**JSTL(JSP Standard Tag Library)**

* 1. JSTL 개요
     1. 라이브러리란 여러 프로그램이 공통으로 사용하는 코드를 모아 놓은 코드의 집합
     2. JSTL의 경우 우리가 사용하는 Tomcat컨테이너에 포함되어 있지 않으므로, 별도의 설치를 하고 사용한다.
  2. JSTL을 가지고 할 수 있는 일
     1. 간단한 프로그램 로직의 구사(자바의 변수 선언, if문, for문 등에 해당하는 로직)
     2. 다른 JSP 페이지 호출(<c:redirect>, <c:import>)
     3. 날짜, 시간, 숫자의 포맷
     4. JSP 페이지 하나를 가지고 여러 언어의 웹 페이지 생성

|  |  |  |  |
| --- | --- | --- | --- |
| lib | uri | prefix | ex |
| **Core** | <http://java.sun.com/jsp/jstl/core>  일반 프로그래밍 언어에서 제공하는 것과 유사한 변수 선언, 실행 흐름의 제어기능을 제공하고 다른 JSP페이지로 제어를 이동하는 기능도 제공 | c | <c:tag |
| XML processing | <http://java.sun.com/jsp/jstl/xml>  XML문서를 처리할 때 필요한 기능을 제공 | x | <x:tag |
| **I18N formatting** | <http://java.sun.com/jsp/jstl/fmt>  숫자, 날짜, 시간을 포맷팅하는 기능과 국제화, 다국어 지원기능을 제공 | fmt | <fmt:tag |
| SQL | <http://java.sun.com/jsp/jstl/sql>  데이터베이스의 데이터를 입력/수정/삭제/조회하는 기능을 제공 | Sql | <sql:tag |
| Functions | <http://java.sun.com/jsp/jstl/functions>  문자열을 처리하는 함수를 제공 | fn | fn:function() |

**가장 먼저 해야 할 일은 페이지 지시자에 태그라이브러리를 사용한다고 명시해 준다.**

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

접두어 라이브러리를 식별하는 라이브러리

Core 라이브러리

1. 출력 태그 <c:out>

<c:out value=“출력값” default=“기본값” escapeXml=“true or false”> false를 주면 특수기호를 해석

1. 변수 설정 태그 <c:set>

<c:set var=“변수명” value=“설정값” target=“객체” property=“값” scope=“범위”>어떤 객체의 속성값을 변수로 지정

1. 변수를 제거하는 태그 <c:remove>

<c:remove var=“변수명” scope=“범위”>

1. 예외 처리 태그 <c:catch>

<c:catch var=“변수명”>

1. 제어문(if)태그 <c:if>

<c:if test=“조건” var=“조건 처리 변수명” scope=“범위”>

1. 제어문 태그 : <c:choose>

<c:choose>

<c:when test=“조건1”> 처리 내용 </c:when>

<c:when test=“조건2”> 처리 내용 </c:when>

<c:when test=“조건3”> 처리 내용 </c:when>

<c:otherwise> 처리 내용 </c:otherwise>

</c:choose>

1. 반복문(for) 태그 : <c:forEach>

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

<c:forTokens var=“변수명” item=전체스트링 delims=구분자>

텍스트이(가) 표시된 사진

자동 생성된 설명

1. 페이지 이동태그 <c:redirect url=”url”>

<c:redirect url=“url”>

fmt 라이브러리 : : 국제화/형식화 액션. 국제화/형식화 기능을 제공해 주는 JSTL 라이브러리

* 1. 국제화는 다국어 내용을 처리, 형식화는 날짜와 숫자 형식 등을 처리

<[%@taglib prefix=”fmt” uri=”http://java.sum.com/jsp/jstl/fmt” %](mailto:%25@taglib%20prefix=)>

* 1. 인코딩 관련 태그 <fmt:requestEncoding>

Request 객체로 부터 전달 받은값들을 인코딩할 때 사용 - 보통 한글 값이 넘어올 경우 <fmt:requestEncoding value="utf-8"/>코드로 Request 객체로 받는 값을 한글에 맞게 인코딩해 준다

<% request.setCharacterEncoding(“utf-8”);%>역할

* 1. 국제화 관련 태그 <fmt:setLocale> <fmt:timeZone> <fmt:setTimeZone> <fmt:bundle> <fmt:setBundle> <fmt:message> <fmt:param>
  2. 형식화 관련 태그
     + <fmt:formatNumber>

<fmt:formatNumber value=”값” type=”타입” pattern=”패턴” currencyCode=”값” currencySymbol=”값” groupingUsed=”true or false” maxIntegerDigits=”값” minIntegerDigits=”값” maxFractionDigits=”값” minFractionDigits=”값” var=”변수명” scope=”범위”>

* + - * 태그는 숫자형식의 패턴을 설정할 때 사용
      * value 속성에는 패턴을 적용 시킬 원래의 값을 입력
      * type은 숫자의 타입을 의미. 숫자, 통화, 퍼센트중 원하는 타입으로 설정 가능
      * pattern 속성은 지정한값을 어떤 패턴으로 변화 시킬지를 정 할수 있다
      * currencyCode는 통화코드를 의미하며, 숫자타입이 통화일 경우만 유효하다
      * currencySymbol도 숫자타입이 통화일 유효하며, 통화기호를 지정할 수 있다
      * groupingUsed는 그룹기호의 포함 여부를 나타낸다
      * maxIntegerDigits는 최대정수 길이
      * minIntegerDigits는 최소 정수 길이
      * maxFractionDigits은 최대 소수점 자릿수
      * minFractionDigits는 최소 소수점 자릿수를 의미
    - <fmt:formatDate>

<fmt:formatDate value=”값” type=”타입” dateStyle=”값” timeStyle=”값” pattern=”패턴” timeZone=”값” var=”변수명” scope=”범위”>

* + - * 날짜 형식의 패턴을 설정할 때 사용되는 태그
      * value 속성에는 날짜 또는 시간을 입력
      * type 속성으로 날짜인지 시간인지 또는 날짜와 시간 둘다 포함한 타입인지를 지정
      * dateStyle은 날짜의 스타일을 지정. type 속성이 date 또는 both일 때만 적용
      * timeStyle은 시간의 스타일을 지정. type 속성이 time 또는 both일 때만 적용
      * timeZone속성은 날짜와 시간이 표시될 시간대를 지정

<**c:set** var=*"nowDate"* value=*"*<%=new Date(System.currentTimeMillis()) %>*"*/>

<**c:set** var=*"nowTimes"* value=*"*<%=new Timestamp(System.currentTimeMillis()) %>*"*/>

<**c:set** var=*"notDateUtil"* value=*"*<%=new java.util.Date() %>*"*/> //하나만 임폴트

위에서 임폴트를 하나 해놨기 때문에 java.util을 쓰려면 임폴트를 하지말고 java.util.Date처럼 써야한다

<**fmt:formatDate** value=*"*${nowDate }*"* pattern=*"yyyy년MM월dd일(E요일) h:mm:ss:SS(a)"*/><**br**> //패턴을 더 많이 쓴다.

<**fmt:formatDate** value=*"*${nowDate }*"* pattern=*"yy-MM-dd(E) HH:mm:ss:SS"*/><**br**>