

자바 웹 개발 시작하기

(9주차 : 프로젝트 구현 - 추가적인 뷰)

2011. 12. 21

DEVELOPMENT #2

이덕곤

지난 시간에는... (8주차)

- 프로젝트 설계
 - 명세서를 만들어 보자
 - 로그인과 보드의 통합
 - 단위테스트
-
- 숙제 : 프로젝트 새로 시작해서 만들어 보기
다들 시작 해 보셨나요?

이번 시간에는... (9주차)

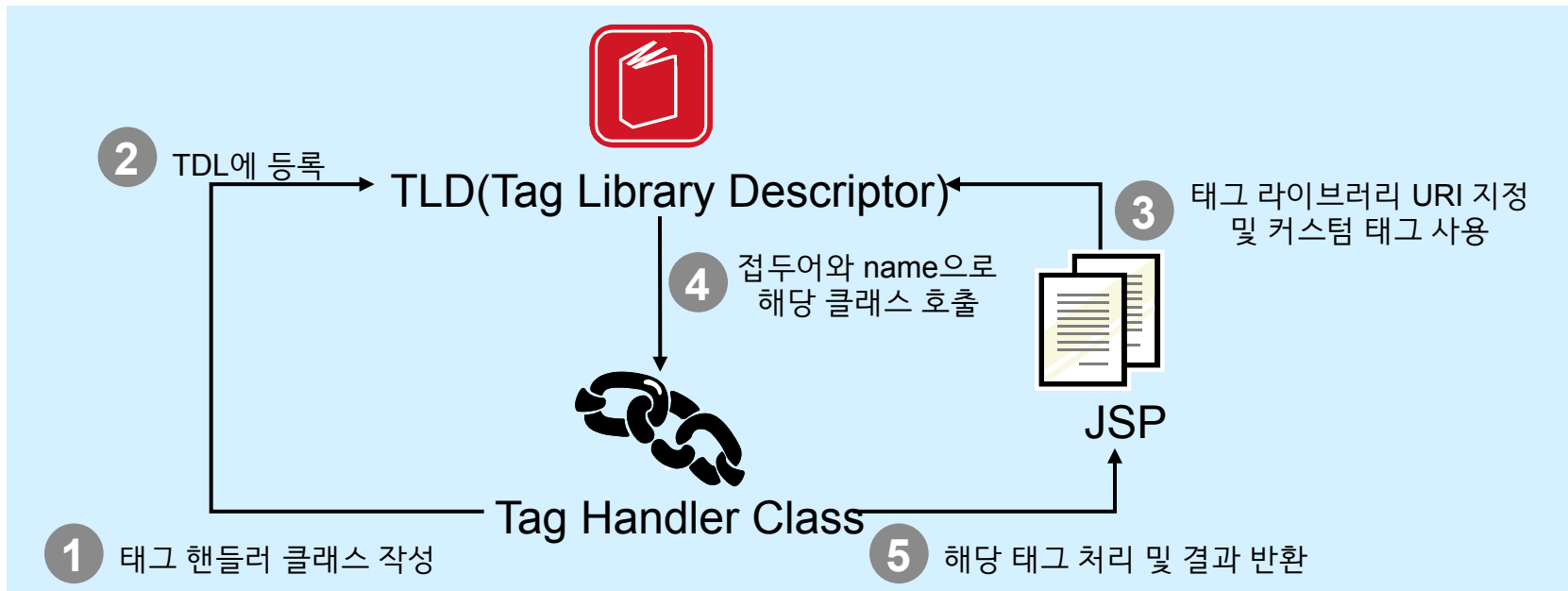
- 프로젝트 구현
- 추가적인 뷰에 대한 공부
 - 커스텀 태그
 - RSS뷰
- @Async
- 과제 : 작품 명세서 완성하기

커스텀 태그

- 커스텀 태그(Custom Tag)란?
 - 사용자가 직접 정의해서 사용하는 태그
- 태그 라이브러리(Tag Library)
 - 사용자가 정의한 태그들을 모아 라이브러리 형태로 만든 것
- 커스텀 태그의 장점
 - 한번 작성한 커스텀 태그는 어디서든 사용 가능함
 - JSP 페이지 내에 스크립트 코드를 줄임으로써 가독력을 높일 수 있음
 - 태그 라이브러리를 잘 정의함으로써 JSP를 모르는 디자이너도 쉽게 동적인 페이지를 만들 수 있음

커스텀 태그의 사용

- 태그 핸들러 클래스(Tag Handler Class)
- 태그 라이브러리 서술파일(Tag Library Descriptor, TLD)
- 태그를 사용하는 JSP 파일
 - 사용할 TLD를 정의하고 커스텀 태그를 사용



커스텀 태그 종류

- Tag
 - SimpleTagSupport 상속받아 제작
 - 여러가지 파라미터 및 필수 값 등 설정 가능
- Function
 - 하나의 메서드가 하나의 함수
 - 간단한 유틸리티를 만들 때 사용
- .tag 파일
 - 특별한 설정이나 클래스, 메서드 없음
- 상황에 맞게 사용

TLD 파일 작성

`<?xml version="1.0" encoding="UTF-8" ?>`

`<taglib xmlns="http://java.sun.com/xml/ns/javaee" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"`

`xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-jsptaglibrary_2_1.xsd"`

`version="2.1">`

`<tlib-version>2.1</tlib-version>`

`<short-name>simpleTags</short-name>`

`<uri>http://study.starpl.com/taglib/sstag</uri>`

`<tag> ... </tag>`

`<function> ... </function>`

`</taglib>`

Tag Handler Class의 구현

다양하고 많은 클래스가 있지만... **SimpleTagSupport** 사용

```
public class StarplFormatDateTag extends SimpleTagSupport
{
    @Override
    public void doTag() throws JspException, IOException
    {
        PageContext pageContext = (PageContext) getJspContext();
        ..... // 처리 할 일들

        pageContext.getOut().print(출력할_값);
    }
}
```


TLD 파일 Tag 정의

<tag>

<name>formatDate**</name>**

<tag-class>com.starpl.study.base.taglib.FormatDateTag**</tag-class>**

<body-content>empty**</body-content>**

<attribute>

<name>locale**</name>**

<required>false**</required>** <!-- 필수 값인가? -->

<rtexprvalue>true**</rtexprvalue>** <!-- EL 표현식 사용 여부 -->

</attribute>

<attribute>

.....

</attribute>

.....

</tag>

Function Class 구현

Tag Class처럼 상속받아 만들 필요가 없음

```
public class ElFunctions
{
    public static int rollDice()
    {
        int result = (int) (Math.random() * 6) + 1;
        return result;
    }
    ..... // 추가할 메서드
}
```

TLD 파일 Function 정의

<function>

<name>rollIt</name>

<function-class>

com.starpl.study.base.taglib.ElFunctions</function-class>

<function-signature>int rollDice()</function-signature>

</function>

<function>

<name>escapeXml</name>

<function-class>

com.starpl.study.base.taglib.ElFunctions</function-class>

<function-signature>

String escapeXml(java.lang.String)</function-signature>

</function>

web.xml 파일의 정의

<!-- 5. TAG-LIBRARY 설정입니다. -->

<jsp-config>

<taglib>

<**taglib-uri**>http://study.starpl.com/taglib/tag</taglib-uri>

<**taglib-location**>

/WEB-INF/tags/**taglibrary.tld**</taglib-location>

</taglib>

</jsp-config>

커스텀 태그 사용하기

- 정의
 - Tag, function : `<%@ taglib prefix="st" uri="http://study.starpl.com/taglib/tag"%>`
- 사용
 - Tag : `<st:formatDate value="<%=new Date()%>" />`
 - Function : `${st:rollIt()}`
- .tag 파일
 - 정의 : `<%@ taglib prefix="t" tagdir="/WEB-INF/tags"%>`
 - 사용 : `<t:test data="dataTest"/>`

RSS xml 뷰

`<?xml version="1.0" encoding="UTF-8"?>`

`<%@ page contentType="text/xml;charset=UTF-8" pageEncoding="UTF-8"%>`

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

`<%@ taglib prefix="st" uri="http://study.starpl.com/taglib/tag"%>`

`<rss version="2.0">`

`<channel>`

`...`

`</channel>`

`</rss>`

RSS <channel>

```
<rss version="2.0">
```

```
  <channel>
```

```
    <title><![CDATA[보드 RSS]]></title>
```

```
    <link>http://http://127.0.0.1:8080/board</link>
```

```
    <description><![CDATA[아카이브, 타임라인, 소셜 네트워킹, 블로그  
, 커뮤니티 서비스]]></description>
```

```
    <language>ko-kr</language>
```

```
    <generator>http://study.starpl.com</generator>
```

```
  </channel>
```

```
  <item> ... <item>
```

```
</rss>
```

RSS <item>

```
<c:forEach var="item" items="${boardList}">
```

```
<item>
```

```
<title><![CDATA[
```

```
<c:out value="${item.subject}" escapeXml="true" />]]></title>
```

```
<link>http://http://127.0.0.1:8080/board/view/${item.boardIdx}</link>
```

```
<description><![CDATA[${item.content}]]></description>
```

```
<author><![CDATA[
```

```
<c:out value="${item.nickName}" escapeXml="true" />]]></author>
```

```
<guid>http://http://127.0.0.1:8080/board/view/${item.boardIdx}</guid>
```

```
<pubDate><s:formatDate value="${item.regDate}" language="en" locale="US" pattern="EE, dd MMM yyyy HH:mm:ss Z"/></pubDate>
```

```
</item>
```

```
</c:forEach>
```


RSS 컨트롤러

```
@RequestMapping(  
    value = { "/rss" }, method = RequestMethod.GET)  
public String showRssList(Model model)  
{  
    List<Board> boardList = boardDao.getBoardList();  
    model.addAttribute("boardList", boardList);  
return viewBase + "/rss";  
}
```

비동기 처리

- @Async 어노테이션 메서드 위에 작성하면 됨
- 반환값은 **void**, `java.util.concurrent.Future`만 가능
- 사용하기 위한 환경 설정
 - `xmlns:task=`
"`http://www.springframework.org/schema/task`"
 - *`http://www.springframework.org/schema/task` `http://www.springframework.org/schema/task/spring-task.xsd`*
 - `<!-- Spring 3.0 @Task @Async -->`
 - `<task:annotation-driven />`

수고하셨습니다

- 사용자화 된 뷰를 만들어 보았습니다
- 커스텀 태그를 만들 수 있습니다
- RSS뷰를 만들 수 있습니다
- 비동기 처리에 대해 배워보았습니다
- 숙제 : 진행하시고 있는 프로젝트에 커스텀 태그를 한 곳 이상 사용해 주세요

다음 시간에는... (10주차)

- 프로젝트 명세서 발표
- 명세서를 발표하는 시간을 가지겠습니다
미완성이어도 좋습니다
- 페이징 뷰를 고민해 봅니다
- 기타 웹 개발에 대한 고민은 같이 해 봅시다
- 과제 : 뒷풀이 참석, 12월 31일까지 명세서 완성
작품은 2012년 1월까지 완성

참고자료

- 커스텀 태그
 - <http://www.4te.co.kr/566>
 - <http://bum20.springnote.com/pages/2445740>
 - <http://0thinktank.tistory.com/146>
- RSS
 - <http://en.wikipedia.org/wiki/RSS>
 - <http://www.rssboard.org/rss-specification>
 - <http://golmong.tistory.com/8>
- @Async
 - <http://whiteship.me/?tag=async>

Thank You!

WORLD. EVERYTHING. CONTAINED

Telephone 02 6677 4601 Fax 02 6677 4602

Address Second floor, 21-7, Songpa-dong, Songpa-gu, Seoul, Korea

©2007 **wecan** Corporation, All rights reserved