

2024년 상반기 K-디지털 트레이닝

# JSP의 이해

---

[KB] IT's Your Life

# 1 JSP의 기본요소

- ✔ HTML과 태그로 구성되어 있으며, 태그 안에 자바 코드를 삽입하여 구현
  
- ✔ JSP에서 사용 가능한 기본 태그 요소
  - HTML 요소
  - JSP 스크립팅 요소
  - JSP 표준 액션 태그 요소
  - EL 요소
  - 커스텀 태그 라이브러리 요소

# 1 JSP의 기본요소

## ✓ JSP 스크립팅 요소

- JSP 페이지가 서블릿으로 변환시 JSP 엔진에 의해 처리
- `<% %>` 형식을 가짐

스크립팅 요소	예제
Comment tag	<code>&lt;%-- 주석 -- %&gt;</code>
Directive tag	<code>&lt;%@ 지시자 %&gt;</code>
Declaration tag	<code>&lt;%! 자바 선언문 %&gt;</code>
Scriptlet tag	<code>&lt;% 자바 코드 %&gt;</code>
Expression tag	<code>&lt;%= 자바 표현식 %&gt;</code>

# 1 JSP의 기본요소

## ✓ 주석 태그

### ○ JSP 페이지 주석

<!-- JSP 태그 주석 --%>

### ○ 자바 코드 주석

```
<%  
/*  
    여러 줄 주석  
*/  
// 한 줄 주석  
%>
```

# 1 JSP의 기본요소

## ✓ JSP 지시어(Directive tag)

- 해당 페이지를 어떻게 처리할 것인지에 대한 설정 정보를 지시하는 용도

## ✓ page 지시어

- `<%@ page 속성명="속성값" 속성명2="속성값2" %>`
- 하나의 page 지시어를 사용하여 여러 개의 속성을 동시에 지정 가능
- 여러 개의 page 지시어를 사용하여 개별적으로 속성 지정 가능
- import 속성을 제외한 나머지 속성들은 중복 지정이 불가능

## 1 JSP의 기본요소

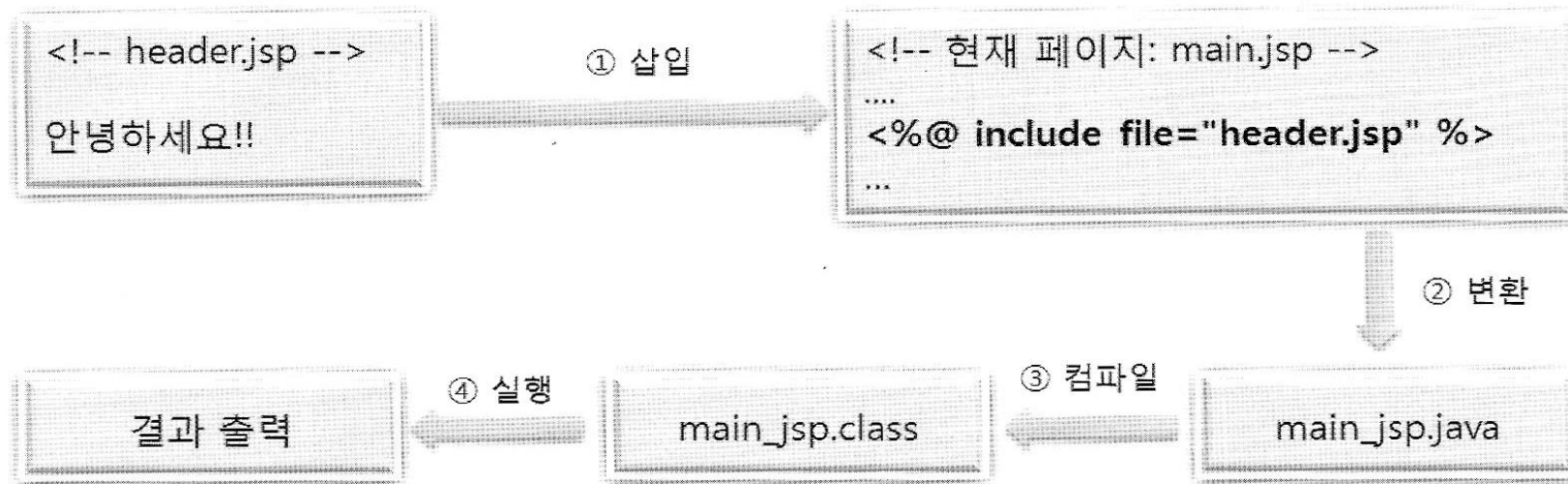
| 속성명          | 기본값        | 설 명                       | 예                            |
|--------------|------------|---------------------------|------------------------------|
| contentType  | text/html  | 응답 시의 MIME 타입             | contentType="text/html"      |
| import       | 없음         | import할 패키지 지정            | import="java.util.ArrayList" |
| session      | true       | HttpSession 사용 여부         | session="true"               |
| errorPage    | 없음         | 에러 발생 시 처리할 에러 페이지 지정     | errorPage="error.jsp"        |
| isErrorPage  | false      | 현재 페이지를 에러 페이지로 지정        | isErrorPage="false"          |
| language     | java       | 사용할 언어 설정                 | language="java"              |
| extends      | 없음         | 상속받은 클래스명 지정              | extends="패키지포함 클래스명"         |
| buffer       | 8kb        | 출력 버퍼 크기 지정               | buffer="8kb"                 |
| autoFlush    | true       | 출력 버퍼가 모두 찼을 경우에 처리 방법 지정 | autoFlush="true"             |
| isThreadSafe | true       | 스레드 동시 실행 여부              | isThreadSafe="true"          |
| info         | 없음         | 현재 페이지 정보 출력              | info="장바구니 페이지"              |
| pageEncoding | ISO-8859-1 | 현재 페이지의 문자 인코딩 지정         | pageEncoding="EUC-KR"        |
| isELIgnored  | false      | EL 표기법 사용 여부 지정           | isELIgnored="false"          |

# 1 JSP의 기본요소

## ✓ include 지시어

- 현재 페이지에서 다른 html 및 jsp 페이지를 삽입할 수 있는 방법 제공
- 각 페이지마다 공통적인 내용을 가지는 경우 매번 작성하지 않고 삽입
- footer, header, side 영역의 공통 내용 처리
- 정적인 방법으로 삽입

`<%@ include file="포함할 페이지"%>`



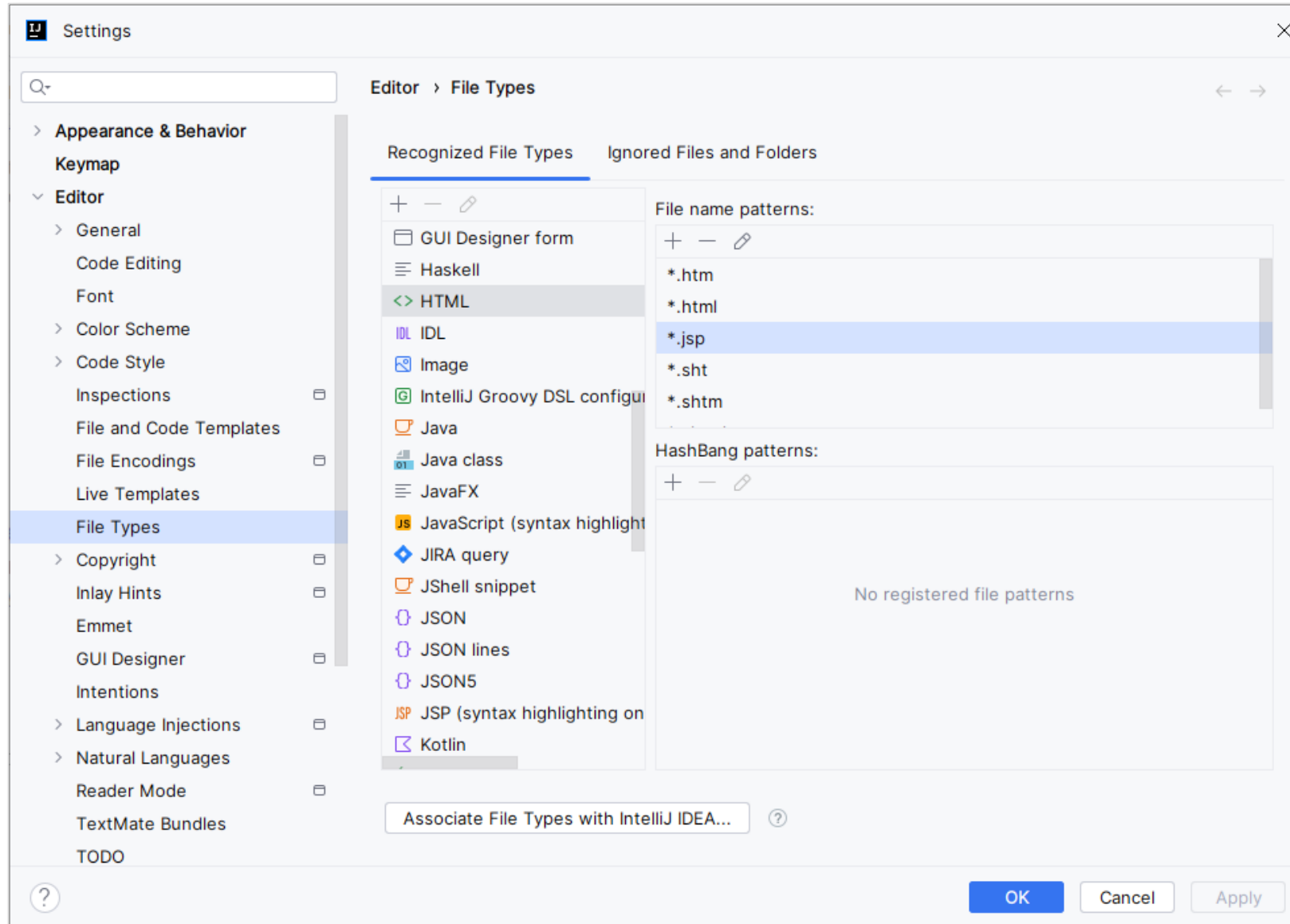
# 1 JSP의 기본요소

## ✓ include directive 태그 실습





## ✓ IntelliJ Community 버전인 경우



# 1 JSP의 기본요소

## copyright.jsp

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

```
<div>  
Copyright (c) ..... All rights reserved.  
</div>
```

# 1 JSP의 기본요소

## main.jsp

```
<%@ page contentType="text/html; charset=UTF-8"%>
<!DOCTYPE>
<html lang="ko">
<head>

    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>메인 화면 페이지</title>
</head>
<body>
    <h1>include 지시어 태그 실습</h1>
    내용 수정중입니다. 다음줄에 삽입이 됩니다. <br>

    <%@ include file="copyright.jsp" %>
</body>
</html>
```

# 1 JSP의 기본요소

## ✓ taglib 지시어

- JSP에서 외부 라이브러리로 만든 태그를 지정할 때 사용
- JSTL(JSP Standard Tag Library)

```
<%@ taglib uri="TLD 파일 URI" prefix="네임스페이스명" %>
```

# 1 JSP의 기본요소

## ✓ JSP 선언 태그(Declare tag)

- 자바의 멤버 변수와 메서드를 선언할 때 사용

```
<%!
```

```
    멤버 변수 선언
```

```
    메서드 선언;
```

```
%>
```

```
<%! int i; %>
```

```
<%! int i=0; %>
```

```
<%! public void setInfo(int i) {
```

```
    if( i < 3) ...
```

```
} %>
```

# 1 JSP의 기본요소

## ✓ 스크립틀릿 태그(Scriptlet tag)

<%  
자바 코드 문장;  
문장2;  
%>

```
<% if (Calendar.getInstance().get(Calendar.AM_PM) == Calendar.AM) {%}  
    Good Morning  
<% } else { %}  
    Good Afternoon  
<% } %>
```

# 1 JSP의 기본요소

## currentTime.jsp

```
<%@ page import="java.util.Date" %>
<%@ page contentType="text/html; charset=UTF-8"%>
<!DOCTYPE>
<html lang="ko">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>스크립트릿 실습</title>
</head>
<body>
  <h1>현재 날짜 출력 실습</h1>
  <%
    Date d = new Date();
  %>
  현재 날짜 : <%= d %>
</body>
</html>
```

# 1 JSP의 기본요소

## ✓ 표현식 태그(Expression tag)

<%=변수%>

<%= 리턴 값이 있는 메서드 %>

```
<%=(new java.util.Date()).toLocaleString() %>
```



# 1 JSP의 기본요소

## ☑ 내장 객체

| 내장 객체(내장 변수) | 설명   |
|--------------|--|
| request      | HttpServletRequest 객체 참조   |
| response     | HttpServletResponse 객체 참조  |
| out          | 웹 브라우저 출력에 사용되는 JspWriter 객체 참조                                  |
| session      | HttpSession 객체 참조(session="true"인 경우에만 사용 가능)                    |
| application  | ServletContext 객체 참조   |
| page         | 자바 클래스의 this와 동일   |
| config       | ServletConfig 객체 참조  |
| exception    | 발생되는 예외의 Throwable 객체에 대한 참조<br>(isErrorPage="true"인 경우에만 사용 가능) |

# 1 JSP의 기본요소

## ✓ request 객체

### ○ 서블릿의 HttpServlet Request 객체를 참조하는 변수

```
<body>
<h1>로그인 입력 화면</h1>
<form action="loginInfo.jsp" method="get">
  <fieldset>
    <legend>로그인 폼</legend>
    <ul>
      <li>
        <label for="userid">아이디</label>
        <input type="text" name="userid" >
      </li>
      <li>
        <label for="Password">비밀번호</label>
        <input type="password" name="Password" >
      </li>
      <li><input type="submit" value="전송"></li>
    </ul>
  </fieldset>
</form>
</body>
</html>
```

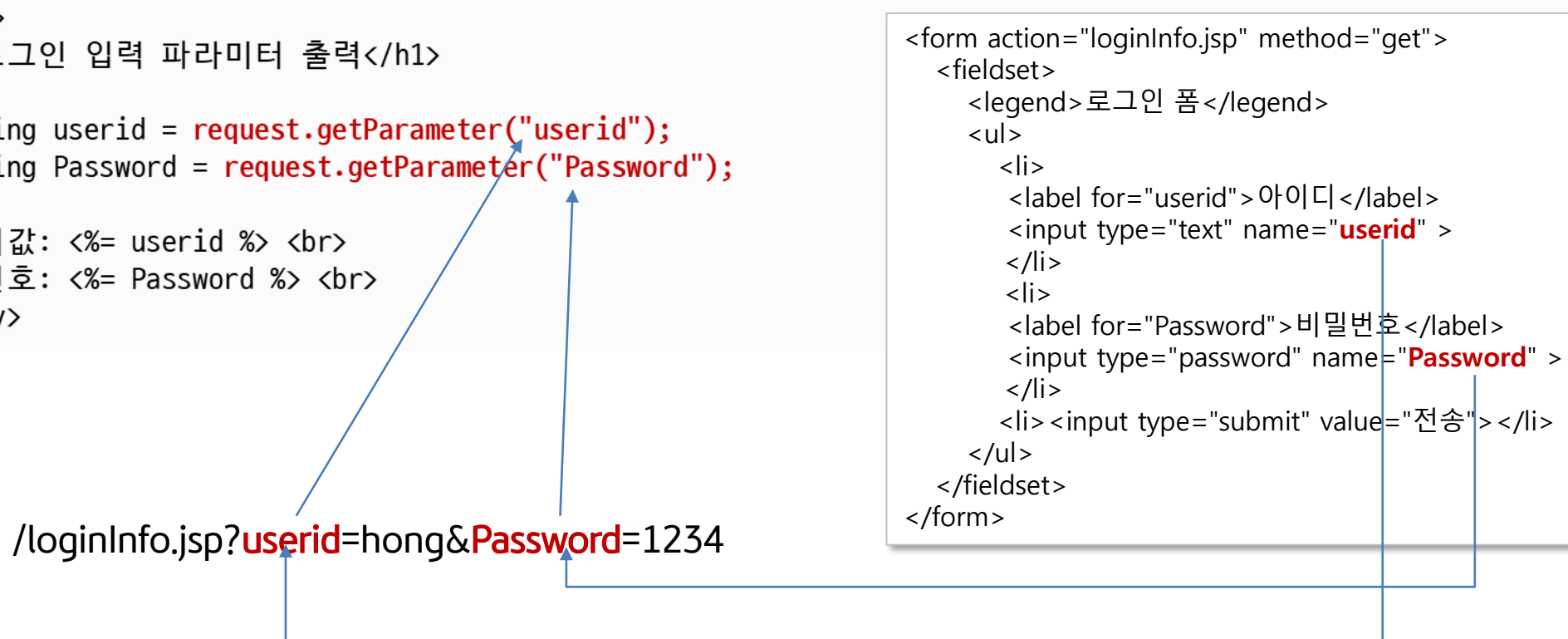
# 1 JSP의 기본요소

## request.jsp 내장 객체 실습

```
<body>
<h1>로그인 입력 파라미터 출력</h1>
<%
    String userid = request.getParameter("userid");
    String Password = request.getParameter("Password");
%>
아이디값: <%= userid %> <br>
비밀번호: <%= Password %> <br>
</body>
```

```
<form action="loginInfo.jsp" method="get">
  <fieldset>
    <legend>로그인 폼</legend>
    <ul>
      <li>
        <label for="userid">아이디</label>
        <input type="text" name="userid" >
      </li>
      <li>
        <label for="Password">비밀번호</label>
        <input type="password" name="Password" >
      </li>
      <li><input type="submit" value="전송"></li>
    </ul>
  </fieldset>
</form>
```

/loginInfo.jsp?userid=hong&Password=1234



# 1 JSP의 기본요소

## ✓ response 내장 객체

- HttpServletResponse 객체를 참조하는 변수
- 쿠키 저장, 요청 redirect 처리등 수행

# 1 JSP의 기본요소

## ✓ out 내장 객체

- 서블릿의 PrintWriter 객체와 동일한 기능의 JspWriter 객체를 참조
- 데이터 출력시 사용

- expression tag 이용

```
<%  
    String name="홍길동";  
%>  
<%=name%>
```

- out 내장 객체 이용

```
<%  
    String name="홍길동";  
    out.print(name);  
%>
```

# 1 JSP의 기본요소

## out.jsp

```
<body>
<%
    String name="홍길동";

    out.print("이것은 out 내장 객체로 출력 : " + name + "<br>" );
%>
이것은 expression tag로 출력 : <%= name %>
</body>
```

# 1 JSP의 기본요소

## ✓ application 내장 객체

- ServletContext 객체를 참조하는 변수
- 로그 및 파일 접근, application scope에 저장되는 속성 설정 등을 구현

### count.jsp

```
<body>
<h1>방문자수 설정하기 화면</h1>
<%! int count; %>
<%
    count++;

    application.setAttribute("countValue", count);
%>
현재 방문자수 : <%= count %>
</body>
```

# 1 JSP의 기본요소

## count\_view.jsp

```
<body>
<h1>방문자수 조회하기 화면</h1>
<%
    int count = (Integer)application.getAttribute("countValue");
%>
현재까지 총 방문자수: <%= count %>
</body>
```



# 1 JSP의 기본요소

## ✓ session 내장 객체

- HttpSession 객체를 참조하는 변수
- page 지시어에서 session="false"를 지정하면 사용 불가

# 1 JSP의 기본요소

## loginForm.html

```
<body>
<h1>session 내장 객체 실습</h1>
<form action="login.jsp" method="get">
  <fieldset>
    <legend>로그인 폼</legend>
    <ul>
      <li>
        <label for="userid">아이디</label>
        <input type="text" name="userid" >
      </li>
      <li>
        <label for="Password">비밀번호</label>
        <input type="password" name="Password" >
      </li>
      <li><input type="submit" value="전송"></li>
    </ul>
  </fieldset>
</form>
</body>
```

# 1 JSP의 기본요소

## login.jsp

```
<body>
<h1>로그인 정보 세션 저장</h1>
<%
    String id = request.getParameter("userid");
    if (id == null) {
        response.sendRedirect("loginForm.html");
    } else {
        String pw = request.getParameter("Password");
        session.setAttribute("userid", id);
        session.setAttribute("Password", pw);
        out.print("안녕하세요" + id);
        out.print("<a href='loginInfo.jsp'>정보보기</a>");
    }
%>
</body>
```

# 1 JSP의 기본요소

## loginInfo.jsp

```
<body>
<h1>로그인 정보 보기</h1>
<%
    String id = (String) session.getAttribute("userid");
    if (id == null) {
        response.sendRedirect("loginForm.html");

    } else {
        String pw = (String) session.getAttribute("Password");
        out.print("사용자 아이디값: " + id + "<br>");
        out.print("사용자 비밀번호값: " + pw + "<br>");
        out.print("<a href='logout.jsp'>로그 아웃</a>");

    }
%>
</body>
```

# 1 JSP의 기본요소

## logout.jsp

```
<body>
<%
    session.invalidate();
    response.sendRedirect("loginForm.html");
%>
</body>
```

# 1 JSP의 기본요소

## ✓ exception 객체

- JSP 페이지 내에서 발생한 예외 클래스인 `java.lang.Throwable` 객체를 참조하는 변수
- 예외 발생시 예외를 처리하는 JSP 페이지를 따로 구현할 목적으로 사용
- 반드시 `page` 지시어에서 `isErrorPage="true"`인 경우에만 사용 가능
  - 기본값은 `false`

# 1 JSP의 기본요소

## divide.jsp

```
<%@ page contentType="text/html; charset=UTF-8" errorPage="error.jsp"%>
<!DOCTYPE>
<html lang="ko">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>exception 실습</title>
</head>
<body>
<%
    // 0으로 나누어 예외 발생
    int n = 4/0;
%>
</body>
</html>
```

# 1 JSP의 기본요소

## error.jsp

```
<%@ page contentType="text/html; charset=UTF-8" isErrorPage="true"%>
<!DOCTYPE>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>exception 실습</title>
</head>
<body>
  <h1>divide.jsp 발생한 예외를 처리하는 페이지</h1>
  <%
    out.print("발생된 예외는 : " + exception.getMessage());
  %>
</body>
</html>
```



## 2 JSP 표준 액션 태그

### ✓ 표준 액션 태그

- 스크립트릿 태그 사용을 대체하기 위한 목적
- 기능이 제한적임 → JSTL로 보완

| 표준 액션 태그                     | 설명  |
|------------------------------|---|
| <jsp:useBean />              | 자바빈 컴포넌트를 사용하기 위한 액션 태그                     |
| <jsp:setProperty />          | 자바빈 인스턴스에 데이터를 저장하기 위한 액션 태그                |
| <jsp:getProperty />          | 자바빈 인스턴스에서 데이터를 얻기 위한 액션 태그                 |
| <b>&lt;jsp:include /&gt;</b> | <b>제공된 JSP 파일을 삽입하기 위한 액션 태그</b>            |
| <jsp:forward />              | 제공된 JSP 파일로 forward하기 위한 액션 태그              |
| <jsp:param />                | include 및 forward할 때 요청 파라미터를 추가하기 위한 액션 태그 |

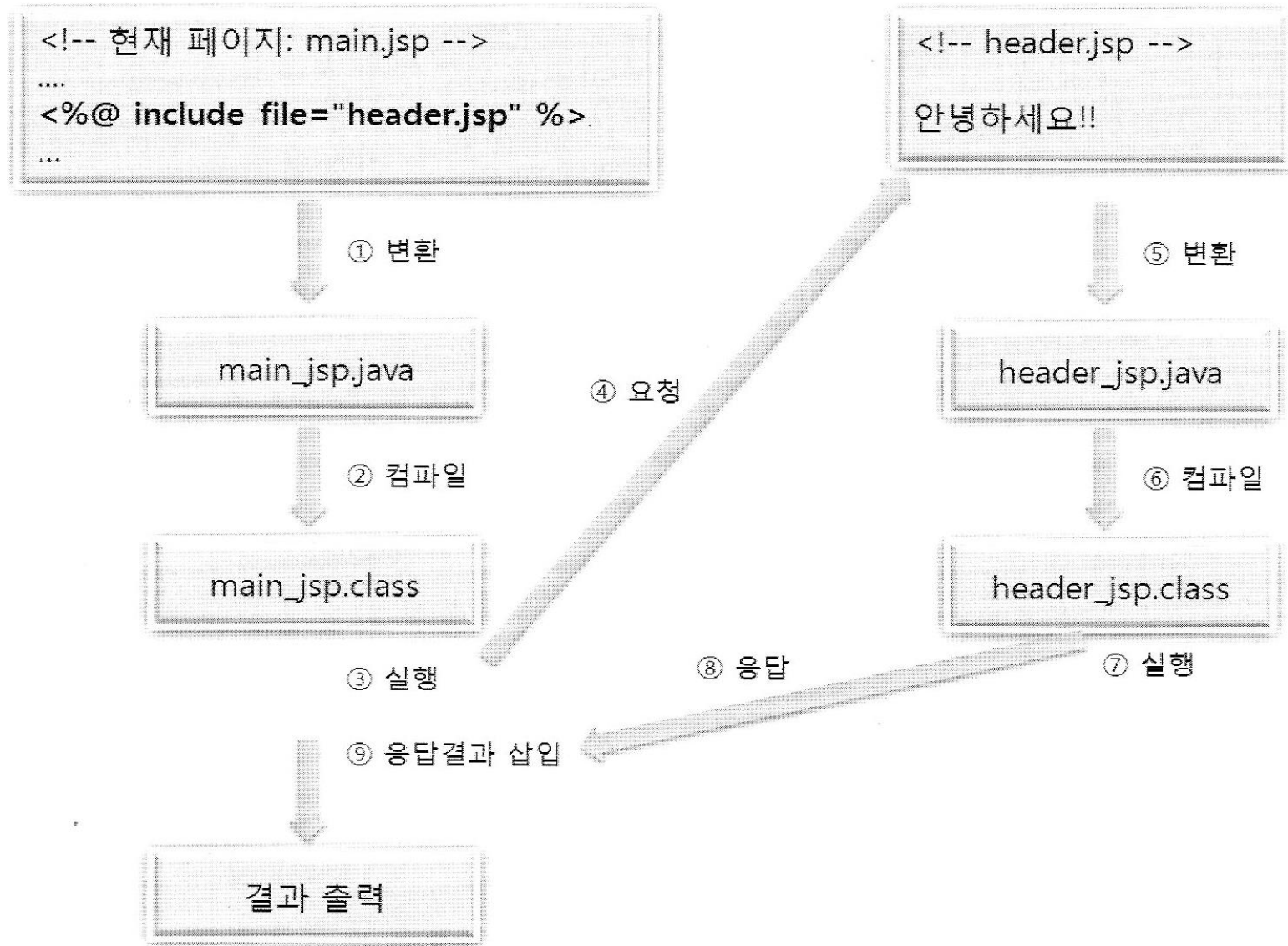
## 2 JSP 표준 액션 태그

### ✓ <jsp:include/> 액션태그

<jsp:include page="삽입될 페이지" flush="true"/>

- 동적으로 삽입이 이루어짐.
- 내용물이 소스 차원에 포함되는 것이 아니고 지정한 페이지의 실행 결과가 포함됨(메서드 호출과 유사)

## 2 JSP 표준 액션 태그



## 2 JSP 표준 액션 태그

### header.jsp

```
<%@page import="java.util.Calendar"%>
<%@ page contentType="text/html; charset=UTF-8"%>

<%
    Calendar cal = Calendar.getInstance();
    int hour = cal.get(Calendar.HOUR_OF_DAY);
    int minute = cal.get(Calendar.MINUTE);
    int second = cal.get(Calendar.SECOND);
%>
현재 시간은 <%= hour %> 시 <%= minute %> 분 <%= second %> 초 입니다.
```

## 2 JSP 표준 액션 태그

### main2.jsp

```
<%@ page contentType="text/html; charset=UTF-8"%>
<!DOCTYPE>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>include 액션 실습</title>
</head>
<body>
  <h1>include 액션 태그 실습</h1>
  현재 시간을 구하는 예제입니다. 다음줄에 삽입이 됩니다. <br>

  <jsp:include page="header.jsp" flush="true" />

</body>
</html>
```

## 2 JSP 표준 액션 태그

### ✔ 포함될 페이지에 임의의 파라미터 값을 지정하여 요청 가능

```
<jsp:include page="삽입될 페이지" flush="true">  
  <jsp:param name="파라미터 명" value="값" />  
  <jsp:param name="파라미터 명" value="값" />  
</jsp:include>
```

## 2 JSP 표준 액션 태그

### header2.jsp

```
<%@page import="java.util.Calendar"%>
<%@ page contentType="text/html; charset=UTF-8"%>

<%

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

    Calendar cal = Calendar.getInstance();
    int hour = cal.get(Calendar.HOUR_OF_DAY);
    int minute = cal.get(Calendar.MINUTE);
    int second = cal.get(Calendar.SECOND);
%>
안녕하세요 당신의 닉네임은 <%= nickName %>입니다.<br>
현재 시간은 <%= hour %> 시 <%= minute %> 분 <%= second %> 초 입니다.
```

## 2 JSP 표준 액션 태그

### main3.jsp

```
<%@ page contentType="text/html; charset=UTF-8"%>
<!DOCTYPE>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>include 액션 실습</title>
</head>
<body>
  <h1>include 태그 실습</h1>
  현재 시간을 구하는 예제입니다. 다음줄에 삽입이 됩니다. <br>

  <jsp:include page="header2.jsp" flush="true" >
    <jsp:param name="nickName" value="Mr Hong" />
  </jsp:include>
</body>
</html>
```