

JSP 프로그래밍

(재)대덕인재개발원

06 기본 객체와 영역

- ▶ 1. 기본 객체
- ▶ 2. out 기본 객체
 - ▶ out 기본 객체의 출력 메서드
 - ▶ out 기본 객체와 버퍼의 관계
- ▶ 3. pageContext 기본 객체
 - ▶ 기본 객체 접근 메서드
- ▶ 4. application 기본 객체
 - ▶ 웹 어플리케이션 초기화 파라미터 읽어오기
 - ▶ 서버 정보 읽어오기
 - ▶ 로그 메시지 기록하기
 - ▶ 웹 어플리케이션의 자원 구하기
- ▶ 5. JSP 기본 객체와 영역
- ▶ 6. 기본 객체의 속성 사용하기
 - ▶ 속성의 값 타입
 - ▶ 속성의 활용 방법



6.1 기본 객체와 영역

▶ JSP가 제공하는 기본 객체

기본 객체	실제 타입	설 명
request	javax.servlet.http.HttpServletRequest 또는 javax.servlet.ServletRequest	클라이언트의 요청 정보를 저장한다.
response	javax.servlet.http.HttpServletResponse 또는 javax.servlet.ServletResponse	응답 정보를 저장한다.
pageContext	javax.servlet.jsp.PageContext	JSP 페이지에 대한 정보를 저장한다.
session	javax.servlet.http.HttpSession	HTTP 세션 정보를 저장한다.
application	javax.servlet.ServletContext	웹 어플리케이션에 대한 정보를 저장한다.
out	javax.servlet.jsp.JspWriter	JSP 페이지가 생성하는 결과를 출력할 때 사용되는 출력 스트림이다.
config	javax.servlet.ServletConfig	JSP 페이지에 대한 설정 정보를 저장한다.
page	java.lang.Object	JSP 페이지를 구현한 자바 클래스 인스턴스이다.
exception	java.lang.Throwable	예외 객체. 에러 페이지에서만 사용된다.

6.2 out 기본 객체

- ▶ out 기본 객체는 웹 브라우저에 데이터를 전송하는 출력 스트림으로서 JSP 페이지가 생성한 데이터를 출력한다.

```
<%@ page contentType = "text/html; charset=euc-kr" %>
<html>
<head><title>out 기본 객체 사용</title></head>
<body>

<%
    out.println("안녕하세요?");
%>
<br>
out 기본 객체를 사용하여
<%
    out.println("출력한 결과입니다.");
%>

</body>
</html>
```

```
out.println("<html>");
out.println("<head>");
out.println("<title>");
...
out.println(someValue);
...
```

6.2 out 기본 객체

▶ out 기본 객체의 버퍼 관련 메서드

메서드	리턴 타입	설 명
getBufferSize()	int	버퍼의 크기를 구한다.
getRemaining()	int	현재 버퍼의 남은 크기를 구한다.
clear()	void	버퍼의 내용을 비운다. 만약 버퍼가 이미 플러시 되었다면 IOException을 발생시킨다.
clearBuffer()	void	버퍼의 내용을 비운다. clear() 메서드와 달리 버퍼를 플러시 한 경우에도 IOException을 발생시키지 않는다.
flush()	void	버퍼를 플러시 한다.
isAutoFlush()	boolean	버퍼가 다 찼을 때 자동으로 플러시 할 경우 true를 리턴한다.



6.3 pageContext 기본 객체

- ▶ pageContext 기본 객체는 하나의 JSP 페이지와 1:1 매핑되는 객체.
- ▶ pageContext 기본 객체의 기능
 - ▶ 다른 기본 객체 구하기 (7가지(request, response외) 내장 객체중 가장 먼저 생성되는 객체)
 - ▶ 속성 처리하기 (pageScope에 해당하는 영역에 속성으로 데이터를 전달)
 `${pageScope.속성명 }`의 형태로 속성 데이터를 표현.
 - ▶ 페이지의 흐름 제어하기(forward, include 메소드를 이용.)
 - ▶ 에러 데이터 구하기(page 디렉티브에서 isErrorPage 속성을 true로 준 경우 사용.)



6.3 pageContext 기본 객체

- ▶ pageContext의 기본 객체 접근 메서드 (컨테이너에 의해 jsp페이지가 호출될때 가장 먼저 생성되는 객체로, 이녀석을 통해 나머지 내장객체들을 생성한다.)

메서드	리턴 타입	설 명
getRequest()	ServletRequest	request 기본 객체를 구한다.
getResponse()	ServletResponse	response 기본 객체를 구한다.
getSession()	HttpSession	session 기본 객체를 구한다.
getServletContext()	ServletContext	application 기본 객체를 구한다.
getServletConfig()	ServletConfig	config 기본 객체를 구한다.
getOut()	JspWriter	out 기본 객체를 구한다.
getException()	Exception	exception 기본 객체를 구한다.
getPage()	Object	page 기본 객체를 구한다.

6.4 application 기본 객체

- ▶ application 기본 객체는 웹 어플리케이션과 관련된 기본 객체이다.
- ▶ 특정 웹 어플리케이션에 포함된 모든 JSP 페이지는 하나의 application 기본 객체를 공유하게 된다.
- ▶ application 기본 객체의 웹 어플리케이션 초기화 파라미터 메서드

메서드	리턴 타입	설 명
getInitParameter(String name)	String	이름이 name인 웹 어플리케이션 초기화 파라미터의 값을 읽어온다. 존재하지 않을 경우 null을 리턴한다.
getInitParameterNames()	Enumeration	



6.4 application 기본 객체

- ▶ 웹 어플리케이션 초기화 패러미터 읽어오기

```
<web-app xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
  version="2.5">

  <context-param>
    <description>오라클 드라이버</description>
    <param-name>driver</param-name>
    <param-value>oracle.jdbc.driver.OracleDriver</param-value>
  </context-param>

  <context-param>
    <description>디버깅 레벨</description>
    <param-name>debugLevel</param-name>
    <param-value>5</param-value>
  </context-param>

</web-app>
```

6.4 application 기본 객체

▶ 서버 정보 읽어오기

메서드	리턴 타입	설 명
getServerInfo()	String	서버 정보를 구한다.
getMajorVersion()	String	서버가 지원하는 서블릿 규약의 메이저 버전을 리턴한다. 버전의 정수 부분을 리턴한다.
getMinorVersion()	String	서버가 지원하는 서블릿 규약의 마이너 버전을 리턴한다. 버전의 소수 부분을 리턴한다.



6.4 application 기본 객체

▶ 로그 메시지 기록하기

- ▶ application 기본 객체는 웹 컨테이너가 사용하는 로그 파일에 로그 메시지를 기록할 수 있도록 메서드를 제공한다.
- ▶ application.log(“로그 메시지 기록”) : application 이 제공하는 log() 메서드
- ▶ log(“로그 메시지 기록”) : JSP 페이지가 제공하는 log() 메서드

메서드	리턴 타입	설 명
log(String msg)	void	로그 메시지 msg를 기록한다.
log(String msg, Throwable throwable)	void	로그 메시지 msg를 기록한다. 예외 정보도 함께 로그 파일에 기록한다.



6.4 application 기본 객체

▶ 웹 어플리케이션의 자원 구하기

메서드	리턴 타입	설 명
<code>getRealPath(String path)</code>	<code>String</code>	웹 어플리케이션 내에서 지정한 경로에 해당하는 자원의 시스템상에서의 자원 경로를 리턴한다.
<code>getResource(String path)</code>	<code>java.net.URL</code>	웹 어플리케이션 내에서 지정한 경로에 해당하는 자원에 접근할 수 있는 URL 객체를 리턴한다.
<code>getResourceAsStream(String path)</code>	<code>java.io.InputStream</code>	웹 어플리케이션 내에서 지정한 경로에 해당하는 자원으로 부터 데이터를 읽어 올 수 있는 InputStream 을 리턴한다.

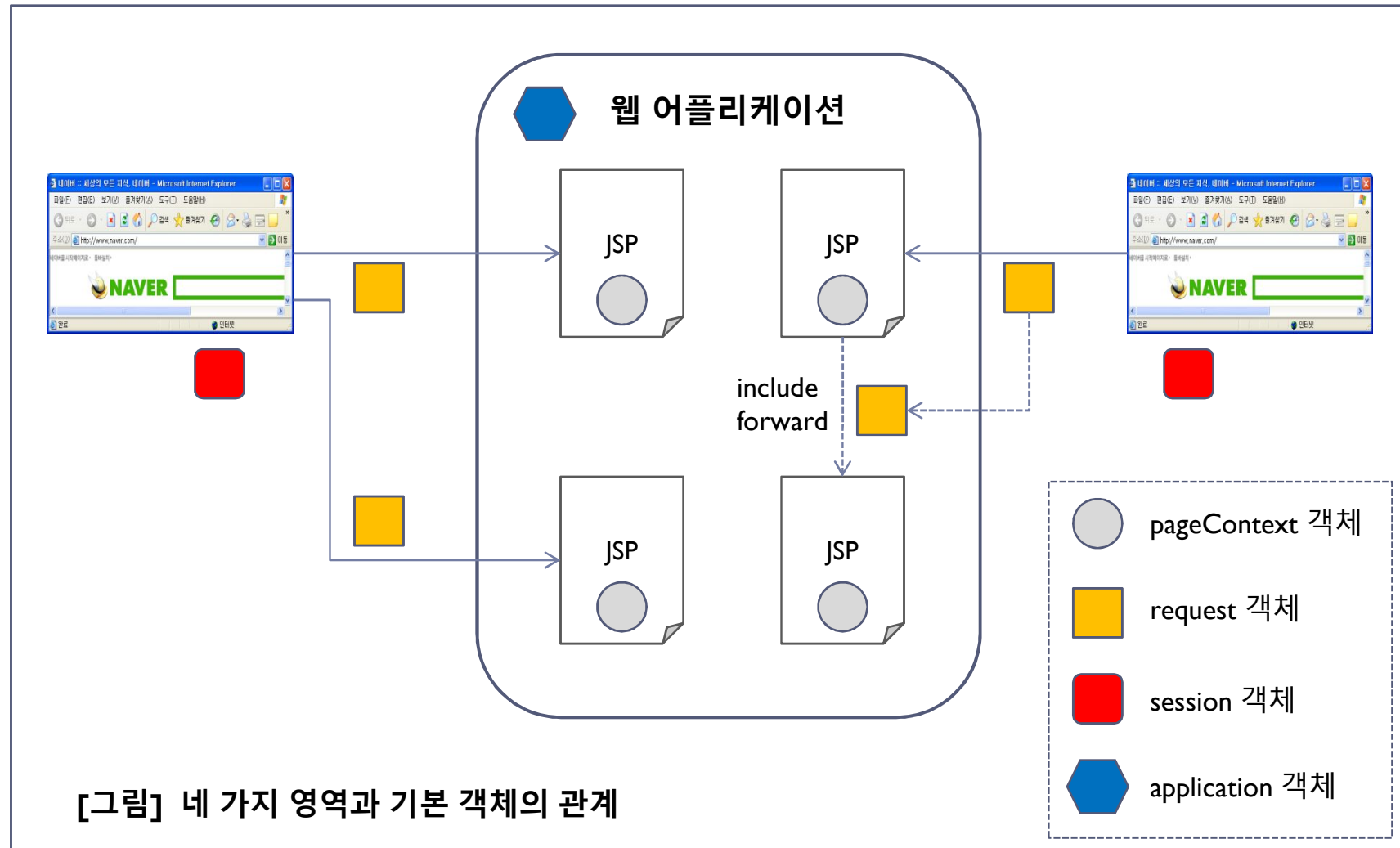


6.5 JSP 기본 객체와 영역

- ▶ 웹 어플리케이션의 영역(Scope)
- ▶ 각영역의 속성을 EL로 사용하는 방법 - `${영역명Scope.속성명}`

영역(scope)	설 명
PAGE 영역	<ul style="list-style-type: none"> - 하나의 JSP 페이지를 처리할 때 사용되는 영역. - 한번의 클라이언트 요청에 대해서 하나의 JSP 페이지를 범위로 갖는다. - <code>pageContext</code> 기본 객체.
REQUEST 영역	<ul style="list-style-type: none"> - 하나의 HTTP 요청을 처리할 때 사용되는 영역. - 웹 브라우저가 웹 서버에 전송하는 요청이 하나의 REQUEST 영역이 된다. - 웹 브라우저가 요청을 보낼 때마다 매번 새로운 REQUEST 영역이 생성된다. - PAGE영역은 오직 하나의 페이지만을 포함하는데 반해, REQUEST 영역은 하나의 요청을 처리하는데 사용되는 모든 JSP 페이지를 포함한다. - <code>request</code> 기본 객체.
SESSION 영역	<ul style="list-style-type: none"> - 하나의 웹 브라우저와 관련된 영역. - 세션이 생성되면 하나의 웹 브라우저와 관련된 모든 요청은 하나의 SESSION 영역에 포함된다. - <code>session</code> 기본 객체.
APPLICATION 영역	<ul style="list-style-type: none"> - 하나의 웹 어플리케이션과 관련된 전체 영역. - 웹 어플리케이션에 포함된 모든 JSP페이지, 요청, 세션은 모두 하나의 APPLICATION 영역에 속한다. - <code>application</code> 기본 객체.

6.5.1 JSP 기본 객체와 영역



6.6 기본 객체의 속성(attribute) 사용하기

- ▶ 네 개의 기본 객체 `pageContext`, `request`, `session`, `application`은 속성을 갖는다.
- ▶ 속성은 각각의 기본 객체가 존재하는 동안에 사용될 수 있으며, JSP 페이지 사이에서 정보를 주고 받거나 공유하기 위한 목적으로 사용된다.
- ▶ 속성은 <속성이름, 값>의 형태를 가진다.

메서드	리턴 타입	설 명
<code>setAttribute(String name, Object value)</code>	<code>void</code>	이름이 <code>name</code> 인 속성의 값을 <code>value</code> 로 지정한다.
<code>getAttribute(String name)</code>	<code>Object</code>	이름이 <code>name</code> 인 속성의 값을 구한다. 지정한 이름의 속성이 존재하지 않을 경우 <code>null</code> 을 리턴한다.
<code>removeAttribute(String name)</code>	<code>void</code>	이름이 <code>name</code> 인 속성을 삭제한다.
<code>getAttributeNames()</code>	<code>Enumeration</code>	속성의 이름 목록을 구한다. (<code>pageContext</code> 기본 객체는 이 메소드를 제공하지 않는다.)

6.6 기본 객체의 속성(attribute) 사용하기

▶ 속성의 활용 방법

기본 객체	영역	쓰임새
pageContext	PAGE	(한번의 요청을 처리하는) 하나의 JSP 페이지 내에서 공유될 값을 저장한다. 주로 커스텀 태그에서 새로운 변수를 추가할 때 사용된다.
request	REQUEST	한번의 요청을 처리하는 데 사용되는 모든 JSP 페이지에서 공유될 값을 저장한다. 주로 하나의 요청을 처리하는 데 사용되는 JSP 페이지 사이에서 정보를 전달하기 위해 사용된다.
session	SESSION	한 사용자와 관련된 정보를 JSP 들이 공유하기 위해서 사용된다. 사용자의 로그인 정보와 같은 것 들을 저장한다.
application	APPLICATION	모든 사용자와 관련해서 공유할 정보를 저장한다. 임시 디렉토리 경로와 같은 웹 어플리케이션의 설정 정보를 주로 저장한다.