

JSP

– JSP 내장 객체

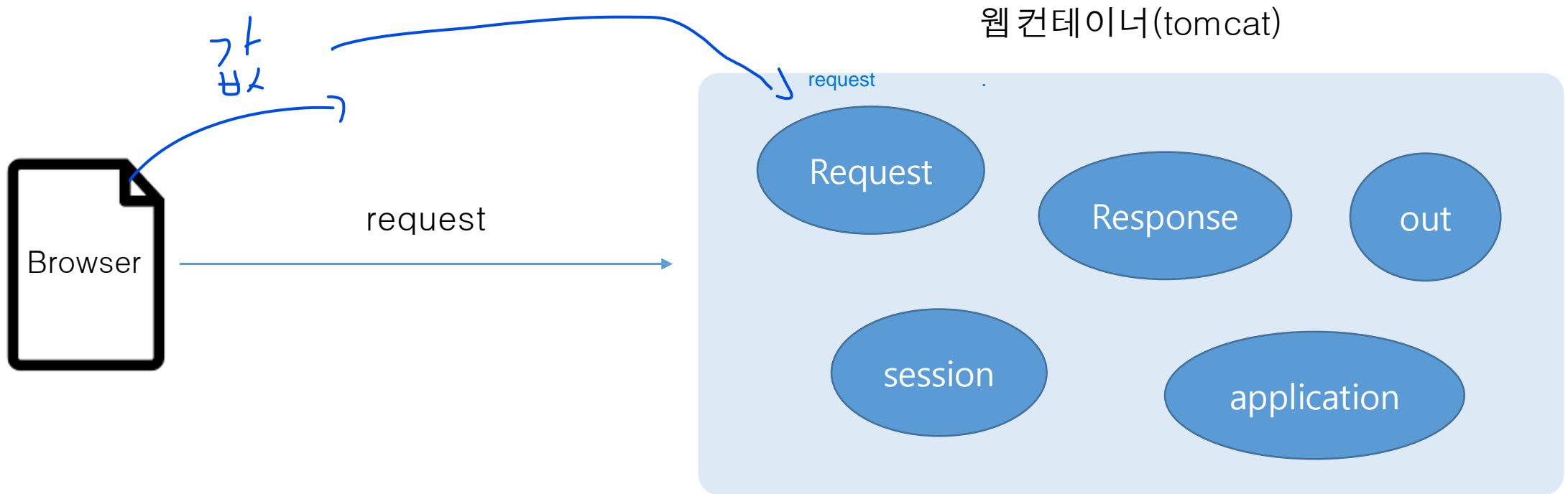
- 1.Request
- 2.Response
- 3.Out

## \* JSP 내장 객체 종류

종류	해당 클래스
1. request	javax.servlet.http.HttpServletRequest
2. response	javax.servlet.http.HttpServletResponse
3. out	javax.servlet.jsp.JspWriter
4. session	javax.servlet.http.HttpSession
5. application	javax.servlet.ServletContext
6. pageContext	javax.servlet.jsp.PageContext
7. page	javax.servlet.jsp.HttpJspPage
8. config	javax.servlet.ServletConfig
9. exception	java.lang.Throwable

## \* JSP 내장 객체

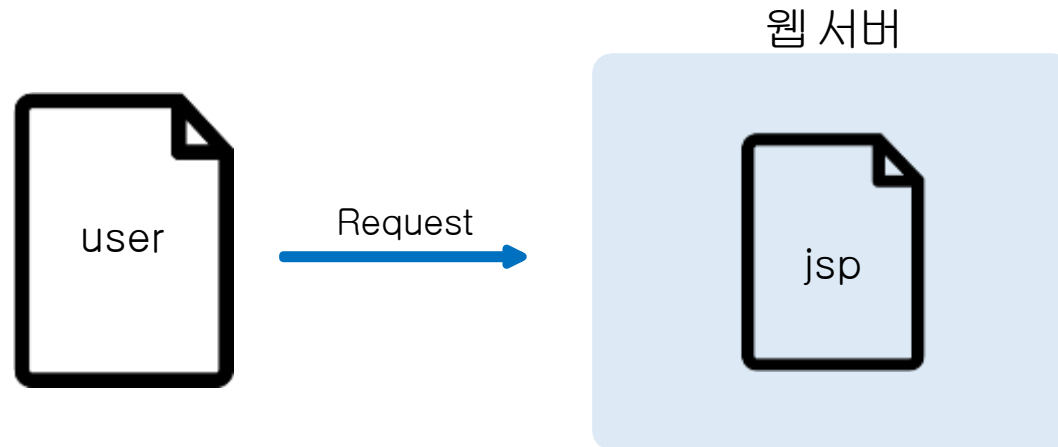
- 개발자가 JSP 파일 내에 객체를 생성하지 않고 바로 사용할 수 있는 객체가 내장 객체입니다.
- JSP에서 제공되는 내장객체는 JSP 컨테이너에 의해 서블릿으로 변환될 때 자동으로 객체가 생성됩니다.



자동 객체 생성!

## \* request 객체의 이해

- 웹 브라우저를 통해 서버에 어떤 정보를 요청하는 것을 request라고 합니다.
- 이러한 요청 정보가 담기고 관리되는 곳이 request객체입니다.
- request 객체가 제공하는 기능.
  1. 클라이언트(웹 브라우저)와 관련된 정보 읽기 기능.
  2. 서버와 관련된 정보 읽기 기능.
  3. 클라이언트가 전송한 요청 파라미터 읽기 기능.
  4. 클라이언트가 전송한 쿠키 읽기 기능.

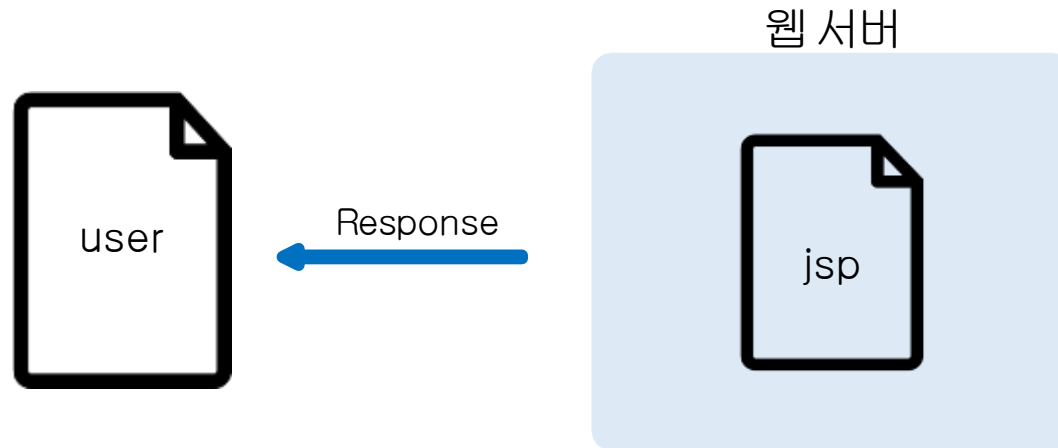


## - request 객체 관련 주요 메서드

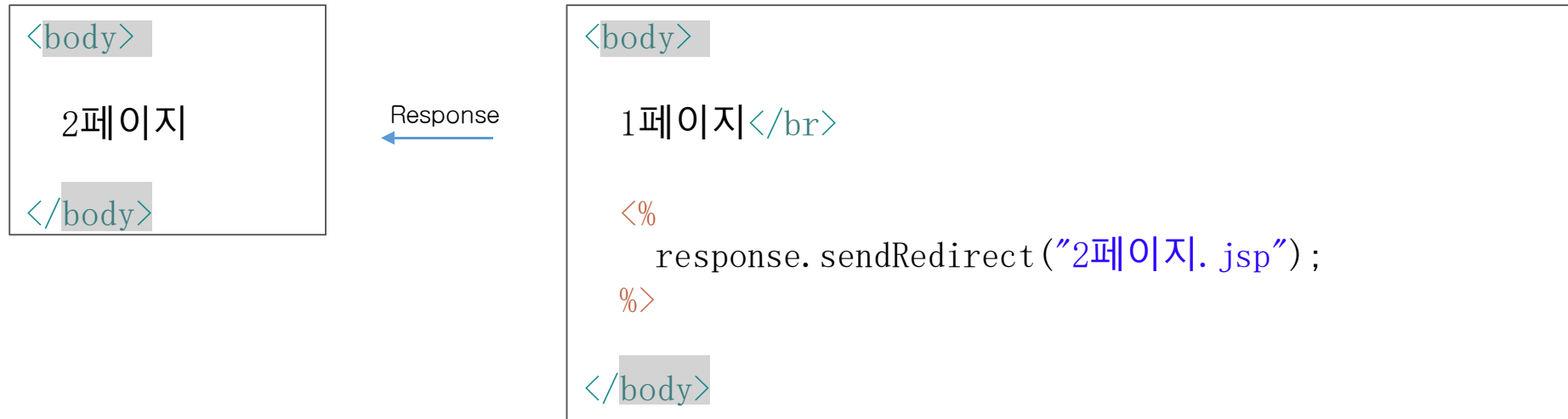
메서드	기능
1. <code>getContextPath()</code>	String - 웹 어플리케이션의 컨텍스트 루트의 경로를 얻습니다.
2. <code>getMethod():</code>	String - 웹 브라우저가 정보를 전송할 때 사용한 요청 방식을 구합니다.(get, post)
3. <code>getServerName()</code>	String - 연결할 때 사용한 서버 이름을 구합니다.
4. <code>getServerPort()</code>	int - 서버가 실행중인 포트 번호를 구합니다.
5. <code>getRequestURL()</code>	String - 요청 URL을 얻습니다.
6. <code>getRequestURI()</code>	String - 요청 URI를 얻습니다.
7. <code>getRemoteAddr()</code>	String - 웹 서버에 연결한 클라이언트의 IP주소를 구합니다.
8. <code>getProtocol()</code>	String - 해당 프로토콜을 얻습니다.
9. <u><code>getParameter(name)</code></u>	이름이 name인 파라미터 값을 구합니다. 존재하지 않을 경우 null을 반환합니다.
10. <u><code>getParameterValues(name)</code></u>	이름이 name인 모든 파라미터 값들을 배열로 구합니다. 존재하지 않을 경우 null을 반환합니다.

## \* response 객체의 이해

- 웹 브라우저의 요청에 응답하는 것을 response라고 합니다.
- 이러한 응답의 정보를 가지고 있는 객체를 response객체라고 합니다.



## \* response 객체의 이해



메서드	기능
1. <code>getCharacterEncoding()</code>	응답할 때의 문자의 인코딩 형태를 구합니다.
2. <code>addCookie(Cookie c)</code>	쿠키를 지정합니다.
3. <code>sendRedirect(URL)</code>	지정한 URL로 이동합니다.

## \* out 객체의 이해

- JSP 페이지가 생성하는 모든 내용은 out 기본 객체를 통해 전송됩니다.
- JSP 페이지 내에서 사용하는 비스크립트 요소들(HTML코드와 텍스트)이 out 객체에 전달됩니다.
- 값을 출력하는 표현식(expression)의 결과값도 out객체에 전달됩니다.
- 표현식으로 대체합니다

```
<%  
out.print("<h1>안녕!<h1> ");  
out.print(" <h2>반가워</h2>");  
%>
```



