

01. 표현 언어

1. 표현 언어란

- 표현 언어(Expression Language)는 처음 JSTL(JSP Standard Tag Library)이 소개되었을 때 나온 것으로, MVC 패턴에 따라 뷰(view) 역할을 수행하는 JSP를 더욱 효과적으로 만들려는 목적으로 개발되었다.
- 표현 언어는 간단한 방법으로 데이터를 표현하기 위해 고안된 언어인 SPEL(Simplest Possible Expression Language)에 기본을 두고 있다.

JSP 표현식
사용

```
<H2>
<jsp:useBean id="test" class="TestBean" />
<%= test.getName() %> 혹은 <jsp:getProperty name="test" property="name" />
</H2>
```



EL 사용

```
<H2>
${test.name}
</H2>
```

01. 표현 언어

- 표현 언어 사용을 위해서는 현재 페이지에서 출력하고자 하는 데이터가 미리 확보 되어 있어야 한다. 예를 들면 page, request, application, session 내장 객체중 하나에 사용하고자 하는 객체가 있어야만 표현언어를 이용해 데이터 출력이 가능 하다.
- 표현언어의 기본적인 문법은 다음과 같다.

- 표현 언어는 '\$'로 시작한다.
- 모든 내용은 '{표현식}'과 같이 구성된다.
- 표현식에는 기본적으로 변수 이름, 혹은 '객체_이름.멤버변수_이름'구조로 이루어진다.
- 표현식에는 부가적으로 숫자, 문자열, boolean, null과 같은 상수 값도 올 수 있다.
- 표현식에는 기본적인 연산을 할 수 있다.

2. 표현 언어에서 사용할 수 있는 내장 객체

- 표현언어에서는 객체가 생성되어 전달된다는 것을 가정하고 있다. 따라서 표현언어 에서 사용 시점에 객체를 선언할 필요가 없음.
- 표현언어에서는 다음과 같이 객체에 접근 할 수 있음.

`${member.id}` 혹은 `${member["id"]}` → member 객체의 getId() 메서드 호출과 동일
`${row[0]}` → row라는 이름의 컬렉션 객체의 첫 번째 값

- 또한 몇몇 내장객체를 통해 컨테이너가 제공하는 다른 객체에 접근할 수 있는 방법을 제공하고 있다.
- 표현언어에서 사용할 수 있는 내장객체는 [표 10-1] 참조

01. 표현 언어

[표 10-1] 표현 언어에서 사용할 수 있는 내장객체

내장객체	기능
pageScope	page 범위에 포함된 속성 값에 접근할 수 있는 객체다.
requestScope	request 범위에 포함된 속성 값에 접근할 수 있는 객체다.
sessionScope	session 범위에 포함된 속성 값에 접근할 수 있는 객체다.
applicationScope	application 범위에 포함된 속성 값에 접근할 수 있는 객체다.
param	request.getParameter("xxx")로 얻을 수 있는 값들이다. \${param.xxx}처럼 사용한다.
paramValues	request.getParameterValues("xxx")와 동일한 기능을 수행한다. \${paramValues.xxx}처럼 사용한다.
header	request.getHeader("xxx")와 동일한 기능을 수행한다. \${header.xxx}처럼 사용한다.
headerValues	request.getHeaderValues("xxx")와 동일한 기능을 수행한다. \${headerValues.xxx}처럼 사용한다.
initParam	컨텍스트의 초기화 매개변수 값이다.
cookie	쿠키 정보에 접근할 수 있는 객체다.
pageContext	pageContext 객체다.

01. 표현 언어

3. 표현 언어에서 사용할 수 있는 연산자

- 표현 언어에서는 표현식 부분에서 기본적인 연산을 할 수 있으며 다음과 같은 연산자 사용이 가능하다.

[표 10-2] 산술 연산자

연산자	기능	연산자	기능
+	더하기	-	빼기
*	곱하기	/ or div	나누기
% of mod	몫		

[표 10-3] 비교/조건 연산자

연산자	기능	연산자	기능
== 혹은 eq	같다.	!= 혹은 ne	같지 않다.
< 혹은 lt	좌변이 우변보다 작다.	> 혹은 gt	좌변이 우변보다 크다.
<= 혹은 le	좌변이 우변보다 같거나 작다.	>= 혹은 ge	좌변이 우변보다 같거나 크다.
a?b : c	a가 참이면 b, 거짓이면 c를 반환한다.		

[표 10-4] 관계 연산자

연산자	기능
&& 혹은 and	AND 연산
혹은 or	OR 연산
! 혹은 not	NOT