**HTML5模板引擎Thymeleaf笔记**

# 1简介

Thymeleaf提供了一个用于整合Spring MVC的可选模块，在应用开发中，你可以使用Thymeleaf来完全代替JSP，或其他模板引擎，如Velocity、FreeMarker等。Thymeleaf的主要目标在于提供一种可被浏览器正确显示的、格式良好的模板创建方式，因此也可以用作静态建模

# 2使用

## 2.1概述

1.第一步：在html页面加入头文件相应的schema



2.第二步：加入原始的HTML页面内容

3.第三步：采用ThyMeleaf标签替换页面显示内容，而不用改动原始HTML以及样式，thyMeleaf是一个基于标签自定义属性实现的一个模板引擎

## 2.2关键字

# 表示从messages bundle获取对象，也就是消息的资源本地化文件

$ 表示从model里面获取，相当于JSTL中的$

# $这2个可以一起用比如#{'system.'+${model.id}} -----这相当于 #{system.01}的资源本地化文件中的system.01内容

## 2.3常用标签

**1.文本内容标签：**

th:text="${data}",将data的值替换该属性所在标签的body。字符常量要用引号，比如th:text="'hello world'",th:text="2011+3",th:text="'my name is '+${user.name}"

注：th:utext，和th:text的区别是"unescaped text"。

**2.判断标签：**

th:if or th:unless，条件判断，支持布尔值，数字（非零为true)，字符，字符串等。

th:switch，th:case，选择语句。 th:case="\*"表示default case。

**3.公共内容引用标签**

th:fragment为片段标记，指定一个模板内一部分代码为一个片段，然后在其它的页面中用th:include进行包含

**4.循环数据：th:each举例：**

<trth:each="user : ${users}">

<td th:text="${user.id}">01</td>

<td th:text="${user.name}">朱遇平</td>

</tr>

<tr th:each="user,userStat:${users}">,userStat是状态变量，有 index,count,size,current,even,odd,first,last等属性，如果没有显示设置状态变量，thymeleaf会默 认给个“变量名+Stat"的状态变量。

注：标签内的静态数据不用管，当运行时候会自动替换

**5.资源地址：th:href举例：**

th:href="@{/users/delete(id=${u.id})}">括号中是传递的参数，链接会自动进行url-encoding的处理。@{...}内部可以是需要计算的表达式，比如：

th:href=”@{'/details/'+${user.login}(orderId=${o.id})}"

表单提交：

<form action="#" th:action="@{/users/add}" th:object="${myuser}" method="post">

这里th：Object表示表单与该myuser注入的实体映射，

在表单中th:field="\*{id}” 则表示该表单的值与myuser的id属性绑定， 等价于${#object.firstName},#object代表当前选择的对象。

**6.内联JS：**

<script th:inline="javascript">

/\*<![CDATA[\*/

var username = /\*[[${sesion.user.name}]]\*/ 'Sebastian';

/\*]]>\*/

</script>

## 2.4表达式功能对象

#dates：java.util.Date的功能方法类。

#calendars:类似#dates，面向java.util.Calendar

#numbers:格式化数字的功能方法类。

#strings：字符串对象的功能类，contains,startWiths,prepending/appending等等。

#objects:对objects的功能类操作。

#bools:对布尔值求值的功能方法。

#arrays：对数组的功能类方法。

#lists:对lists功能类方法

#sets

#maps

#aggregates:对数组或者集合创建聚合的功能方法，

th:text="${#aggregates.sum(o.orderLines.{purchasePrice \* amount})}"

#messages:在变量表达式中获取外部信息的功能类方法。

#ids：处理可能重复的id属性的功能类方法。