

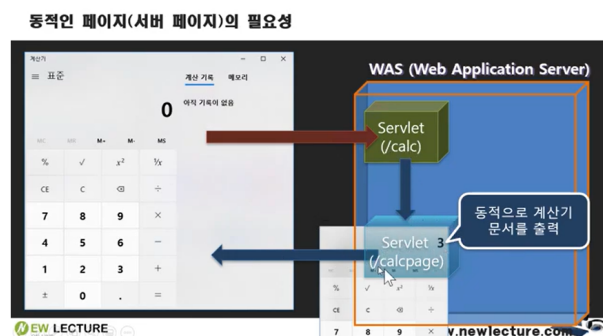
Servlet / JSP - 3

서버에서 페이지 전환해주기(redirect)

- `response.sendRedirect("calc2.html");`
 - 사용자에게 페이지를 전환해준다.
 - 동일한 주소로 전환하면 사용자가 페이지가 전환된 것을 눈치채기 힘들다

동적인 페이지(서버 페이지)의 필요성

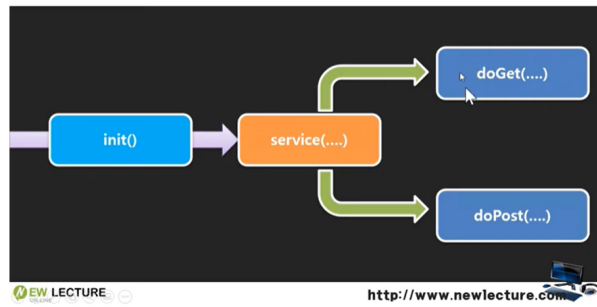
- 사용자가 입력한 내용을 서버에서 페이지를 만들 때 끼워 넣어야 함.
- 정적인 문서가 아니라 사용자가 누적인 값이 있으면 그 것을 심어서 보여줘야함.
 - html 문서로는 안되고, 동적으로 만들어주는 것이 필요하다



- 요청이 있을때 부랴부랴 만들어주는 문서 (동적 문서)

GET / POST 에 특화된 서비스 함수

요청 Method에 따른 특화된 메소드들을 이용하기



JSP 시작

서블릿 출력 코드에서 HTML을 출력하는 방법

```
<html>
<head></head>
<body>
  Calculator
  <form action="add" method="get">
    <input type="text" value="x" />
    <input type="text" value="y" />
    <input type="submit" value="Sum" />
  </form>
</body>
</html>
```

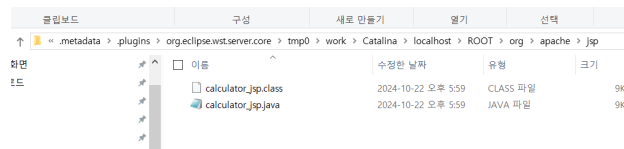
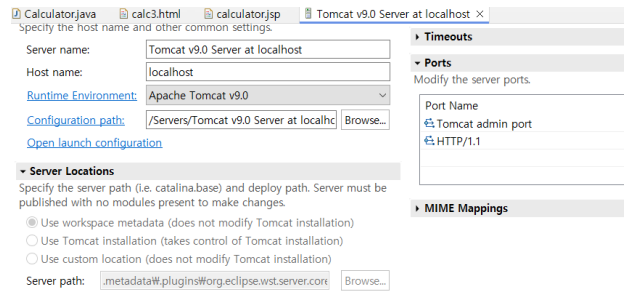
The image shows a code editor with HTML and Java code. The HTML part defines a simple calculator form with two input fields for 'x' and 'y', and a 'Sum' button. The Java part uses `out.write()` to output the HTML. The URL `http://www.newlecture.com` is visible at the bottom right.

- 서블릿 출력 코드에서 매번 이렇게 사용하기에는 불편함이 따른다.

Jasper가 만들어주는 서블릿 출력 코드

```
<html>
<head></head>
<body>
  Calculator
  <form action="add" method="get">
    <input type="text" value="x" />
    <input type="text" value="y" />
    <input type="submit" value="Sum" />
  </form>
</body>
</html>
```

The image shows the same HTML code as before, but with a green box labeled `add.jsp` next to it. A red arrow points from the `add.jsp` box to a green box containing the file path: `토크트 설치 디렉토리\#work\Catalina\localhost\newlec\org\Wapache\jsp\add.jsp.java`. The word `Jasper` is written next to the arrow.



jsp의 work 디렉토리

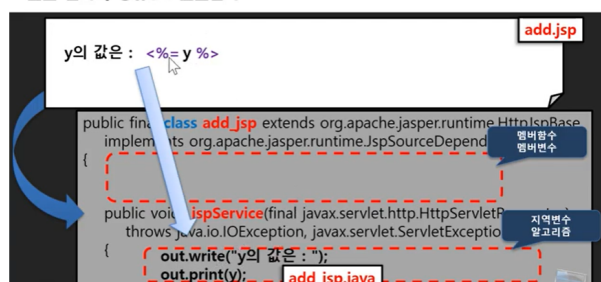
- 이것은 “코드”이고, 출력해서는 안될때?

○

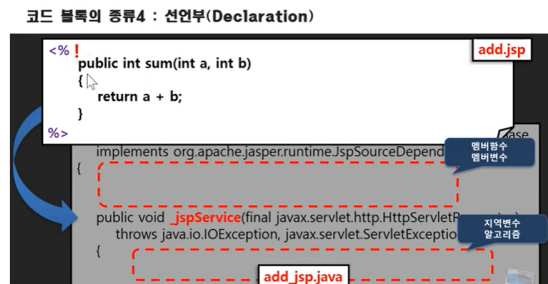
Jasper를 이용한 코드 작성 방법

```
<%
    int x = 3;
    int y = 4;
%>
<html>
...
<form action="add.jsp" method="get">
    <ul>
        <li><label for="x">X :</label><input name="x" /></li>
        <li><label for="y">Y :</label><input name="y" /></li>
    </ul>
    <p><input type="submit" value="Sum" /></p>
</form>
</body>
</html>
```

코드 블록의 종류3 : 코드블록



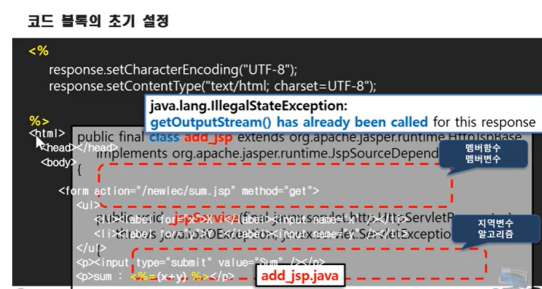
- 일반적인 코드블록은 서비스함수에 들어간다
- 그렇지 않고 멤버 함수로 들어가고 싶을때는 느낌표(!)를 쓰면 된다
 -



- page 지시자 - 초기설정을 위한
 - @ 사용

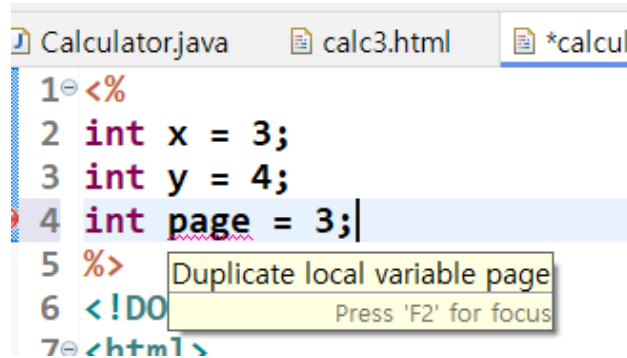


- 코드에서 write를 하기 전에 설정하기 위해 코드블록으로는 해결이 되지 않는다
-



JSP의 내장객체 알아보기

•



- jsp가 만들어낸 servlet에는 내가 모르는 변수들을 만들어놓은게 있을 수 있다.
- jsp가 만들어놓은 변수들을 내장객체라고 한다.

JSP 내장 객체

```
public void _jspService(final javax.servlet.http.HttpServletRequest request, final
javax.servlet.http.HttpServletResponse response)
    throws java.io.IOException, javax.servlet.ServletException {

    final javax.servlet.jsp.PageContext pageContext;
    javax.servlet.http.HttpSession session = null;
    final javax.servlet.ServletContext application;
    final javax.servlet.ServletConfig config;
    javax.servlet.jsp.JspWriter out = null;
    final java.lang.Object page = this;
    javax.servlet.jsp.JspWriter _jspx_out = null;
    javax.servlet.jsp.PageContext _jspx_page_context = null;
```

내장 객체 request : HttpServletRequest

메서드	설명
getParameterNames()	사용자가 전달한 키들을 Enumeration 객체로 반환
getParameter(name)	사용자가 전달한 name과 일치하는 값을 반환
getParameterValues(name)	사용자가 전달한 name과 일치하는 값을 배열형식으로 반환
getCookies()	클라이언트에서 전달한 쿠키를 배열 형식으로 반환
getMethod()	현재 요청방식이 GET인지 POST인지를 문자열로 반환
getSession()	현재 세션 객체를 반환
getRemoteAddr()	클라이언트의 IP 주소를 반환
getProtocol()	현재 서버의 프로토콜을 문자열로 반환
setCharacterEncoding()	현재 JSP로 전달되는 내용을 지정한 문자셋으로 변환해준다.
getHeaderNames()	현재 요청이 가지는 헤더의 이름들을 반환
getHeaders(name)	현재 요청한 헤더에서 지정한 이름의 모든 값들을 반환
getQueryString()	현재 요청에 포함된 쿼리문자열을 반환

내장 객체 response : HttpServletResponse

메서드	설명
setContentType(type)	컨텐츠 형식을 설정
setHeader(name, value)	클라이언트에게 헤더로 전달할 값을 설정
setDateHeader(name, date)	클라이언트에게 헤더로 전달할 날짜를 설정
sendError(status, msg)	클라이언트에게 에러코드와 메시지를 전달
sendRedirect(url)	클라이언트 요청을 다른 페이지로 전달
addCookie(cookie)	클라이언트에게 전달할 쿠키를 설정
encodeURL(url)	URL로 유효하지 않은 문자를 인코딩
setStatus(sc)	상태 코드를 설정

내장 객체 session : javax.servlet.http.HttpSession

메서드	설명
getId()	각 접속에 대한 세션 고유 ID를 문자열형태로 반환
getCreationTime()	세션이 생성된 시간을 밀리세컨드값으로 반환
getLastAccessedTime()	현재 세션으로 마지막 작업한 시간을 밀리세컨드값으로 반환
getMaxInactiveInterval()	세션 유지 시간을 초로 반환
setMaxInactiveInterval(t)	세션 유효시간을 t에 설정된 초 값으로 설정
invalidate()	현재 세션을 종료. 세션과 관련한 값을 모두 지움
getAttribute(attr)	문자열 attr로 설정된 세션값을 object형태로 반환
setAttribute(name, attr)	문자열 name으로 attr을 설정
removeAttribute(name)	세션에 설정한 속성 값을 삭제

내장 객체 application : javax.servlet.ServletContext

메서드	설명
setAttribute(name, value)	application 범위의 값 설정
getAttribute(name)	application 범위의 값 얻기
getRealPath(path)	실제 물리 경로를 반환
getResource(path)	path 경로의 리소스를 가리키는 URL을 반환
getServerInfo()	현재 요청방식이 GET인지 POST인지를 문자열로 반환
getSession()	현재 세션 객체를 반환
getRemoteAddr()	클라이언트의 IP 주소를 반환
getProtocol()	현재 서버의 프로토콜을 문자열로 반환
getCharacterEncoding()	현재 JSP로 전달되는 내용을 지정한 문자셋으로 변환해준다.

JSP로 만드는 Hello 서블릿(간단예제)

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%
String cnt_ = request.getParameter("cnt");

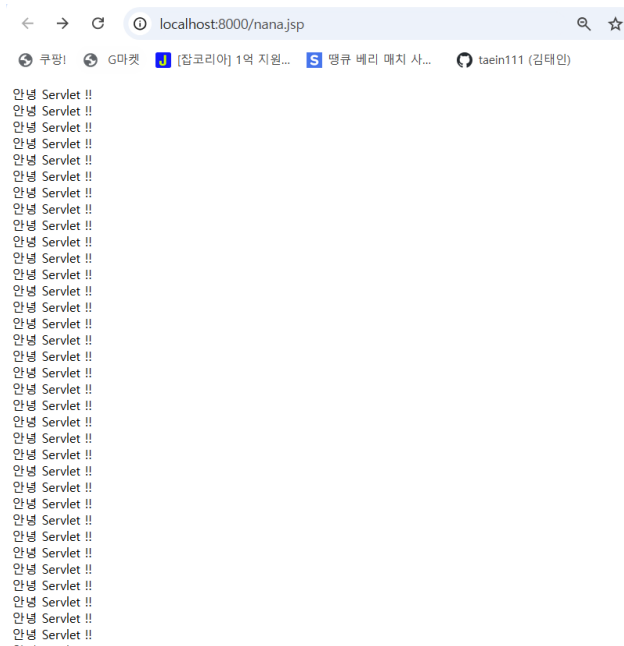
int cnt = 100;

if(cnt_ != null && !cnt_.equals(""))
    cnt = Integer.parseInt(cnt_);
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
```

```

<title>Insert title here</title>
</head>
<body>
    <% for(int i=0; i<cnt; i++){ %>
        안녕 Servlet !!<br >
    <% }%>
</body>
</html>

```



스파게티 코드를 만드는 JSP

-