

- Session 개념 – HttpSession api

1> 동일 브라우저에서 실행하는 동안 데이터 공유

= 브라우저 실행동안 세션 저장 데이터는 서버 내부 메모리 유지

(AServlet – doGet() 지역변수 선언 메모리 유지 메소드 종료까지)

2> 데이터는 정수 실수 문자열 논리값 자바객체 배열등 가능

3> 브라우저 열고 서블릿 실행시 최초로 세션 생성(브라우저당 1개세션)/세션리턴

4> 브라우저 닫으면 세션 자동 삭제

5> 브라우저 열고 30분 동안 액션 없으면 세션 자동 삭제

6> 세션 내부의 특정 데이터값 즉각 삭제

7> 세션의 모든 데이터값 즉각 삭제

8> 세션 시간 연장

<p>a(최초 실행)</p> <pre>HttpSession session= request.getSession(); ==>세션 생성 session.setAttribute("test", new A()); session.setAttribute("test2", new B());</pre>	<p>b(두번째 실행)</p> <pre>HttpSession session= request.getSession(); ==> 세션객체 리턴 A a1 = (A)session.getAttribute("test"); session.removeAttribute("test"); session.invalidate(); session.setMaxInactiveInterval(초);</pre>
--	--

- jsp – java server page

1> html페이지내부에 서버에서 실행 가능한 자바 문장 포함.

2> 서버에서 실행 가능한 자바 문장은 서블릿에서 구현하는 문장들임

3> html태그 + jsp 태그 + 내장객체 + el 로 구성

<p>jsp기본태그</p>	<pre><% 자바문장; {} %> <%= %> <%! 멤버변수나 메소드 선언 %> <%-- 주석 --%> <%@ page import="" contentType="text/html;charste=utf-8" - jsp파일 첫태그 <%@ include file="b.jsp" %></pre>
----------------	---

	내장객체
jsp액션	
jsp커스텀태그	

- 내장객체-자주 사용 객체 미리 생성 제공 . _jspService메소드 지역변수 생성

<%! public void a(){ out.... } %>-->error

<% out.println(); %>

request	javax.servlet.http. HttpServletRequest타입	String 1개 =request.getParameter("id") String[] = request.getParameterValues("id") request.setAttribute() request.getAttribute()
out	javax.servlet.jsp .JspWriter타입	response.setContentType ("text/html;charset=utf-8") out = response.getWriter(); <% out.println("test"); %> <%= "test"%>
response	javax.servlet.http. HttpServletResponse타입	<%@ page contentType="text/html;charset=utf-8" %> response.addCookie(쿠키객체);
session	javax.servlet.http .HttpSession타입	request.getSession() 대신 생성 객체 session.setAttribute session.getAttribute session.removeAttribute
application	javax.servlet.ServletContext타입	현재 파일 저장 컨텍스트 서버종료시 데이터 공유 ServletContext application = getServletContext(); context.setAttribute().. get / remove

exception	java.lang.Throwable타입	jsp 실행 예외발생 객체 미리 생성
page	*.jsp	this 동일 개념 - 클래스
pageContext	javax.servlet.jsp.PageContext	다른 내장객체 전달
config	javax.servlet.ServletConfig타입	현재서블릿이나 jsp파일로 전달하는 데이터 받기 위한 객체

- 서블릿 데이터 공유

요청처리-응답완료	브라우저 실행동안	서버 실행동안
doGet(..request request.setAttribute("", 객체) request.getAttribute("") request.removeAttribute("")	doGet(..request HttpSession session = request.getSession(); session.setAttribute("", 객체) session.getAttribute("") session.removeAttribute("")	doGet(..request ServletContext application = getServletContext(); application.setAttribute("", 객체) application.getAttribute("") application.removeAttribute("")

- jsp 데이터 공유(내장객체)

요청처리-응답완료	브라우저 실행동안	서버 실행동안
request.setAttribute("", 객체) request.getAttribute("") request.removeAttribute("")	session.setAttribute("", 객체) session.getAttribute("") session.removeAttribute("")	application.setAttribute("", 객체) application.getAttribute("") application.removeAttribute("")

MemberDTO객체 공유

request1.jsp / request2.jsp

a.jsp?su=100 <%@page errorPage="b.jsp" try{ Integer.parseInt(request.getParameter("su")); }catch(NumberFormatException e){...} a.jsp a.jsp?su=백	b.jsp <%@ page isErrorPage="true" %> exception 내장객체 사용 exception.printStackTrace(); exception.getMessage() excetion.toString()
--	--

-오류처리

exception-자바로직 처리 도중 발생 오류 객체

404 http프로토콜 정한 오류	http:.../a
405	get->doPost post->doGet get+post-> jsp

- web.xml

ServersW...Wweb.xml	모든 다이나믹 웹 프로젝트 공통 적용 사항 정의-30분 세션설정
다이나믹웹프로젝트WWEBCONTENTW WEB-INFWweb.xml	현재 다이나믹 웹 프로젝트 적용

<http://ip:port/jsptest/c.jsp?num=100>-->ok

<http://ip:port/jsptest/c.jsp?num=3.14> -> 자동 e500.jsp

<http://ip:port/jsptest/c.jsp?num=일>

<http://ip:port/jsptest/c.jsp>

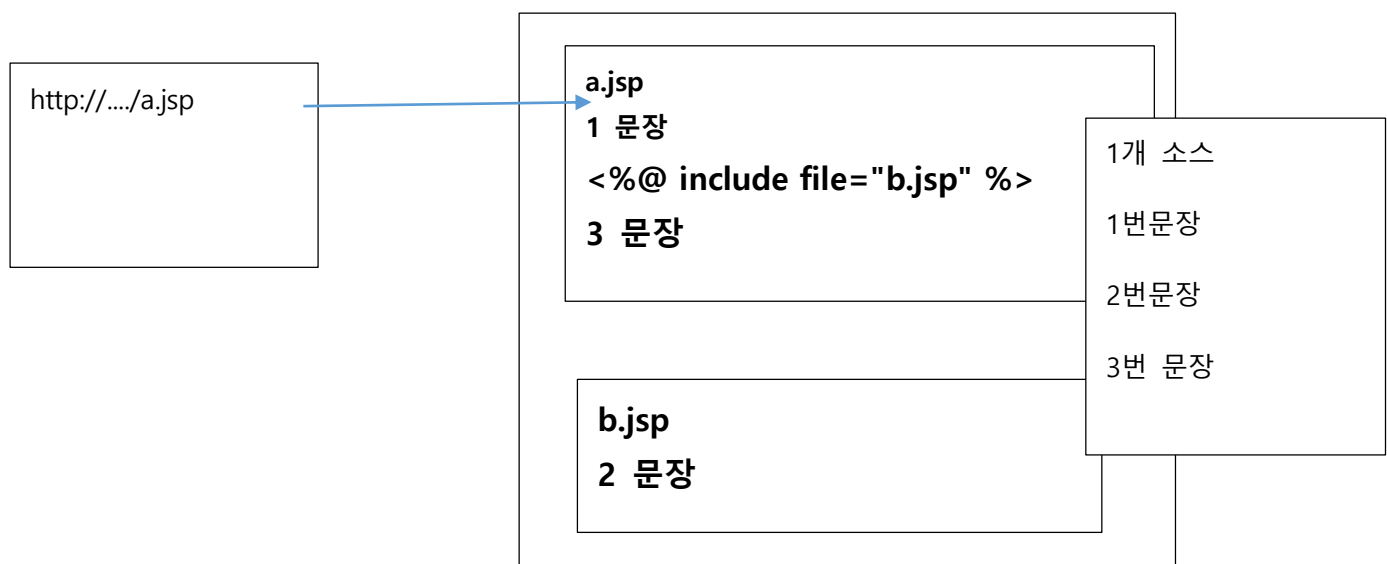
13장 action -"동작"- 자바 코드를 없애는 액션 태그

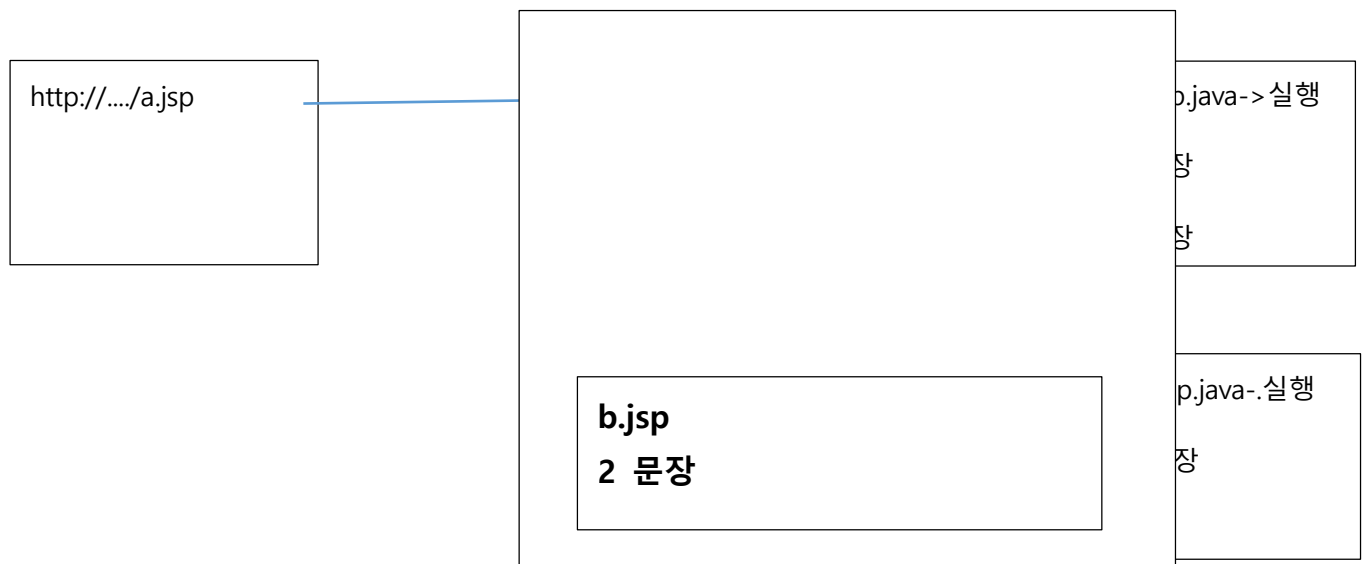
<code><jsp:xxxxx /></code> <code><jsp:xxxxx > 다른 태그 포함</code> <code></jsp:xxx></code>	
<code><jsp:forward page="이동파일명" /></code> <code><jsp:include page="포함파일명" /></code> <code><%@ include file="포함파일명" %></code>	<pre>RequestDispatcher rd = request. getRequestDispatcher("이동,포함파일명"); request.setAttribute("", 객체);/getAtt rd.forward(request, ...) rd.include (request, ...)</pre>
<code><jsp:param name="" value="" /></code>	
<code><jsp:useBean /></code>	jsp 자바객체 사용 쉽게 표현 =
<code><jsp:setProperty /></code>	
<code><jsp:getProperty /></code>	

- 인클루드

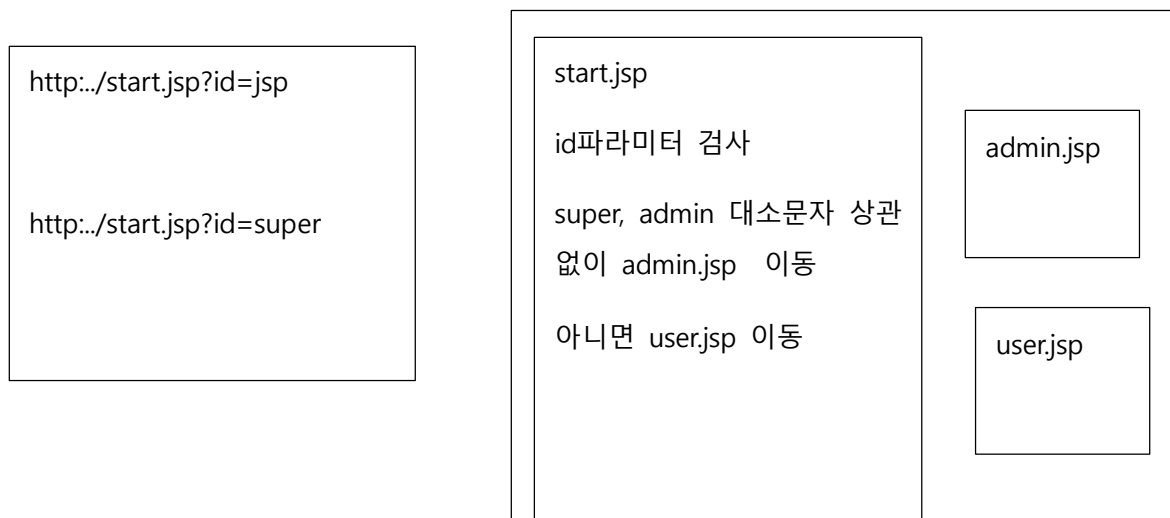
`<%@ include file="포함파일명" %>` -> tomcat 지시. jsp 내부 서블릿 소스 변환 시점에서 포함. 1개 자바 소스 (정적 include)

`<jsp:include page="포함파일명" />`

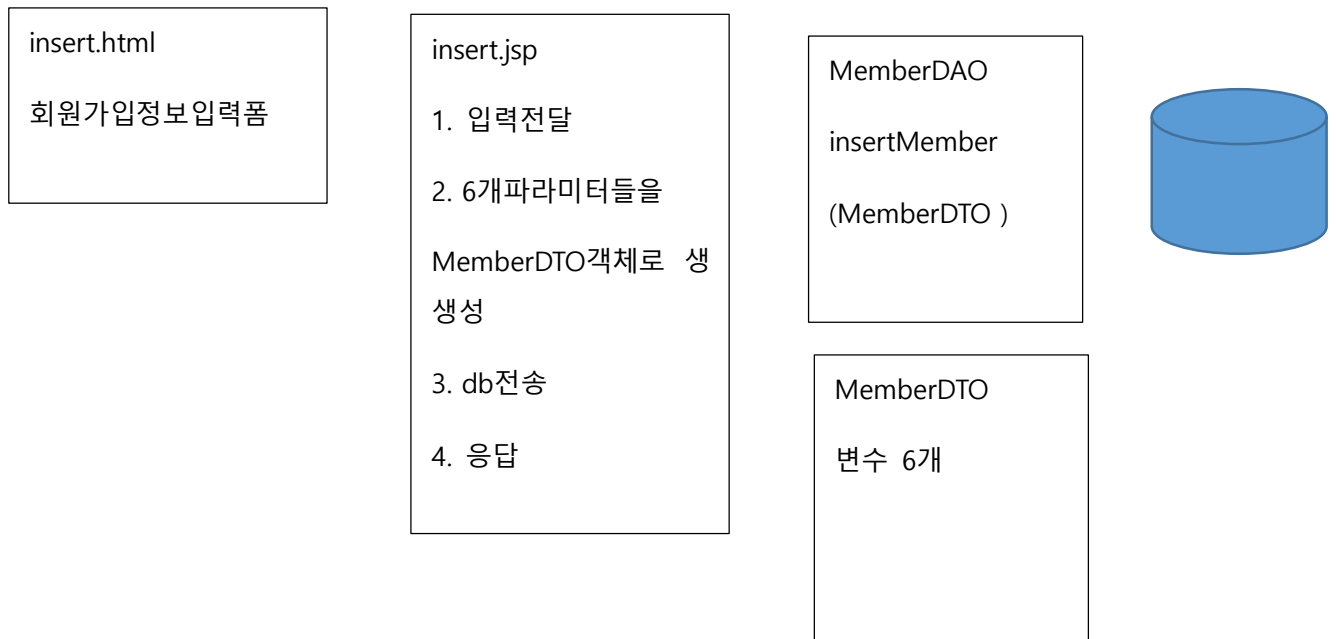




- forward



1> <jsp:forward page="" > <jsp:param name="" value="" /> </jsp:forward>	1> request.setAttribute("", 객체); <jsp:forward page="" />
2> request.getParameter("name값")	2> Object o = request.getAttribute("이름")



HTML파일 내부 자바문장 포함 형태

웹디자이너+웹개발자 협업 = JSP

자바 문장 최소화

```
MemberDTO dto= new MemberDTO();
```

```
dto.setid("jsp");
```

```
out.println(dto.getId());
```

객체 생성문장 없애고 태그

```
<jsp:useBean id="dto" class="dto.MemberDTO" />
```

```
<jsp:setProperty name="dto" property="id" value="jsp" />
```

```
<jsp:getProperty name="dto" property="id" />
```

2. 6개파라미터들을

MemberDTO객체로 생성

```
new MemberDTO(6개요청파라미터..)
```