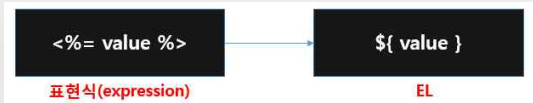


# 제14장

## EL,JSTL태그 라이브러리

# 1. EL(Expression Language)란?

EL(Expression Language)란, 표현식 또는 액션태그를 대신해서 값을 표현하는 언어입니다.



EL 연산자

산술 : +, -, \*, / %  
관계형 : ==, !=, <, >, <=, >=  
조건 : a ? b : c  
논리 : &&, ||

## 2. 액션태그로 사용되는 EL

액션태그를 아래와 같이 EL표기법으로 사용하면, 가독성과 코드의 절약성이 좋아진다.

```
<jsp:getProperty name="member" property="name"/>
```



```
${ member.name }
```

### 3. EL의 내장객체

pageScope : page객체를 참조하는 객체  
requestScope : request객체를 참조하는 객체  
sessionScope : session객체를 참조하는 객체  
applicationScope : application객체를 참조하는 객체

param : 요청 파라미터를 참조하는 객체  
paramValues : 요청 파라미터(배열)를 참조하는 객체  
initParam : 초기화 파라미터를 참조하는 객체  
cookie : cookie객체를 참조하는 객체

## 4. JSTL(JSP Standard Tag Library)의 개요 및 설치

JSP의 경우 HTML태그와 같이 사용되어 전체적인 코드의 가독성이 떨어진다. 하여, 이러한 단점을 보완하고자 만들어진 태그 라이브러리가 JSTL이다. JSTL의 경우 우리가 사용하는 Tomcat컨테이너에 포함되어 있지 않으므로, 별도의 설치를 하고 사용한다.

### 1. JSTL 설치

<http://jakarta.apache.org/> 접속 한 후,  
좌측의 Taglibs 클릭

#### Download

Which version?  
Tomcat 10 (alpha)  
Tomcat 9  
Tomcat 8  
Tomcat 7  
Tomcat Connectors  
Tomcat Native  
**Taglibs**  
Archives

### 2. Apache Standard Taglib 클릭



## 4. JSTL(JSP Standard Tag Library)의 개요 및 설치

### 3. Standard 2.0 다운로드

#### Standard Taglib

##### JSP(tm) Standard Tag Library implementations

Apache hosts the Apache Standard Taglib, an implementation of the JSP Standard Tag Library (JSTL) specification. Various versions are available.

Version	JSTL version	Requirements	Getting the Taglib
Standard 1.2.1	JSTL 1.2	Servlet 2.5, JavaServer Pages 2.1	<a href="#">download</a>  (javadoc)
Standard 1.1	JSTL 1.1	Servlet 2.4, JavaServer Pages 2.0	<a href="#">download</a> 
Standard 1.0	JSTL 1.0	Servlet 2.3, JavaServer Pages 1.2	<a href="#">download</a> 

### 4. binary클릭

#### archive.apache.org

This site contains the historical archive of

For current releases, please visit the [mirror](#)

[Name](#)

[Last m](#)



[Parent Directory](#)



[binaries/](#)

2005-1



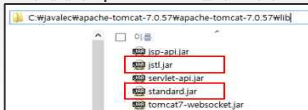
[source/](#)

2005-1

### 5. jakarta-taglibs-standard-1.1.2.zip클릭

	<a href="#">jakarta-taglibs-standard-1.1.1.tar.gz.asc</a>	2004-07-19 21:53 186
	<a href="#">jakarta-taglibs-standard-1.1.1.zip</a>	2004-07-19 21:53 931K
	<a href="#">jakarta-taglibs-standard-1.1.1.zip.asc</a>	2004-07-19 21:53 186
	<a href="#">jakarta-taglibs-standard-1.1.2.tar.gz</a>	2004-10-26 20:57 873K
	<a href="#">jakarta-taglibs-standard-1.1.2.tar.gz.asc</a>	2004-10-26 20:57 186
	<a href="#">jakarta-taglibs-standard-1.1.2.zip</a>	2004-10-26 20:57 938K
	<a href="#">jakarta-taglibs-standard-1.1.2.zip.asc</a>	2004-10-26 20:57 186
	<a href="#">jakarta-taglibs-standard-oidxml-compat.tar.gz</a>	2002-06-21 22:59 1.1M
	<a href="#">jakarta-taglibs-standard-oidxml-compat.tar.gz.asc</a>	2002-06-21 22:59 232
	<a href="#">jakarta-taglibs-standard-oidxml-compat.zip</a>	2002-06-21 23:01 1.1M

### 6. 압축 해제 후 apache-tomcat폴더로 복사



## 5. JSTL의 라이브러리

JSTL에서는 다섯 가지의 라이브러리를 제공 합니다. ( Core, XML Processing, I18N formatting, SQL, Functions )

lib	URI	Prefix	ex
Core	<a href="http://java.sun.com/jsp/jstl/core">http://java.sun.com/jsp/jstl/core</a>	c	<c:tag
XML Processing	<a href="http://java.sun.com/jsp/jstl/xml">http://java.sun.com/jsp/jstl/xml</a>	x	<x:tag
I18N formatting	<a href="http://java.sun.com/jsp/jstl/fmt">http://java.sun.com/jsp/jstl/fmt</a>	fmt	<fmt:tag
SQL	<a href="http://java.sun.com/jsp/jstl/sql">http://java.sun.com/jsp/jstl/sql</a>	sql	<sql:tag
Functions	<a href="http://java.sun.com/jsp/jstl/functions">http://java.sun.com/jsp/jstl/functions</a>	fn	fn:function()

### Core

Core 라이브러리는 기본적인 라이브러리로 출력, 제어문, 반복문 같은 기능이 포함되어 있습니다.

```
<%@ taglib uri=http://java.sun.com/jsp/jstl/core prefix="c" %>
```

## 5. JSTL의 라이브러리

출력 태그 : <c:out>

```
<c:out value="출력값" default="기본값" escapeXml="true or false">
```

변수 설정 태그 : <c:set>

```
<c:set var="변수명" value="설정값" target="객체" property="값" scope="범위">
```

변수를 제거하는 태그 : <c:remove>

```
<c:remove var="변수명" scope="범위">
```

예외 처리 태그 : <c:catch>

```
<c:catch var="변수명">
```



## 5. JSTL의 라이브러리

제어문(if) 태그 : <c:if>

```
<c:if test="조건" var="조건 처리 변수명" scope="범위">
```

제어문(switch) 태그 : <c:choose>

```
<c:choose>  
<c:when test="조건"> 처리 내용 </c:when>  
<c:otherwise> 처리 내용 </c:otherwise>  
</c:choose>
```

반복문(for) 태그 : <c:forEach>

```
<c:forEach items="객체명" begin="시작 인덱스" end="끝 인덱스" step="증감식"  
var="변수명" varStatus="상태변수">
```

## 5. JSTL의 라이브러리

페이지 이동 태그 : <c:redirect>

---

```
<c:redirect url="url">
```

파라미터 전달 태그 : <c:param>

---

```
<c:param name="파라미터명" value="값">
```

감사합니다.

