

# JSTL (JSP Standard Tag Library)

2.3

이번 Chapter에서는 JSP가 제공하는 표준 커스텀 태그인 JSTL의 기본 개요와 태그의 사용법을 학습한다.

# 목차(1/2)

Section  
01

## JSTL의 개요

Section  
02

## JSTL의 사용

**JSTL을 사용하기 위한 라이브러리 추가하기**

**JSTL 라이브러리 개요 및 사용 방법**

**JSTL Core 태그 라이브러리**

# 목차(2/2)

Section  
02

## JSTL의 사용

**JSTL XML 태그 라이브러리**

**JSTL I18N 태그 라이브러리**

**JSTL SQL 태그 라이브러리**

**JSTL Functions 라이브러리**

- JSTL(JSP Standard Tag Library) 은 표준 커스텀 태그(Custom Tag)
- ▣ 커스텀 태그 : 사용자 정의 태그
- JSTL은 JSP 페이지의 로직을 담당하는 부분인 제어문 및 데이터베이스 처리 등을 표준 커스텀 태그로 제공
- ▣ 코드를 깔끔하게 하고 가독성을 좋게 함

- JSTL과 커스텀 태그도 XML기반
- ▣ 시작 태그와 종료 태그를 쌍으로 작성
  - 시작 태그와 종료 태그가 존재하는 경우

```
<c:if test="empty ${c.getEmail}"> <!--시작 태그-->
    <input type="text" name="email"><!--태그 내용-->
</c:if><!--종료 태그-->
```
- ▣ 종료 태그가 없는 단독 태그의 경우
  - 태그를 닫는 기호(>)전에 /를 입력해 태그의 종료를 알림

```
<c:set var="name" scope="page"/><!--단독 태그-->
```

## C JSTL을 사용하기 위한 라이브러리 추가하기

- ▣ JSTL 라이브러리 다운로드
  - JSTL 라이브러리 다운로드 사이트
    - <https://jstl.java.net>
  - 라이브러리 다운로드
    - javax.servlet.jsp.jstl-api-1.2.1.jar와 javax.servlet.jsp.jstl-1.2.1.jar 다운로드
- ▣ 라이브러리 배치
  - [프로젝트]-[WEB-INF]-[lib]에 라이브러리 복사

## C JSTL 라이브러리 개요 및 사용 방법

- ▣ JSTL은 많은 태그를 제공하며, 이들을 사용하기 위해 각각 네임스페이스로 라이브러리를 제공
- Core(코어), XML, I18N(국제화, 포매팅), SQL(데이터베이스), Functions(함수)

## C JSTL 라이브러리 개요 및 사용 방법

- ▣ 라이브러리들은 URI로 제공되며 이들을 사용할 때는 접두어(prefix)를 사용
- ▣ JSTL 라이브러리에서 제공하는 태그를 사용
  - taglib 디렉티브에 사용할 라이브러리의 prefix 속성과 uri 속성에 해당하는 값을 기술
    - ▶ 예 : <%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>

## C JSTL Core 태그 라이브러리

- ▣ 라이브러리
  - Core(코어)
- ▣ URI
  - <http://java.sun.com/jsp/jstl/core>
- ▣ Prefix
  - c
- ▣ 제공 가능
  - 변수의 선언 및 삭제 등의 변수와 관련된 작업
  - if, for문 등과 같은 제어문
  - URL처리 및 그밖에 예외처리 및 화면 출력

## C JSTL Core 태그 라이브러리

### ▣ Core 태그 리스트

#### ■ <c:catch>

- body 위치에서 실행되는 코드의 예외를 잡아내는 역할을 담당

#### ■ <c:choose>

- 자바의 switch 문과 같지만, 조건에 문자열 비교도 가능하고 쓰임의 범위가 넓음. 서브 태그로 <when>과 <otherwise>를 가짐

#### ■ <c:when>

- choose의 서브 태그로 조건의 비교 시에 조건을 만족한 경우에 사용

#### ■ <c:otherwise>

- 조건을 만족하지 못한 경우에 사용

## C JSTL Core 태그 라이브러리

### ▣ Core 태그 리스트

- <c:if>
  - 조건문을 사용할 때 씀
- <c:import>
  - 웹 애플리케이션 내부의 자원 접근은 물론이고, http, ftp 같은 외부에 있는 자원(html, jsp등)을 가져옴
- <c:param>
  - 파라미터 사용 시 필요. <import> 태그의 URL 뒤에 파라미터로 붙여서 사용되기도 함
- <c:forEach>
  - 객체 전체에 걸쳐 반복 실행을 할 때 사용

## C JSTL Core 태그 라이브러리

### ■ Core 태그 리스트

- <c:forToken>
  - 자바의 StringTokenizer 클래스를 사용하는 것과 같음
- <c:out>
  - 화면 출력. JSP의 표현식 대체
- <c:redirect>
  - response.sendRedirect()를 대체하는 태그로 지정한 다른 페이지로 이동

## C JSTL Core 태그 라이브러리

### ▣ Core 태그 리스트

#### ■ <c:remove>

- JSP의 removeAttribute()와 같은 역할.  
(page|request|session|application) 범위의 변수(속성)  
를 제거

#### ■ <c:set>

- JSP의 setAttribute()와 같은 역할.  
(page|request|session|application) 범위의 변수(속성)  
를 설정

#### ■ <c:url>

- 쿼리 파라미터로 부터 URL을 생성

## C JSTL XML 태그 라이브러리

- ▣ 라이브러리
  - XML
- ▣ URI
  - <http://java.sun.com/jsp/jstl/xml>
- ▣ Prefix
  - X
- ▣ 제공 기능
  - XML 데이터 접근 및 파싱, 흐름 제어 및 XML 변환 등

## C JSTL XML 태그 라이브러리

### ▣ XML 태그 리스트

- <x:choose>
  - 이 태그는 <c:choose> 태그와 기능이 같음
- <x:out>
  - XPath에 지정한 패턴에 따라 xml 내용을 출력한다.
- <x:if>
  - 조건문에 사용
- <x:forEach>
  - 이 태그는 XPath에 따라서 해당하는 엘리먼트 수만큼 반복해서 수행

## C JSTL XML 태그 라이브러리

### ▣ XML 태그 리스트

- <x:otherwise>
  - 조건이 false일 때 수행 <c:otherwise>와 같음
- <x:param>
  - 파라미터의 사용 시에 필요
- <x:parse>
  - xml 문서를 읽어서 파싱

## C JSTL XML 태그 라이브러리

### ▣ XML 태그 리스트

- <x:set>
  - XPath에 따라 선택된 내용을 변수에 저장
- <x:transform>
  - xml과 xslt 파일을 결합해서 새로운 형식의 문서를 생성
- <x:when>
  - 조건이 true일 때 수행 <c:when>과 같음

## C JSTL I18N 태그 라이브러리

- ▣ 라이브러리
  - I18N(국제화, 포매팅)
- ▣ URI
  - <http://java.sun.com/jsp/jstl/fmt>
- ▣ Prefix
  - fmt
- ▣ 제공 기능
  - 로케일, 메시지, 숫자 문자 형식 등

## C JSTL I18N 태그 라이브러리

### ▣ JSTL I18N 태그 리스트

- <fmt:requestEncoding>
  - request.setCharacterEncoding()와 같은 역할
- <fmt:setLocale>
  - 다국어를 지원하는 페이지를 만들 경우 사용
  - ResourceBundle로 불러오는 \*.properties 파일들과 연계
- <fmt:timeZone>
  - 타임 존(Time Zone)을 적용할 때 사용
- <fmt:setTimeZone>
  - 특정 scope의 타임 존을 설정할 때 사용

## C JSTL I18N 태그 라이브러리

### ▣ JSTL I18N 태그 리스트

- <fmt:bundle>
  - properties 확장자를 사용하는 자원 파일을 읽어오는 역할
- <fmt:setBundle>
  - 페이지 전체에서 사용할 수 있는 번들(bundle)을 지정하는데 사용
- <fmt:message>
  - <fmt:bundle> 태그에서 정한 값들을 가져옴
- <fmt:param>
  - <fmt:message> 태그의 서브태그
  - <fmt:message> 태그에서 설정하지 않은 값을 설정

## C JSTL I18N 태그 라이브러리

### ▣ JSTL I18N 태그 리스트

- <fmt:formatNumber>
  - 숫자 형식을 표현할 때 사용
- <fmt:parseNumber>
  - 문자열로부터 수치를 파싱
  - 즉, 문자열을 숫자로 변환 시에 사용
- <fmt:formatDate>
  - 날짜 형식을 표현할 때 사용
- <fmt:parseDate>
  - 문자열에서 날짜를 파싱
  - 즉, 문자열을 날짜로 변환 시에 사용



# JSTL의 사용

## C JSTL SQL 태그 라이브러리

- ▣ 라이브러리
  - SQL(데이터베이스)
- ▣ URI
  - <http://java.sun.com/jsp/jstl/sql>
- ▣ Prefix
  - sql
- ▣ 제공 기능
  - SQL 처리 등

## C JSTL SQL 태그 라이브러리

### ▣ JSTL SQL 태그 리스트

- <sql:transaction>
  - 트랜잭션을 구현할 때 사용
- <sql:query>
  - sql 태그의 속성 또는 body에 정의된 SQL 쿼리 문장을 실행 executeQuery()와 같은 기능
- <sql:update>
  - sql 태그의 속성 또는 body에 정의된 SQL 쿼리 문장을 실행 executeUpdate()와 같은 기능

## C JSTL SQL 태그 라이브러리

### ▣ JSTL SQL 태그 리스트

- <sql:param>
  - java.sql.PreparedStatement.setString()과 같은 역할
- <sql:dateParam>
  - java.sql.PreparedStatement.setTimestamp과 같은 역할
- <sql:setDataSource>
  - DataSource를 지정



# JSTL의 사용

## C JSTL Functions 라이브러리

- ▣ 라이브러리
  - Functions(함수)
- ▣ URI
  - <http://java.sun.com/jsp/jstl/functions>
- ▣ Prefix
  - fn
- ▣ 제공 기능
  - 함수를 사용한 문자열 및 컬렉션 처리 등

## C JSTL Functions 라이브러리

### ▣ JSTL Functions 라이브러리의 함수 리스트

- boolean contains(String string, String substring)
  - string이 substring을 포함하면 true값을 리턴
- boolean containsIgnoreCase(String string, String substring)
  - 대소문자에 관계없이, string이 substring을 포함하면 true값을 리턴
- boolean endsWith(String string, String substring)
  - string이 substring로 끝나면 true값을 리턴

## C JSTL Functions 라이브러리

### ▣ JSTL Functions 라이브러리의 함수 리스트

#### ■ String escapeXml(String string)

- string에 있는 < > & ' " 문자들을 각각 &lt; &gt; &#039; &#034;로 바꿔준 뒤 문자열을 리턴

#### ■ int indexOf(String string, String substring)

- string에서 substring이 처음으로 나타나는 인덱스 번호를 리턴

#### ■ String join(String[] array, String separator)

- 배열 요소들을 separator를 구분자로 하여 모두 연결해서 리턴

## C JSTL Functions 라이브러리

### ▣ JSTL Functions 라이브러리의 함수 리스트

#### ■ int length(Object item)

- item이 배열이나 컬렉션이면 요소의 개수를, 문자열이면 문자의 개수를 리턴

#### ■ String replace(String string, String before, String after)

- string 내에 있는 before 문자열을 after 문자열로 모두 바꿔서 리턴

#### ■ String[] split(String string, String separator)

- string 내의 문자열을 separator에 따라 분할하고, 이 분할한 문자열들을 배열로 구성해서 리턴

## C JSTL Functions 라이브러리

### ▣ JSTL Functions 라이브러리의 함수 리스트

- boolean startsWith(String string, String prefix)
  - ▣ string이 prefix로 시작하면 true값을 리턴
- String substring(String string, int begin, int end)
  - ▣ string에서 begin인덱스 번호부터 end인덱스 번호전 까지의 문자열을 리턴
- String substringAfter(String string, String substring)
  - ▣ string에서 substring이 나타난 이후 부분에 있는 문자열을 리턴

## C JSTL Functions 라이브러리

### ▣ JSTL Functions 라이브러리의 함수 리스트

- String substringBefore(String string, String substring)
  - string에서 substring이 나타나기 이전 부분에 있는 문자열을 리턴
- String toLowerCase(String string)
  - string을 모두 소문자로 바꿔 리턴
- String toUpperCase(String string)
  - string을 모두 대문자로 바꿔 리턴
- String trim(String string)
  - string 앞뒤의 공백을 모두 제거하여 리턴