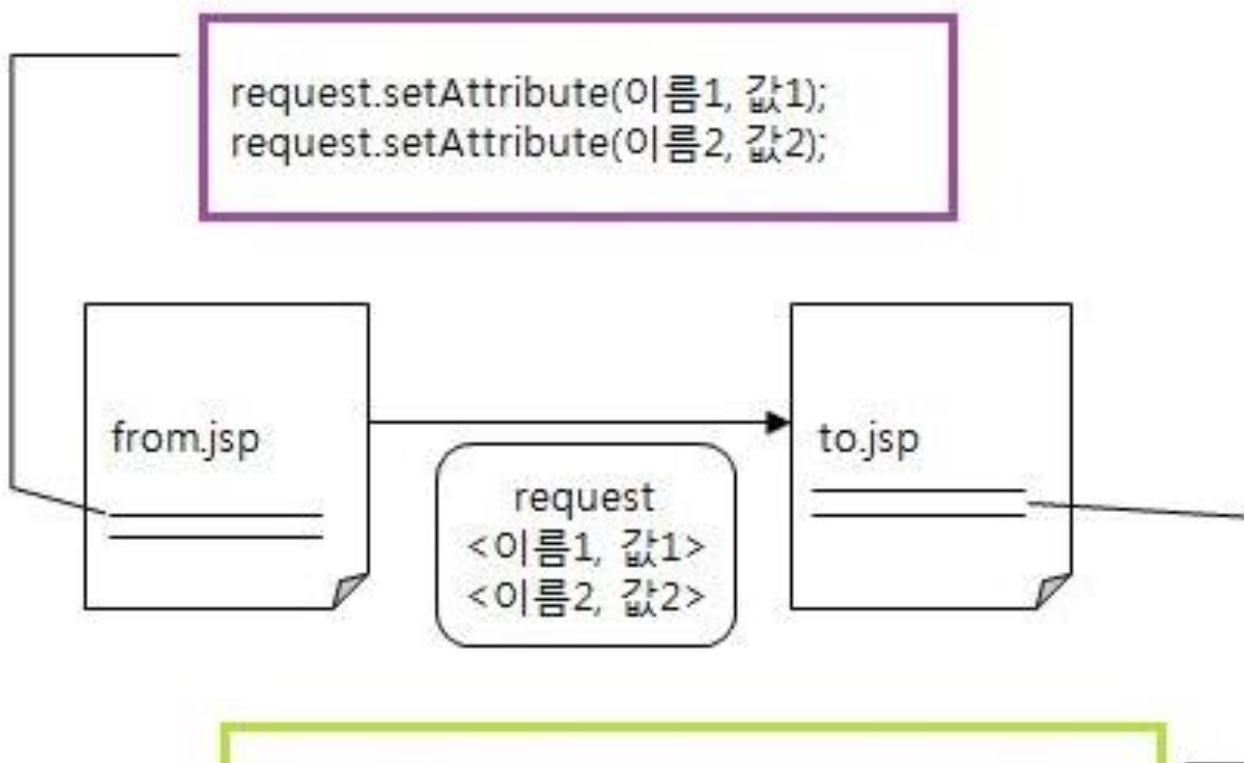
```
<body>
    <h3>이 페이지는 to.jsp가 생성한 것입니다.</h3>

    <h3>jsp:forward의 param 액션 태그 결과</h3>
    <ul>
        <li>아이디 : <%=request.getParameter("id") %> </li>
        <li>패스워드 : <%=request.getParameter("password") %> </li>
        <li>이름 : <%=request.getParameter("name") %> </li>
        <li>주소 : <%=request.getParameter("addr") %> </li>
        <li>이메일 : <%=request.getParameter("email") %> </li>
    </ul>
</body>
```

## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

- ❖ 기본 객체(request)의 속성을 이용해서 값 전달하기

- 기본 객체의 속성 이용



## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ 기본 객체의 속성을 이용한 값 전달(setAttribute.jsp)

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html><html><head><meta charset="UTF-8">
<title>Insert title here</title></head>
<body>
이 페이지는 setAttribute.jsp가 생성한 것입니다.

<%
    request.setAttribute("id", "admin");
    request.setAttribute("password", "adminpw");
    request.setAttribute("name", "홍길동");
    request.setAttribute("addr", "서울");
    request.setAttribute("email", "hgd@naver.com");
%
<jsp:forward page="/getAttribute.jsp"/>
</body></html>
```

## 4. <jsp:forward> 액션 태그를 이용한 흐름 이동

### ❖ 기본 객체의 속성을 이용한 값 전달(getAttribute.jsp)

```
<body>
    <h3>이 페이지는 getAttribute.jsp가 생성한 것입니다.</h3>

    <h3>request 기본객체의 속성을 이용한 값 전달 결과</h3>
    <ul>
        <li>아이디 - <%=request.getAttribute("id") %> </li>
        <li>패스워드 - <%=request.getAttribute("pw") %> </li>
        <li>이름 - <%=request.getAttribute("name") %> </li>
        <li>주소 - <%=request.getAttribute("addr") %> </li>
        <li>이메일 - <%=request.getAttribute("email") %> </li>
    </ul>
</body>
```

# request 객체를 이용한 데이터 전달 실습

- ❖ 로그인 화면 만들기(로그인 – “admin” , “1234” 인 경우)

<loginForm.jsp>

A screenshot of a web browser window. The address bar shows "localhost:8181/week5/loginForm.jsp". The page title is "★ 로그인 화면 ★". It contains a form with two input fields: "아이디" (id) with value "sunyae" and "비밀번호" (password) with value "....". A blue arrow points from this screen to the next one.

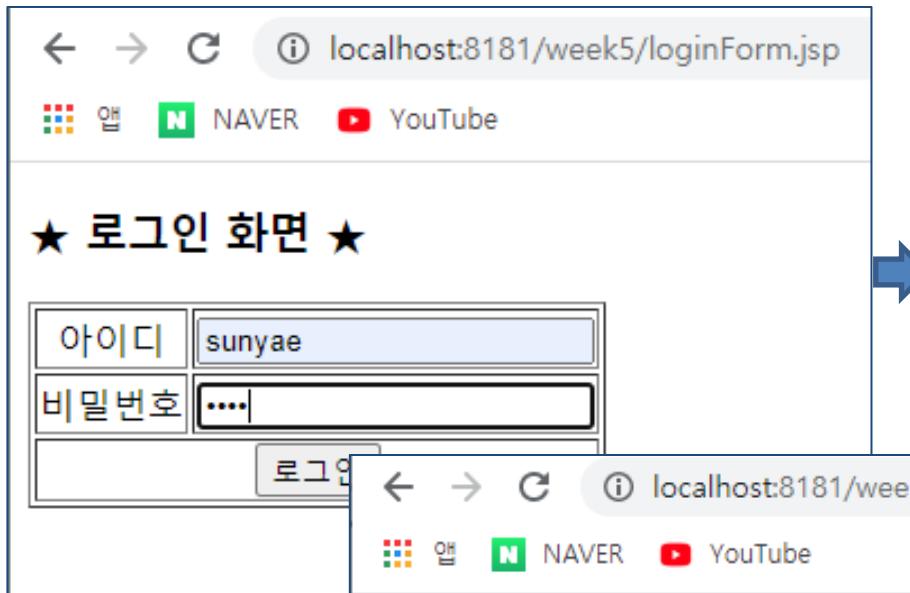
<loginResult.jsp>

A screenshot of a web browser window. The address bar shows "localhost:8181/week5/loginResult.jsp". The page displays the message "admin님, 안녕하세요!" (Hello admin!) and a button labeled "초기화면" (Initial Screen).

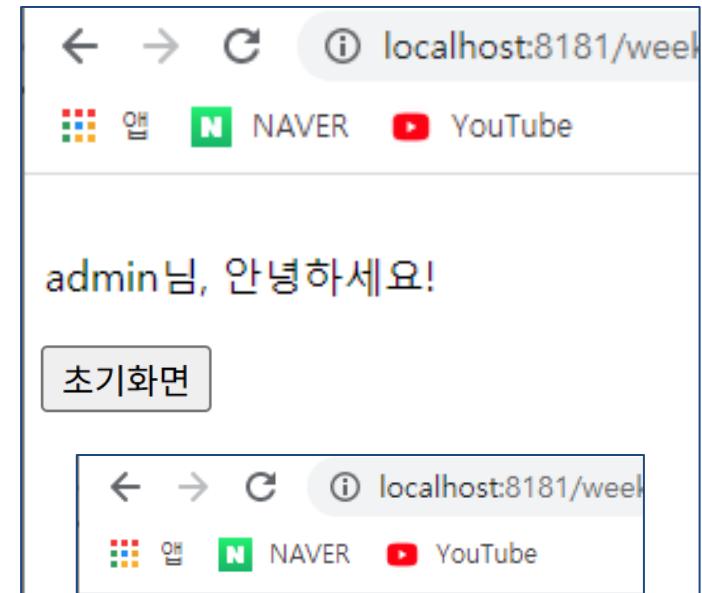
# request 객체를 이용한 데이터 전달 실습

- ❖ 로그인 화면 만들기(로그인 – “admin” , “1234” 인 경우)

<loginForm.jsp>



<loginResult.jsp>



비밀번호가 틀립니다.

초기화면

아이디가 없습니다.

초기화면

# request 객체를 이용한 데이터 전달 실습

- ❖ 로그인 화면 만들기(로그인 – “admin” , “1234” 인 경우)  
`<loginForm.jsp>`

```
<body>
    <h2> * 로그인 화면 *</h2>
    <form action="LoginResult.jsp">
        <table border="1">
            <tr>
                <td>아이디 </td>
                <td><input type="text" name="id"> </td>
            </tr>
            <tr>
                <td>비밀번호 </td>
                <td><input type="password" name="password"> </td>
            </tr>
            <tr style="text-align: center;">
                <td colspan="2"><input type="submit" value="확인"/> </td>
            </tr>
        </table>
    </form>
</body>
```

# request 객체를 이용한 데이터 전달 실습

- ❖ 로그인 화면 만들기(로그인 – “admin” , “1234” 인 경우)  
`<loginResult.jsp>`

```
<body>
<%
    String id = request.getParameter("id");
    String pw = request.getParameter("password");

    if (id.equals("admin")) {
        if (pw.equals("1234")){
            out.println(id + "님, 반갑습니다.");
        } else {
            out.println("비밀번호가 틀립니다.");
        }
    } else {
        out.println("아이디가 없습니다.");
    }
%>
<br><br>
<form action="LoginPage.jsp">
    <input type="submit" value="초기화면">
</form>
</body>
```

# request 객체를 이용한 데이터 전달 실습

- ❖ 로그인 화면 만들기(로그인 – “admin” , “1234” 인 경우)  
`<loginResult.jsp>`

```
<body>
<%
    String id = request.getParameter("id");
    String pw = request.getParameter("password");

    if (id.equals("admin")) {
        if (pw.equals("1234")){
            out.println(id + "님, 반갑습니다.");
        } else {
            out.println("비밀번호가 틀립니다.");
        }
    } else {
        out.println("아이디가 없습니다.");
    }
%>
<br><br>
<form action="LoginPage.jsp">
    <input type="submit" value="초기화면">
</form>
</body>
```

로직처리 코드

화면 출력 코드

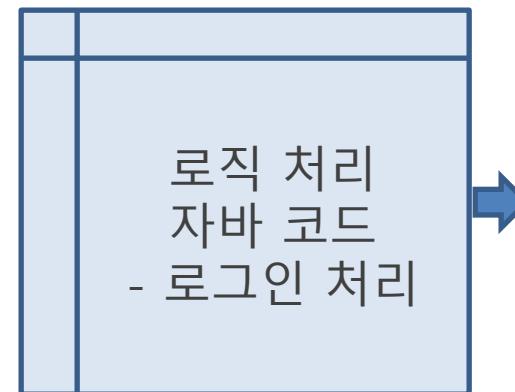
# request 객체를 이용한 데이터 전달 실습

- ❖ 로그인 화면 만들기(로그인 – “admin” , “1234” 인 경우)

<loginForm.jsp>

A screenshot of a web browser window showing the URL `localhost:8181/week5/loginForm.jsp`. The page title is "★ 로그인 화면 ★". It contains two input fields: "아이디" with value "sunyae" and "비밀번호" with value "...". Below the fields is a "로그인" button.

<loginProcess.jsp>



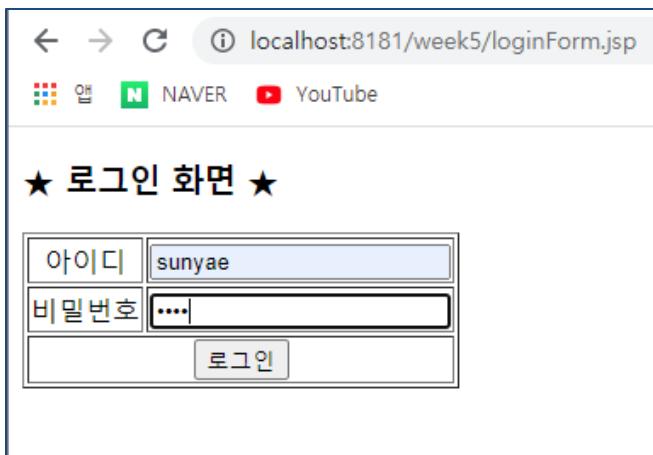
<loginResult.jsp>

A screenshot of a web browser window showing the URL `localhost:8181/week5/loginResult.jsp`. The page displays the message "admin님, 안녕하세요!" (Hello admin!). At the bottom, there is a "초기화면" (Initial Screen) button.

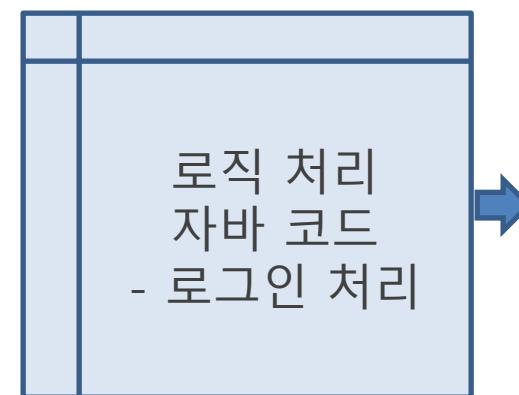
# request 객체를 이용한 데이터 전달 실습

- ❖ 로그인 화면 만들기(로그인 – “admin” , “1234” 인 경우)

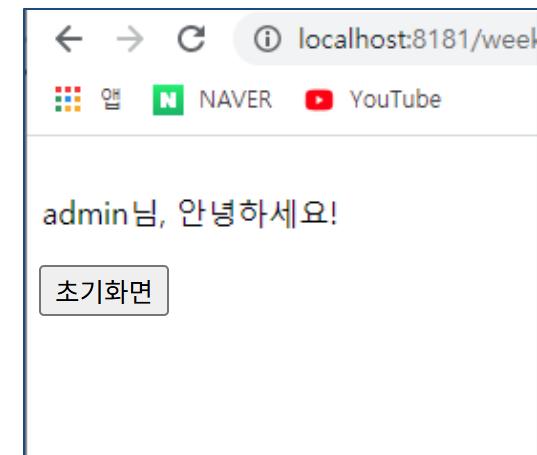
<loginForm.jsp>



<loginProcess.jsp>



<loginResult.jsp>



```
<body>
  <h2> * 로그인 화면 *</h2>
  <form action="LoginProcess.jsp">
    <table border="1">
      <tr>
        <td>아이디 </td>
        <td> <input type="text" name="id"> </td>
      </tr>
    </table>
  </form>
</body>
```

# request 객체를 이용한 데이터 전달 실습

- ❖ 로그인 화면 만들기(로그인 – “admin” , “1234” 인 경우)

## <loginProcess.jsp>

```
<body>
<%
    String id = request.getParameter("id");
    String pw = request.getParameter("password");

    String msg;

    if (id.equals("admin")) {
        if (pw.equals("1234")){
            msg = "admin님, 반갑습니다.";
        } else {
            msg = "비밀번호가 틀립니다.";
        }
    } else {
        msg = "아이디가 없습니다.";
    }

    request.setAttribute("msg", msg);
%>
```

로직처리 코드

화면 입력란 없이  
데이터를 다른  
페이지로 넘길 때  
**request** 객체 사용

```
<jsp:forward page="LoginResult.jsp"></jsp:forward>
</body>
```

# request 객체를 이용한 데이터 전달 실습

- ❖ 로그인 화면 만들기(로그인 – “admin” , “1234” 인 경우)  
**<loginResult.jsp>**

```
<body>
    <h2>* 로그인 결과 *</h2>
    <h3><%=request.getAttribute("msg") %></h3>
    <br>
    <form action="LoginPage.jsp">
        <input type="submit" value="초기화면">
    </form>
</body>
```

화면 출력 코드만