

2020년도 2학기

웹프로젝트실습 과목

- 5주차 -

장용미

H.P: 010-3309-4849

E-MAIL: changmi29@dongyang.ac.kr

[eClass의 쪽지](#)



프로젝트로 배우는 자바 웹 프로그래밍

Servlet, JSP, JDBC

황희정 지음

Chapter 06. JSP 내장객체

목차

1. JSP 내장객체의 개요
2. request
3. response
4. out
5. session
6. 그 밖의 내장객체
7. JSP 내장객체와 속성 관리
8. [기본실습] JSP 내장객체 : 세션을 이용한 장바구니 기능
9. [응용실습] JSP 내장객체 : 트위터 구현

학습목표

- JSP 내장객체 개념과 구조적 특징을 이해한다.
- 내장객체의 종류와 각각의 주요 기능에 대해 살펴본다.
- 내장객체의 속성 관리 기법을 이해하고 활용한다.

01. JSP 내장객체 개요

JSP 내장객체란?

- JSP 내장객체란 'JSP 내에서 선언하지 않고 사용할 수 있는 객체'라는 의미에서 붙여진 이름.
- 구조적으로는 JSP가 서블릿 형태로 자동 변환된 코드 내에 포함되어 있는 멤버변수, 메서드 매개변수 등의 각종 참조 변수(객체)를 말함.

[표 6-1] JSP 내장객체

9개

참조 변수 이름(내장객체)	자바 클래스	주요 역할
request	javax.servlet.http.HttpServletRequest	HTML 폼 요소의 선택 값 등 사용자 입력 정보를 읽으려고 사용한다.
response	javax.servlet.http.HttpServletResponse	사용자 요청에 대한 응답을 처리하려고 사용한다.
pageContext	javax.servlet.jsp.PageContext	현재 JSP 실행에 대한 context 정보를 참조하려고 사용한다.
session	javax.servlet.http.HttpSession	클라이언트의 세션 정보를 처리하려고 사용한다.
application	javax.servlet.ServletContext	웹 서버의 애플리케이션 처리와 관련된 정보를 참조하려고 사용한다.
out	javax.servlet.jsp.JspWriter	사용자에게 전달하기 위한 output 스트림을 처리하려고 사용한다.
config	javax.servlet.ServletConfig	현재 JSP의 초기화 환경을 처리하려고 사용한다.
page	java.lang.Object	현재 JSP의 클래스 정보를 보려고 사용한다.
exception	java.lang.Throwable	예외 처리를 하려고 사용한다.

01. JSP 내장객체 개요

1. JSP 내장객체의 구조적 특징

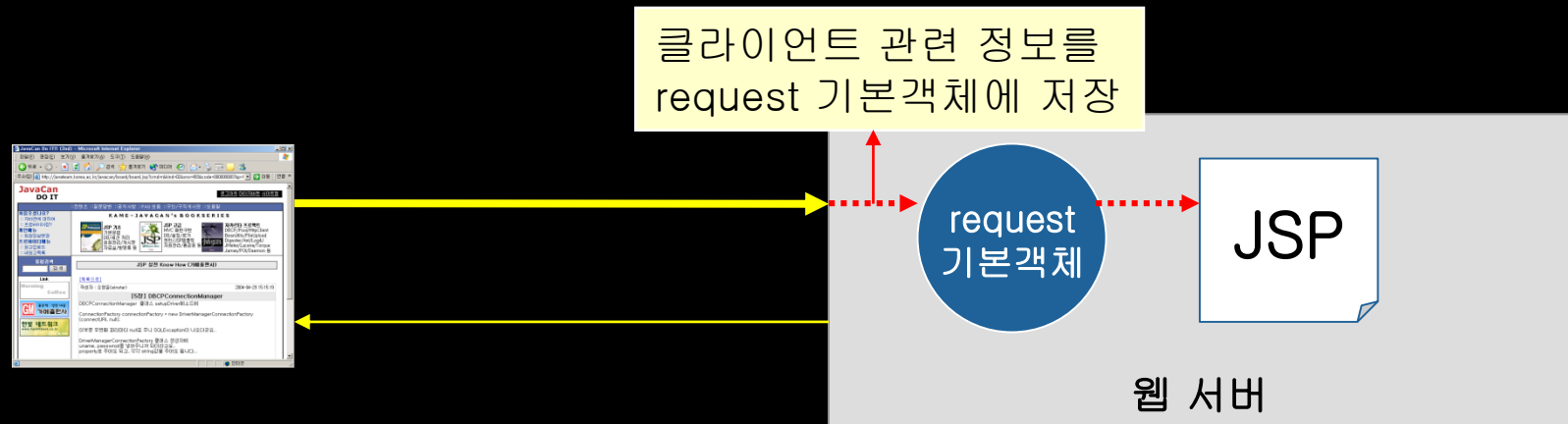
- 서블릿으로 변경된 JSP 코드는 모두 `_jspService()` 메서드에 위치함.
- 메서드 매개변수인 `request`, `response` 를 비롯한 `pageContext`, `session`, `application`, `page`, `config`, `out` 등 메서드 내에서 참조할 수 있는 참조변수들이 내장객체가 됨.

```
public void _jspService(HttpServletRequest request, HttpServletResponse response) throws java.io.IOException, ServletException {  
    JspFactory _jspxFactory = null;  
    javax.servlet.jsp.PageContext pageContext = null;  
    HttpSession session = null;  
    ServletConfig config = null;  
    JspWriter out = null;  
    Object page = this;  
    JspWriter _jspx_out = null;  
    try {  
        pageContext = _jspxFactory.getPageContext(this, request, response, null, true, 8192, true);  
        application = pageContext.getServletContext();  
        config = pageContext.getServletConfig();  
        session = pageContext.getSession();  
        out = pageContext.getOut();  
        ...  
        사용자 작성 JSP 코드가 오는 위치....  
    }  
}
```

2. 내장객체를 이용한 속성 관리 기법

- 내장객체가 단순히 특정한 기능을 제공하는 컨테이너 관리 객체라는 점 외에도 한 가지 특징이 있다. 바로 page, request, session, application 내장객체를 이용한 속성 관리 기법이다. 이들 내장객체는 각자 지정된 생명주기가 있으며 `setAttribute()`, `getAttribute()`라는 메서드를 통해 해당 생명주기 동안 자바 객체를 유지하는 기능을 제공한다.
- 속성 관리 기법은 '7절. JSP 내장객체와 속성 관리'에서 자세히 살펴볼 것이다.

request 기본 객체



request 기본 객체가 제공하는 기능

- 클라이언트 ■ 웹 브라우저 ■ 와 관련된 정보 읽기 기능
- 서버와 관련된 정보 읽기 기능
- 클라이언트가 전송한 요청 파라미터 읽기 기능
- 클라이언트가 전송한 요청 헤더 읽기 기능
- 클라이언트가 전송한 쿠키 읽기 기능
- 속성 처리 기능

02. request

■ request 내장객체

- request는 사용자 요청과 관련된 기능을 제공하는 내장객체로 `javax.servlet.http.HttpServletRequest` 클래스에 대한 참조 변수임.
- 주로 클라이언트에서 서버로 전달되는 정보를 처리하기 위해 사용한다.
- 대표적으로 HTML 폼을 통해 입력된 값을 JSP에서 가져올 때 사용함.

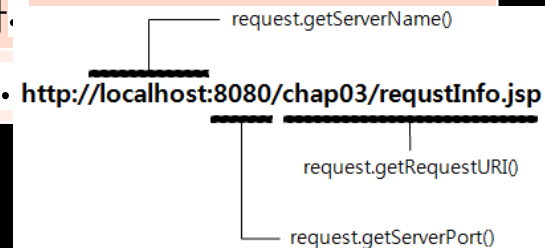
[표 6-2] request 주요 메서드

메서드	설명
<code>getParameterNames()</code>	현재 요청에 포함된 매개변수의 이름을 열거(Enumeration) 형태로 넘겨준다.
<code>getParameter(name)</code>	문자열 <code>name</code> 과 이름이 같은 매개변수의 값을 가져온다.
<code>getParameterValues(name)</code>	문자열 <code>name</code> 과 이름이 같은 매개변수의 값을 배열 형태로 가져온다. checkbox, multiple list 등에 주로 사용한다.
<code>getCookies()</code>	모든 쿠키 값을 <code>javax.servlet.http.Cookie</code> 의 배열 형태로 가져온다.
<code>getMethod()</code>	현재 요청이 GET이나 POST 형태로 가져온다.
<code>getSession()</code>	현재 세션 객체를 가져온다.
<code>getRemoteAddr()</code>	클라이언트의 IP 주소를 알려준다.
<code>getProtocol()</code>	현재 서버의 프로토콜을 문자열 형태로 알려준다.
<code>setCharacterEncoding()</code>	현재 JSP로 전달되는 내용을 지정한 캐릭터셋으로 변환해준다. HTML 폼에서 한글 입력을 정상적으로 처리해주려면 반드시 필요하다.

request 객체 주요메서드

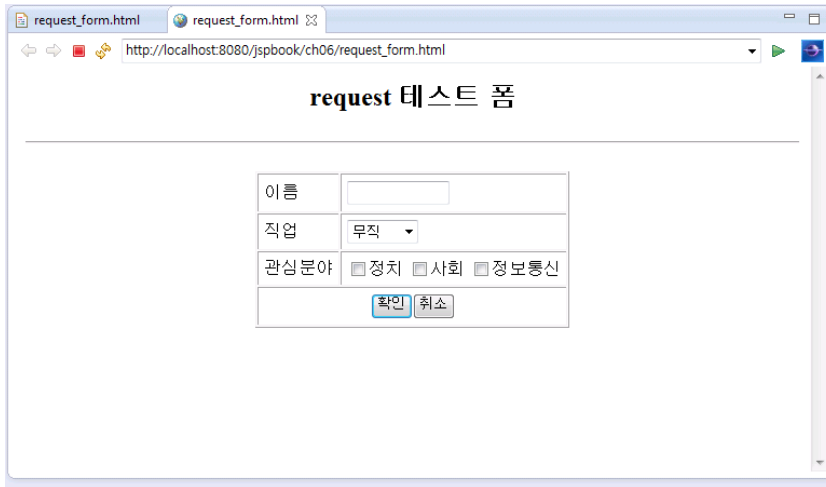
메서드	리턴타입	설명
getParameter(String name)	String	이름이 name인 파라미터의 값을 구한다. 존재하지 않을 경우 null
getParameterValues(String name)	String[]	이름이 name인 모드 파라미터의 값을 배열로 구한다. 존재하지 않을 경우 null
getParameterNames()	java.util.Enumeration	웹 브라우저가 전송한 파라미터의 이름을 구한다
getParameterMap()	java.util.Map	웹 브라우저가 전송한 파라미터의 맵 <파라미터 이름, 값> 세트로 구한다

getRemoteAddr()	String	웹 서버에 연결한 클라이언트의 IP 주소를 구한다. 게시판이나 방명록 등에서 글 작성자의 IP 주소가 자동으로 입력되기도 하는데, 이때 입력되는 IP 주소가 바로 이 메서드를 사용하여 구한 것이다.
getMethod()	String	웹 브라우저가 정보를 전송할 때 사용한 방식을 구한다.
getRequestURI()	String	웹 브라우저가 요청한 URL에서 경로를 구한다.
getContextPath()	String	JSP 페이지가 속한 웹 어플리케이션의 컨텍스트 경로를 구한다.
getServerName()	String	연결할 때 사용한 서버 이름을 구한다.
getServerPort()	int	서버가 실행중인 포트 번호를 구한다.



02. request

■ [실습] request 테스트 폼(request_form.html) - 교재 p.200 ~ 201 참고



[그림 6-1] 입력 화면

■ [실습] request 테스트 결과(request_result.jsp) - 교재 p.202 ~ 201 참고



이름	황희정
직업	무직
관심분야	정치 사회

request 테스트 결과 - 2

- 클라이언트 IP 주소 : 0:0:0:0:0:0:1
- 요청 메서드 : POST
- test 에 설정된 쿠키값 : OK.
- 프로토콜 : HTTP/1.1

[그림 6-2] 실행 결과

02. request

■ 주요 소스코드 분석(request_result.jsp)

- page 지시어 및 한글 캐릭터셋 변환

```
01 <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
02 <% request.setCharacterEncoding("UTF-8"); %>
```

- 한글 캐릭터셋 변환은 HTML 폼에서 POST 방식으로 전달된 한글 데이터의 처리를 위해 필요.

- 폼 입력값 가져오기

```
13 <td>이름</td>
14 <td><%=request.getParameter("username") %></td>
```

- getParameter() 메서드는 파라미터의 이름과 일치하는 HTML form 요소 입력 값을 가져옴.
- 입력 데이터 유형에 상관없이 String 형태로 값을 취함.

02. request

- 체크박스로 복수 선택된 값 가져오기

```
23 String favorites[] = request.getParameterValues("favorite");
24
25 // 배열의 크기만큼 루프를 돌면서 값을 출력함
26 for(String favorite : favorites){
27     out.println(favorite + "<BR>");
28 }
```

- 체크박스의 경우 동일 이름으로 여러 데이터가 선택되기 때문에 선택된 값들을 배열 형태로 가져와야 함.
- request.getParameterValues() 는 파라미터 이름과 일치하는 데이터의 문자열 배열을 가지고 옴.
- 가지고 온 배열 데이터는 for 문을 통해 출력함.

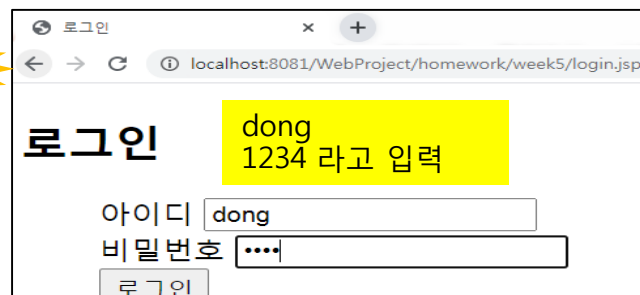
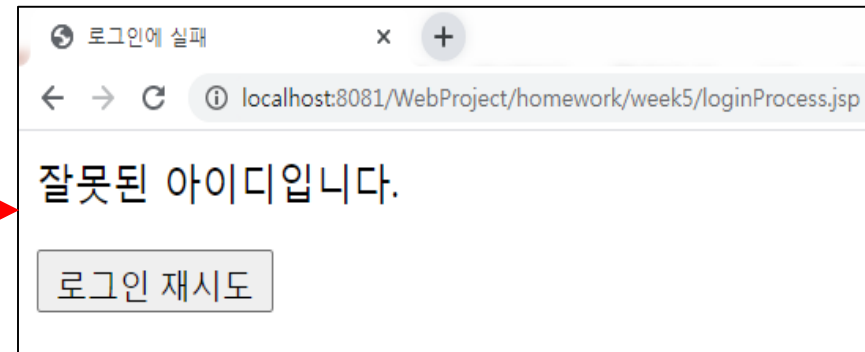
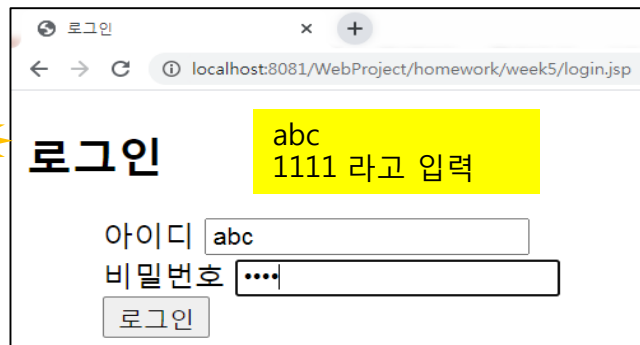
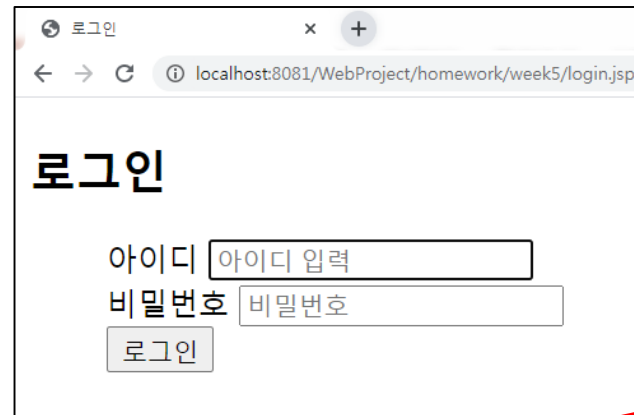
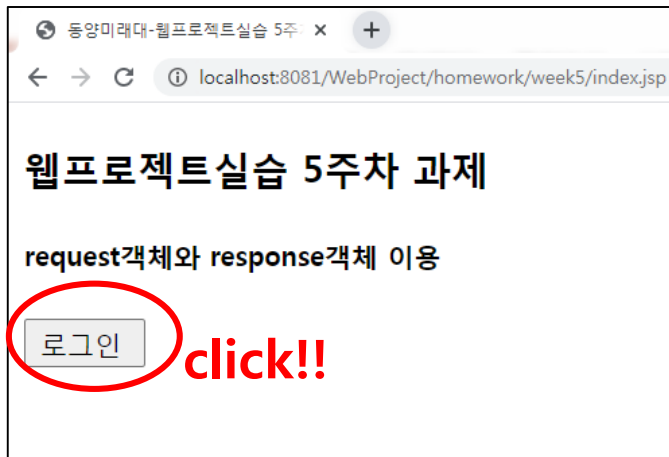
02. request

▪ 기타 request 메서드 사용 결과

```
34 1. 클라이언트 IP 주소 : <%= request.getRemoteAddr() %> <BR>
35 2. 요청 메서드 ; <%= request.getMethod() %> <BR>
36 <%
37     Cookie cookie[] = request.getCookies();
38 %>
39 3. <%= cookie[0].getName() %>에 설정된 쿠키 값 : <%=cookie[0].getValue() %> <BR>
40 4. 프로토콜 : <%= request.getProtocol() %>
```

- getRemoteAddr() 메서드는 접속 클라이언트 컴퓨터의 ip 주소를 가지고 옴.
- getMethod() 는 HTML 폼에서 데이터를 전달한 방법을 알려줌.
- Cookie 는 서버의 요청에 따라 웹 브라우저가 클라이언트에 저장하는 작은 데이터로 비 연결형 프로토콜인 HTTP의 문제점 해결을 위해 고안된 방법으로 getCookies(), cookie[].getName(), cookie[].getValue() 등의 메서드로 쿠키 정보에 접근이 가능함.
- 보안 및 사용성 문제로 인해 쿠키보다는 session 사용이 권장됨.

5주차 과제



다시 첫화면 index.jsp로 이동!!!!!!