

Chapter 14

표현 언어 (EL : Expression Language)

표현 언어의 개요

표현 언어의 개요

■ 표현 언어 (EL : Expression Language)

- JSP 2.0부터 도입된 데이터 출력 기능

- ▶ JSP 문서 내의 스크립트릿 사용을 최소화하고 표현식을 확장하여 편리하게 사용하도록 설계된 언어
- ▶ 현재 JSP 페이지는 스크립트릿 보다는 EL과 JSTL을 사용해 표현하고 있음

- EL의 형식과 구조

`${표현식 or 값}`

- ▶ 표현 언어는 '\$'로 시작함
- ▶ 표현식은 기본적으로 변수 이름, '객체.멤버변수'의 구조로 이루어짐
 - 부가적으로 숫자, 문자열, Boolean, null 과 같은 값도 올 수 있음
- ▶ 표현식은 기본적인 연산이 포함될 수 있음

표현 언어의 개요

■ EL의 특징

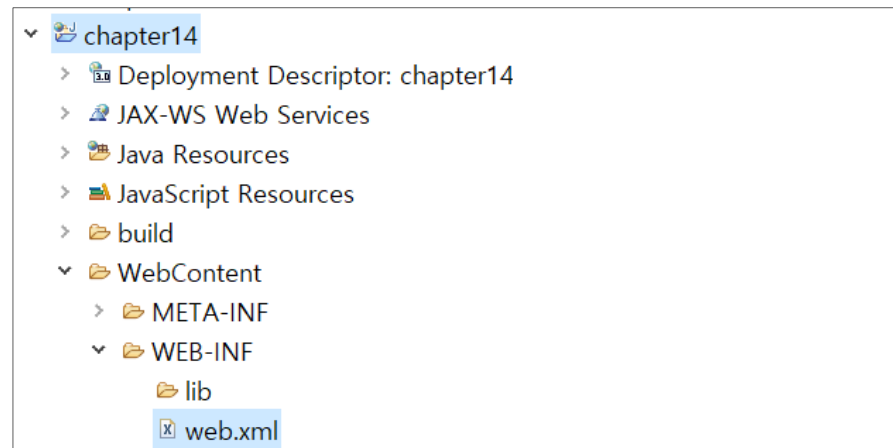
- 기존 표현식보다 편리하게 값을 출력
 - ▶ 변수와 여러 가지 연산자를 포함할 수 있음
- JSP의 내장 객체에 바인딩된 속성을 쉽게 출력할 수 있으며, 빈즈의 속성도 출력할 수 있음
 - ▶ 표현식 내에서 사용할 수 있는 별도의 내장 객체를 제공
 - ▶ 추출하기 위한 메서드를 사용하지 않고 추출해 출력 가능
- page 지시어에서 반드시 `isELIgnored = false`로 설정해야 함
 - ▶ `isELIgnored` 속성은 EL을 지원하지 않는 JSP 버전에서 EL 구문을 문자열로 간주하기 위해 사용됨
 - `isELIgnored` 값을 `true`로 지정하면 EL 구문을 문자열로 표시 (EL을 사용하지 않도록 함)
 - ▶ JSP 2.0 이하 버전에서는 EL을 사용하지 않음
 - ▶ 기본 값은 `false`이므로 생략해도 EL을 사용할 수 있음

EL의 자료형과 연산자

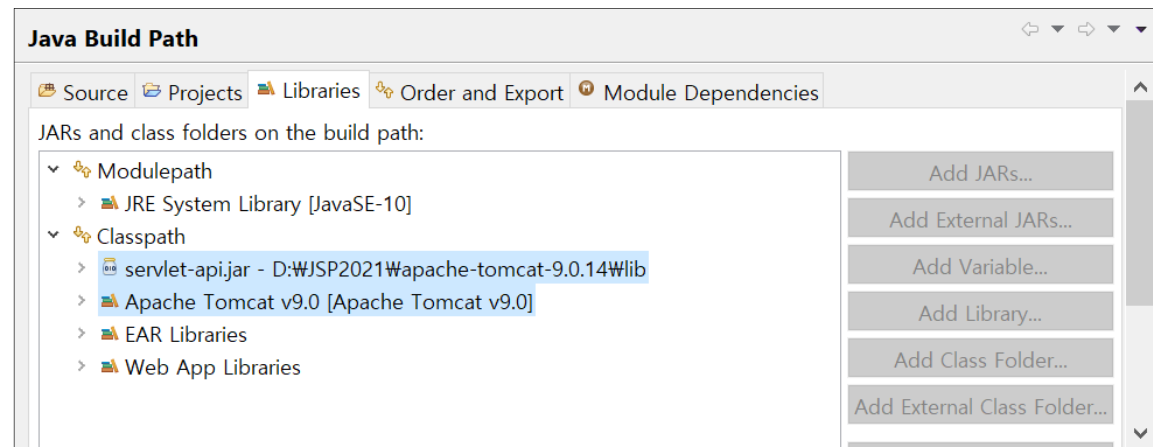
표현 언어의 자료형

■ 프로젝트 생성

- 프로젝트 이름 : chapter14
 - ▶ web.xml 파일은 반드시 생성해야 함



- servlet-api 라이브러리 추가

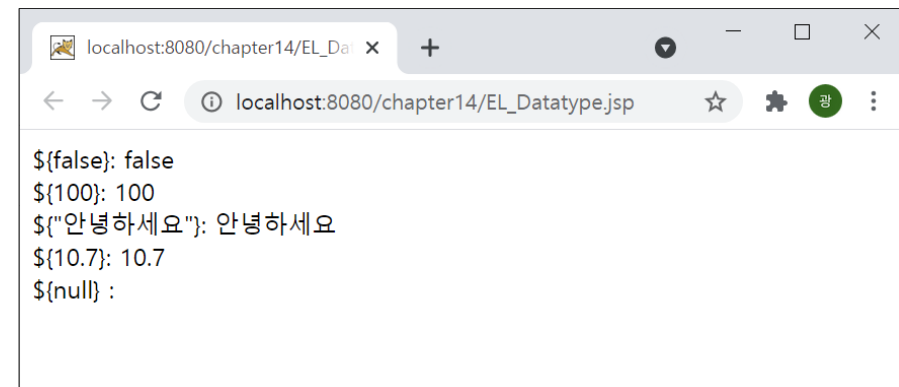


표현 언어의 자료형

■ EL의 자료형

자료형	설명
불	true와 false 값을 가질 수 있음
정수	0~9로 구성. 음수인 경우 마이너스(-)가 사용됨
실수	소수점을 가지는 수. (지수형으로도 표현 가능)
문자열	인용 부호 내에 표현되는 문자 또는 문자열
널	null을 가짐

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5     <head>
6         <meta charset="UTF-8">
7     </head>
8     <body>
9
10         \${false}: \${false}<br>
11         \${100}: \${100}<br>
12         \${"안녕하세요"}: \${"안녕하세요"}<br>
13         \${10.7}: \${10.7}<br>
14         \${null} : \${null}<br>
15
16     </body>
17 </html>
18
```



- 정상적으로 실행은 되지만 이클립스에서 오류 문자가 출력될 경우

- ▶ 프로젝트 단축 메뉴 → Build Path → Configure Build Path →

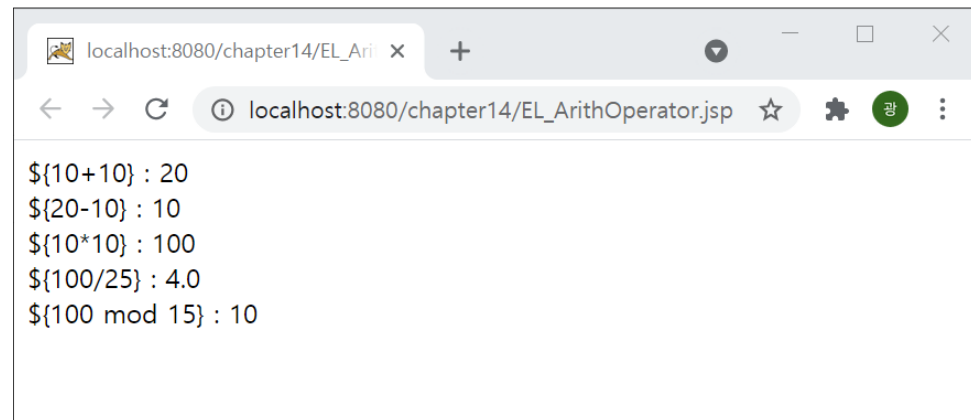
- Libraries/ ClassPath → Add Library → Server Runtimes → Apache Tomcat 9.0

표현 언어의 연산자

■ EL의 산술 연산자

연산자	설명
+	덧셈 연산 수행
-	뺄셈 연산 수행
*	곱셈 연산 수행
/ 또는 div	나눗셈 연산 수행
% 또는 mod	나머지 연산 수행

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6     <meta charset="UTF-8">
7 </head>
8 <body>
9     \${10+10} : \${10+10}<br>
10
11     \${20-10} : \${20-10}<br>
12
13     \${10*10} : \${10*10}<br>
14
15     \${100/25} : \${100/25}<br>
16
17     \${100 mod 15} : \${100 mod 15}<br>
18
19 </body>
20 </html>
```



표현 언어의 연산자

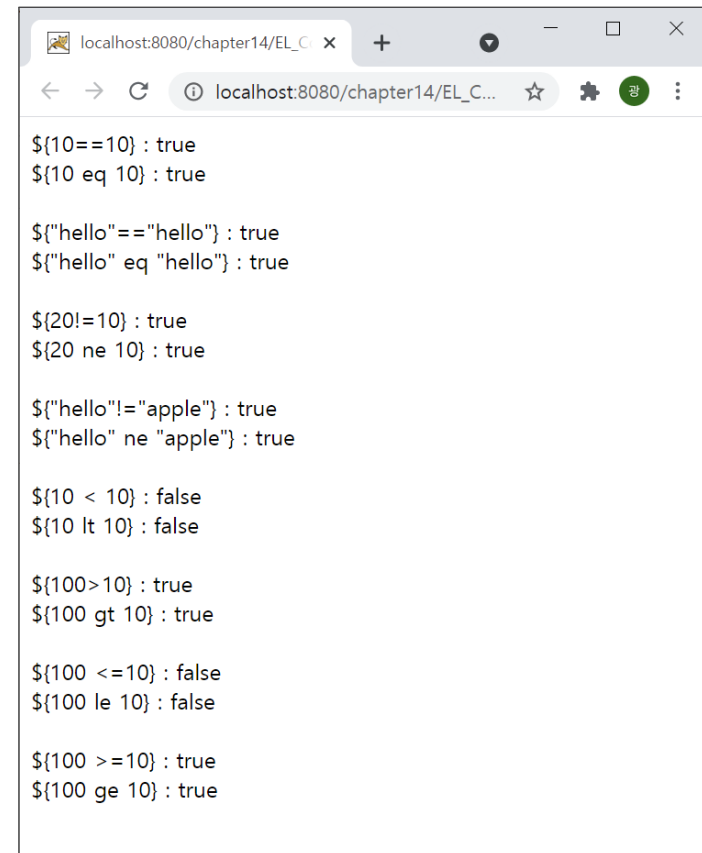
■ EL의 비교 연산자

연산자	설명
<code>=</code> 또는 <code>eq</code>	두 값이 같은지 비교
<code>!=</code> 또는 <code>ne</code>	값이 다른지 비교
<code><</code> 또는 <code>lt</code>	값이 다른 값보다 작은지 비교
<code>></code> 또는 <code>gt</code>	값이 다른 값보다 큰지 비교
<code><=</code> 또는 <code>le</code>	값이 다른 값보다 작거나 같은지 비교
<code>>=</code> 또는 <code>ge</code>	값이 다른 값보다 크거나 같은지 비교

```

1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3
4 \${10==10} : \${10==10} <br>
5 \${10 eq 10} : \${10 eq 10} <br><br>
6
7 \${"hello"=="hello"} : \${"hello"=="hello"} <br>
8 \${"hello" eq "hello"} : \${"hello" eq "hello"} <br><br>
9
10 \${20!=10} : \${20!=10}<br>
11 \${20 ne 10} : \${20 ne 10}<br><br>
12
13 \${"hello"!="apple"} : \${"hello"!="apple"} <br>
14 \${"hello" ne "apple"} : \${"hello" ne "apple"} <br><br>
15
16 \${10 < 10} : \${10 < 10} <br>
17 \${10 lt 10} : \${10 lt 10} <br><br>
18
19 \${100>10} : \${100 > 10}<br>
20 \${100 gt 10} : \${100 gt 10}<br><br>
21
22 \${100 <=10} : \${100 <= 10}<br>
23 \${100 le 10} : \${100 le 10}<br><br>
24
25 \${100 >=10} : \${100 >= 10}<br>
26 \${100 ge 10} : \${100 ge 10}<br><br>

```



표현 언어의 연산자

■ EL의 논리 연산자

연산자	설명
&& 또는 and	논리곱 연산
또는 or	논리합 연산
! 또는 not	부정 연산

■ empty 연산자

연산자	설명
empty <값>	<값>이 null 이거나 빈 문자열이면 true를 반환

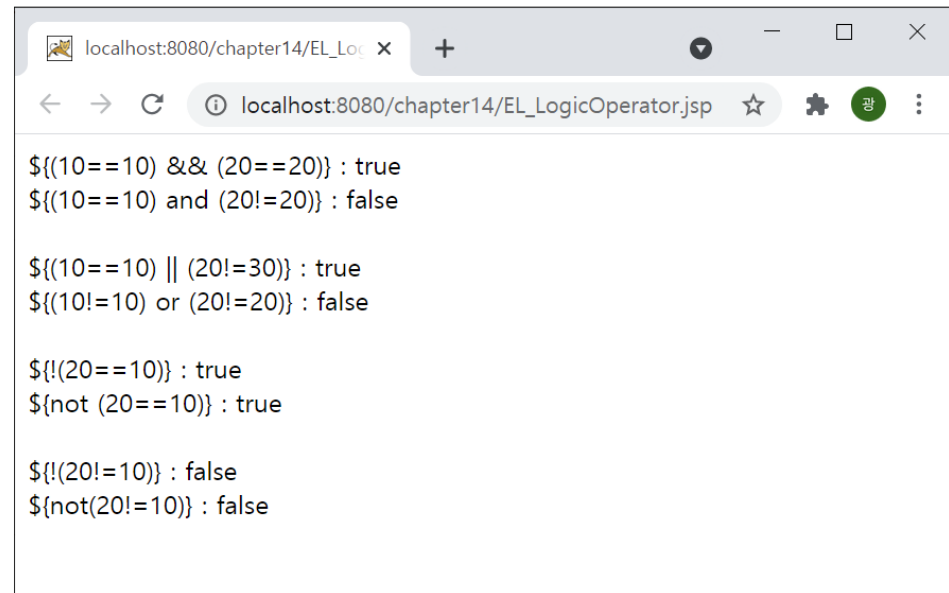
■ 문자열 연산자

연산자	설명
"문자열1" += "문자열2"	문자열1과 문자열2를 결합

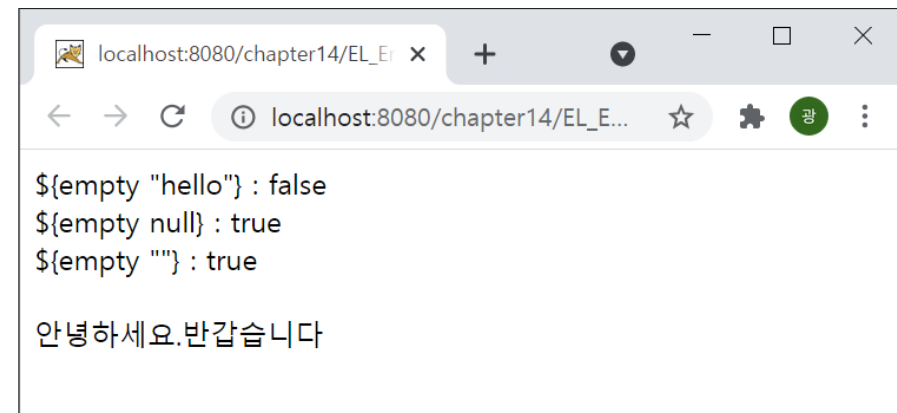
```

1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3
4 \${(10==10) && (20==20)} : ${ (10==10)&&(20==20)} <br>
5 \${(10==10) and (20!=20)} : ${ (10==10) and (20!=20)} <br><br>
6
7 \${(10==10) || (20!=30)} : ${ (10==10)|| (20==30)} <br>
8 \${(10!=10) or (20!=20)} : ${ (10!=10) or (20!=20)} <br><br>
9
10 \${!(20==10)} : ${!(20==10)}<br>
11 \${not (20==10)} : ${not (20==10)}<br><br>
12
13 \${!(20!=10)} : ${!(20!=10)}<br>
14 \${not(20!=10)} : ${not(20!=10)}<br><br>

```



```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6   <meta charset="UTF-8">
7 </head>
8 <body>
9
10   \${empty "hello"} : \${empty "hello" }<br>
11   \${empty null} : \${empty null } <br>
12   \${empty ""} : \${empty "" } <br><br>
13
14   ${"안녕하세요." += "반갑습니다"}<br>
15
16 </body>
17 </html>
```



EL의 내장 객체

EL 내장 객체의 개요

■ EL에서의 내장 객체

- EL은 기본적으로 내장 객체를 제공하고 있음
- JSP에서 제공하는 기본 객체와 EL에서 사용하는 내장객체에는 차이가 있음
 - JSP의 기본 객체
 - 스크립팅 요소 안에서 사용할 수 있는 내장 객체
 - EL에서의 내장 객체
 - 스크립팅 요소 안에서 사용할 수 없으며, 반드시 EL의 표현식 내에서만 사용할 수 있음
- EL에서 사용하는 내장 객체의 분류
 - 객체의 스코프(영역) 관련 내장 객체
 - 클라이언트로부터 전달되는 매개변수 관련 내장 객체
 - 헤더 관련 내장 객체
 - 쿠키 관련 내장 객체
 - 초기화 변수 관련 내장 객체

EL 내장 객체의 개요

■ EL 내장 객체의 종류와 기능

구분	내장 객체	설명
스cope	pageScope	JSP의 page와 같은 기능(page 영역에 바딩인된 객체를 참조)
	requestScope	JSP의 request와 같은 기능(request에 바인딩된 객체를 참조)
	sessionScope	JSP의 session과 같은 기능(session에 바인딩된 객체를 참조)
	applicationScope	JSP의 application과 같은 기능(application에 바인딩된 객체 참조)
요청 매개변수	param	request.getParameter() 메서드를 호출한 것과 같음 (하나의 매개변수 처리)
	paramValues	request.getParameterValues() 메서드를 호출한 것과 같음 (여러 개의 매개변수 처리)
JSP 문서	pageContext	JSP의 pageContext를 참조할 때 사용 가능
쿠키 값	Cookies	쿠키 이름의 값을 반환

스코프 관련 내장 객체

■ 스코프 관련 내장 객체

- page, request, session, application 영역에 바인딩된 속성을 참조하기 위한 객체
 - ▶ 속성을 참조할 때 생략이 가능하지만 영역을 정확하게 지정하기 위해 사용됨
- 속성의 이름이 같지만 바인딩된 영역이 다른 경우, 다음의 순서로 참조하므로 주의해야 함

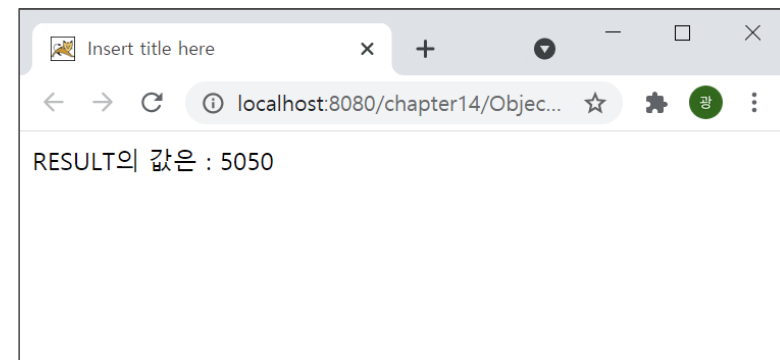
page → request → session → application

- ▶ 예를 들어, RESULT라는 속성이 request 영역과 session 영역에 동시에 바인딩 되어 있는 경우
 - request 영역에 바인딩되어 있는 속성을 참조함
- 특정 영역에 바인딩되어 있는 속성을 참조하려면 객체와 함께 속성을 지정해야 함

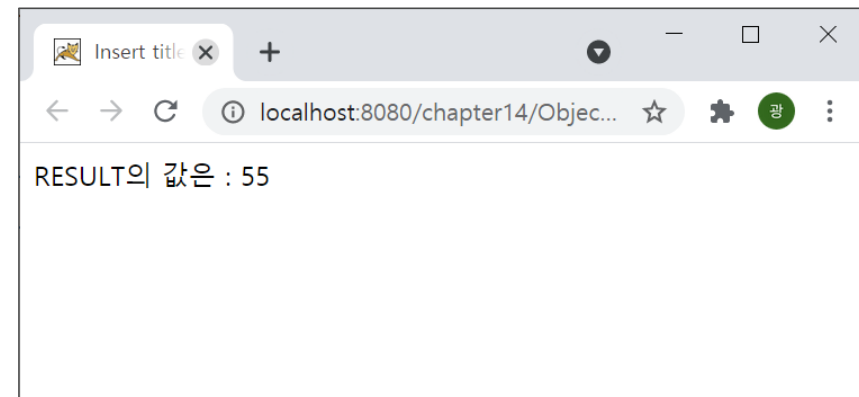
```
${ pageScope.속성명 }  
${ requestScope.속성명 }  
${ sessionScope.속성명 }  
${ applicationScope.속성명 }
```

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <%
4   int sum = 0;
5
6   for (int cnt = 1; cnt <= 10; cnt++)
7     sum += cnt;
8
9   request.setAttribute("RESULT", sum );
10
11   RequestDispatcher dispatcher = request.getRequestDispatcher("Object_ScopeTestProc.jsp");
12   dispatcher.forward(request, response);
13
14 %>
```

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6 <meta charset="UTF-8">
7 <title>Insert title here</title>
8 </head>
9 <body>
10
11 <%
12     int sum = 0;
13
14     for (int cnt = 1; cnt <= 100; cnt++)
15         sum += cnt;
16
17     pageContext.setAttribute("RESULT", sum);
18 %>
19
20     RESULT의 값은 : ${RESULT}
21
22 </body>
23 </html>
```



```
18 %>
19
20 RESULT의 값은 : ${requestScope.RESULT}
21
22 </body>
23 </html>
```



요청 매개변수 관련 내장 객체

■ 요청 매개변수 관련 내장 객체

- param 내장 객체

- ▶ request 내장 객체의 `getParameter()` 메서드를 사용하지 않고 매개변수를 추출하여 출력하기 위해 사용
- ▶ 하나의 매개변수를 추출하여 출력하기 위해 사용

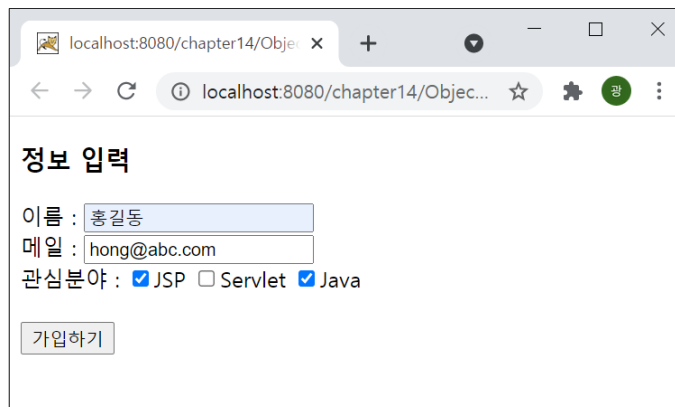
- paramValues 내장 객체

- ▶ request 내장 객체의 `getParameterValues()` 메서드를 사용하지 않고 매개변수를 추출하여 출력하기 위해 사용
- ▶ checkbox와 같이 하나 이상의 매개변수를 추출하여 출력하기 위해 사용

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6   <meta charset="UTF-8">
7 </head>
8 <body>
9
10   <form name="frm1" method="post" action="Object_ParamTestProc.jsp">
11
12     <h3>정보 입력</h3>
13
14     이름 : <input type="text" name="name" size=20><br>
15     메일 : <input type="text" name="email" size=20><br>
16     관심분야 :
17       <input type="checkbox" name="inter" value="JSP">JSP
18       <input type="checkbox" name="inter" value="Servlet">Servlet
19       <input type="checkbox" name="inter" value="Java">Java
20     <br><br>
21     <input type="submit" value="가입하기">
22
23   </form>
24
25 </body>
26 </html>
```



```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3 <% request.setCharacterEncoding("utf-8"); %>
4
5 <!DOCTYPE html>
6 <html>
7 <head>
8     <meta charset="UTF-8">
9 </head>
10 <body>
11
12     이름 : ${param.name }<br>
13     이메일 : ${param.email } <br>
14     관심분야 :
15         ${paramValues.inter[0]}
16         ${paramValues.inter[1]}
17         ${paramValues.inter[2]}
18
19 </body>
20 </html>
```



localhost:8080/chapter14/Objec... x + - □ ×

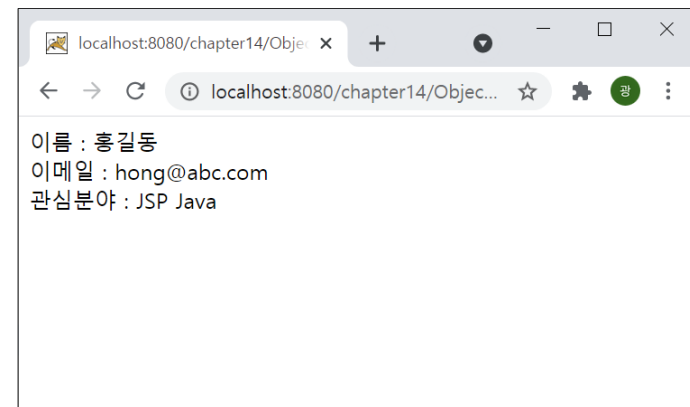
← → ↻ ⓘ localhost:8080/chapter14/Objec... ☆ 🛠️ 📄 ⋮

정보 입력

이름 :

이메일 :

관심분야 : ☒ JSP ☐ Servlet ☒ Java



localhost:8080/chapter14/Objec... x + - □ ×

← → ↻ ⓘ localhost:8080/chapter14/Objec... ☆ 🛠️ 📄 ⋮

이름 : 홍길동
이메일 : hong@abc.com
관심분야 : JSP Java

자바 빈즈와 EL

■ 자바 빈즈와 EL의 사용

- 빈즈의 속성도 EL을 사용해 접근해 추출할 수 있음
 - `getter()` 메서드를 사용하지 않고 빈즈 이름과 속성의 이름으로 추출 가능
- 사용 형식

```
${ 빈즈이름.속성이름 }
```

```
1 package chapter14;
2
3 public class ELMemberBean {
4
5     private String id;
6     private String pwd;
7     private String name;
8     private String email;
9
10    public String getId() {
11        return id;
12    }
13
14    public void setId(String id) {
15        this.id = id;
16    }
17
18    public String getPwd() {
19        return pwd;
20    }
21
22    public void setPwd(String pwd) {
23        this.pwd = pwd;
24    }
25
26    public String getName() {
27        return name;
28    }
29 }
```

```
29
30 public void setName(String name) {
31     this.name = name;
32 }
33
34 public String getEmail() {
35     return email;
36 }
37
38 public void setEmail(String email) {
39     this.email = email;
40 }
41
42 }
```

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6   <meta charset="UTF-8">
7 </head>
8 <body>
9
10   <form name="user" method="post" action="ELMemberProc1.jsp">
11
12     <table border=1 width=300 cellpadding=5 cellspacing=0>
13       <tr>
14         <td width=80>아이디</td>
15         <td width=220><input type=text name="id" size=20></td>
16       </tr>
17       <tr>
18         <td width=80>패스워드</td>
19         <td width=220><input type=password name="pwd" size=20></td>
20       </tr>
21       <tr>
22         <td width=80>이름</td>
23         <td width=220><input type=text name="name" size=20></td>
24       </tr>
25       <tr>
26         <td width=80>메일</td>
27         <td width=220><input type=text name="email" size=20></td>
28       </tr>
29     </table>
```

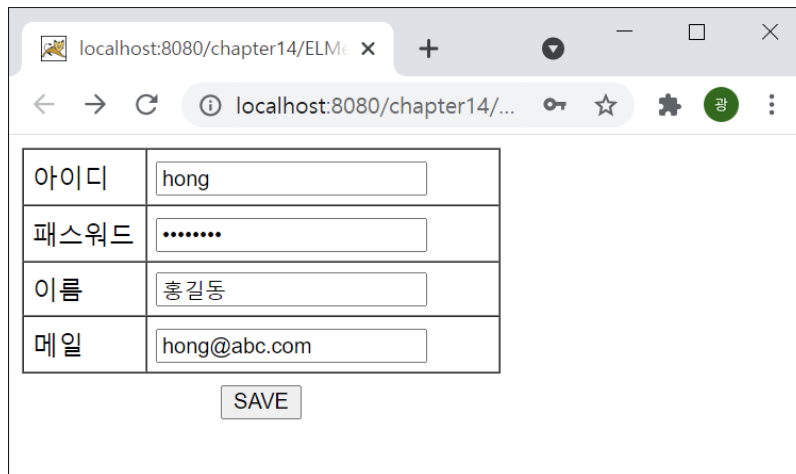
```
30
31<table width=300 cellpadding=5 cellspacing=0>
32<tr align=center>
33<td width=220><input type=submit value="SAVE" size=20></td>
34</tr>
35</table>
36
37</form>
38
39</body>
40</html>
```

```

1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <% request.setCharacterEncoding("UTF-8"); %>
4
5 <jsp:useBean id="member" class="chapter14.ELMemberBean"/>
6 <jsp:setProperty name="member" property="*" />
7
8 <!DOCTYPE html>
9 <html>
10 <head>
11   <meta charset="UTF-8">
12 </head>
13 <body>
14   <table border=1 width=300 cellpadding=5 cellspacing=0>
15
16     <tr>
17       <td width=80>아이디</td>
18       <td width=220>${member.id}</td>
19     </tr>
20
21     <tr>
22       <td width=80>패스워드</td>
23       <td width=220>${member.pwd}</td>
24     </tr>
25
26     <tr>
27       <td width=80>이름</td>
28       <td width=220>${member.name}</td>
29     </tr>

```

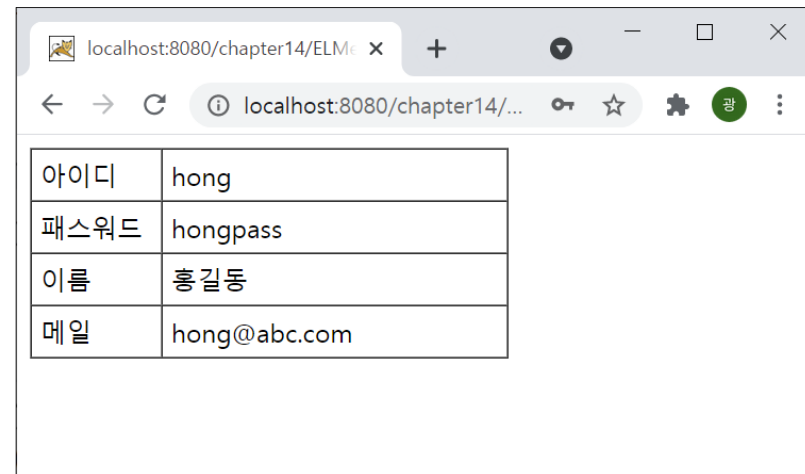
```
30
31     <tr>
32         <td width=80>메일</td>
33         <td width=220>${member.email}</td>
34     </tr>
35 </table>
36
37 </body>
38 </html>
```



localhost:8080/chapter14/ELM...

아이디	<input type="text" value="hong"/>
패스워드	<input type="password" value="....."/>
이름	<input type="text" value="홍길동"/>
메일	<input type="text" value="hong@abc.com"/>

SAVE



localhost:8080/chapter14/ELM...

아이디	hong
패스워드	hongpass
이름	홍길동
메일	hong@abc.com


```
1 package chapter14;
2
3 public class ELMemberBean {
4
5     private String id;
6     private String pwd;
7     private String name;
8     private String email;
```

기존 내용에 생성자 추가

```
10 public ELMemberBean() {
11
12 }
13
14 public ELMemberBean(String id, String pwd, String name, String email) {
15     this.id=id;
16     this.pwd=pwd;
17     this.name=name;
18     this.email=email;
19 }
```

```
20
21 public String getId() {
22     return id;
23 }
24
25 public void setId(String id) {
26     this.id = id;
27 }
```

```
28
29 public String getPwd() {
30     return pwd;
31 }
32
33 public void setPwd(String pwd) {
34     this.pwd = pwd;
35 }
36
37 public String getName() {
38     return name;
39 }
40
41 public void setName(String name) {
42     this.name = name;
43 }
44
45 public String getEmail() {
46     return email;
47 }
48
49 public void setEmail(String email) {
50     this.email = email;
51 }
52
53 }
```

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6   <meta charset="UTF-8">
7 </head>
8 <body>
9
10 <form name="user" method="post" action="ELMemberProc2.jsp">
11
12   <table border=1 width=300 cellpadding=5 cellspacing=0>
13     <tr>
14       <td width=80>아이디</td>
15       <td width=220><input type="text" name="id" size=20></td>
16     </tr>
17     <tr>
18       <td width=80>패스워드</td>
19       <td width=220><input type="password" name="pwd" size=20></td>
20     </tr>
21     <tr>
22       <td width=80>이름</td>
23       <td width=220><input type="text" name="name" size=20></td>
24     </tr>
25     <tr>
26       <td width=80>메일</td>
27       <td width=220><input type="text" name="email" size=20></td>
28     </tr>
29   </table>
```

action 속성 값 변경

```
30
31  <table width=300 cellpadding=5 cellspacing=0>
32    <tr align=center>
33      <td width=220><input type=submit value="SAVE" size=20></td>
34    </tr>
35  </table>
36
37 </form>
38
39 </body>
40 </html>
```

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <%@ page import="chapter14.*" %>
4 <% request.setCharacterEncoding("UTF-8"); %>
5
6 <jsp:useBean id="memberList" class="java.util.ArrayList" />
7
8 <jsp:useBean id="member1" class="chapter14.ELMemberBean"/>
9 <jsp:setProperty name="member1" property="*" />
10 <%
11     ELMemberBean member2 = new ELMemberBean("lee", "leepass", "이순신", "lee@abc.com");
12     memberList.add(member1);
13     memberList.add(member2);
14 %>
15
16 <!DOCTYPE html>
17 <html>
18 <head>
19     <meta charset="UTF-8">
20 </head>
21 <body>
22     <table border=1>
23         <tr align="center" bgcolor="#99ccff">
24             <td width="100"><b>아이디</b></td>
25             <td width="100"><b>비밀번호</b></td>
26             <td width="100"><b>이름</b></td>
27             <td width="200"><b>이메일</b></td>
28         </tr>
```

```

29
30<tr align="center">
31    <td>${memberList[0].id}</td>
32    <td>${memberList[0].pwd}</td>
33    <td>${memberList[0].name}</td>
34    <td>${memberList[0].email}</td>
35</tr>
36
37<tr align="center">
38    <td>${memberList[1].id}</td>
39    <td>${memberList[1].pwd}</td>
40    <td>${memberList[1].name}</td>
41    <td>${memberList[1].email}</td>
42</tr>
43</table>
44</body>
45</html>
    
```

localhost:8080/chapter14/ELMer x +

localhost:8080/chapter14/...

아이디	<input type="text" value="hong"/>
패스워드	<input type="password" value="....."/>
이름	<input type="text" value="홍길동"/>
메일	<input type="text" value="hong@abc.com"/>

SAVE

localhost:8080/chapter14/ELMer x +

localhost:8080/chapter14/ELMem...

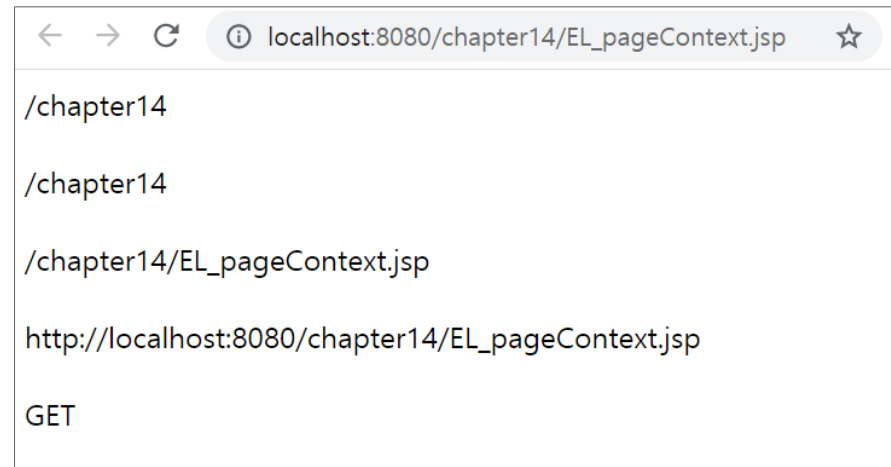
아이디	비밀번호	이름	이메일
hong	hongpass	홍길동	hong@abc.com
lee	leepass	이순신	lee@abc.com

pageContext 내장 객체

■ pageContext 내장 객체

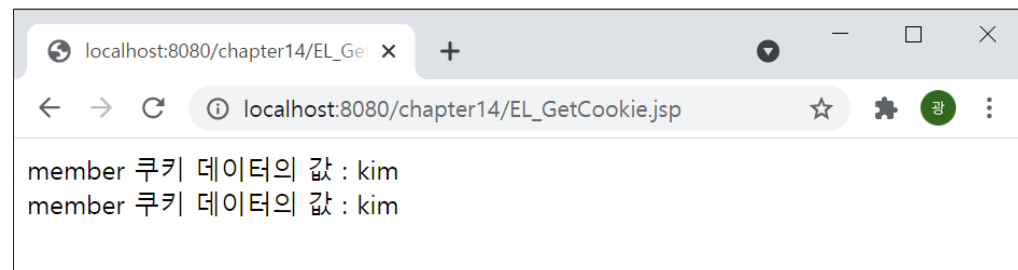
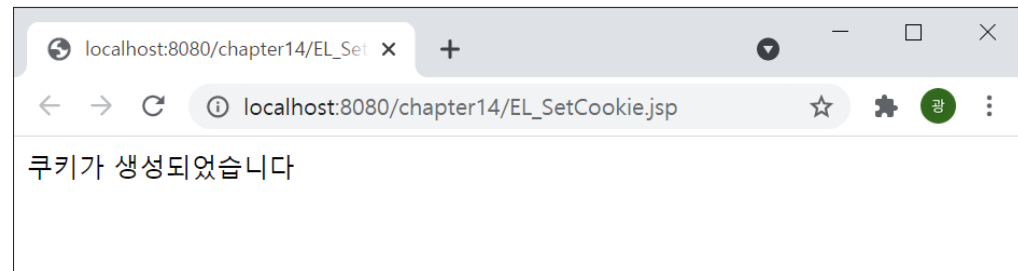
- JSP의 pageContext 기본 객체에 해당하는 EL의 내장 객체
 - ▶ 현재 JSP 페이지에 대한 모든 정보를 가지고 있는 객체
- EL의 내장 객체 중에는 request와 response 객체가 없음
 - ▶ JSP 기본 내장 객체인 request와 response와 같은 객체의 정보를 얻을 수 없음
 - ▶ pageContext 내장 객체를 사용하면 기본 객체의 정보를 얻을 수 있음
- 사용 예
 - ▶ `pageContext.request.contextPath`
 - 현재의 context 정보를 추출할 수 있음 (유용하게 사용됨)
 - JSP의 `request.getContextPath()`와 동일한 기능
 - ▶ `pageContext.method`
 - 현재의 method 타입을 추출할 수 있음

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6   <meta charset="UTF-8">
7 </head>
8 <body>
9
10   ${pageContext.request.contextPath}
11   <br><br>
12
13   <%= request.getContextPath() %>
14   <br><br>
15
16   ${pageContext.request.requestURI}
17   <br><br>
18
19   ${pageContext.request.requestURL}
20   <br><br>
21
22   ${pageContext.request.method}
23
24 </body>
25 </html>
```




```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3
4 <%
5   Cookie cookie = new Cookie("member","kim");
6   response.addCookie(cookie);
7 %>
8
9 <!DOCTYPE html>
10 <html>
11 <head>
12   <meta charset="UTF-8">
13 </head>
14 <body>
15
16   쿠키가 생성되었습니다
17
18 </body>
19 </html>
```

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2     pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6     <meta charset="UTF-8">
7 </head>
8 <body>
9
10     member 쿠키 데이터의 값 : ${cookie.member.value}
11     <br>
12     member 쿠키 데이터의 값 : ${cookie["member"]["value"]}
13
14 </body>
15 </html>
```



수고하셨습니다