

Thymeleaf开发

Thymeleaf 是一个跟 Velocity、FreeMarker 类似的模板引擎，它可以完全替代 JSP。相较于其他的模板引擎，它有如下三个极吸引人的特点：

- Thymeleaf 在有网络和无网络的环境下皆可运行，即它可以让美工在浏览器查看页面的静态效果，也可以让程序员在服务器查看带数据的动态页面效果。这是由于它支持 html 原型，然后在 html 标签里增加额外的属性来达到模板+数据的展示方式。浏览器解释 html 时会忽略未定义的标签属性，所以 thymeleaf 的模板可以静态地运行；当有数据返回到页面时，Thymeleaf 标签会动态地替换掉静态内容，使页面动态显示。
- Thymeleaf 开箱即用的特性。它提供标准和spring标准两种方言，可以直接套用模板实现JSTL、OGNL表达式效果，避免每天套模板、改jstl、改标签的困扰。同时开发人员也可以扩展和创建自定义的方言。
- Thymeleaf 提供spring标准方言和一个与 SpringMVC 完美集成的可选模块，可以快速的实现表单绑定、属性编辑器、国际化等功能

Thymeleaf基础语法

- 变量表达式： `${.....}`

```
1 <input type="text" name="userName" value="James Carrot" th:value="${user.name}" />
```

引用user对象的name属性值。

- 选择/星号表达式 `*{.....}`

```
1 <div th:object="${session.user}">
2   <p>Nationality: <span th:text="*{nationality}">Saturn</span></p>
3 </div>
```

选择表达式一般跟在th:object后，直接取object中的属性。

- URL表达式 `@{.....}`

```
1 <a href="details.html" th:href="@{/order/details(orderId=${o.id})}">view</a>
2
3 @{.....}支持决定路径和相对路径。其中相对路径又支持跨上下文调用url和协议的引用
  ( //code.jquery.com/jquery-2.0.3.min.js )。
4
5 当URL为后台传出的参数时，代码 
```

常用的th标签

简单数据转换（数字，日期）

```

1 <dt>价格</dt>
2 <dd th:text="${#numbers.formatDecimal(product.price, 1, 2)}">180</dd>
3 <dt>进货日期</dt>
4 <dd th:text="${#dates.format(product.availableFrom, 'yyyy-MM-dd')}">2014-12-01</dd>

```

字符串拼接

```

1 <dd th:text="${'$'+product.price}">235</dd>

```

表单应用

```

1 <form th:action="@{/bb}" th:object="${user}" method="post" th:method="post">
2     <input type="text" th:field="*{name}"/>
3     <input type="text" th:field="*{msg}"/>
4     <input type="submit"/>
5 </form>

```

显示页面的数据迭代

```

1 <tbody th:remove="all-but-first">
2 <tr th:each="product:${productList}">
3     <td th:text="${productStat.count}">1</td>
4     <td th:text="${product.description}">Red Chair</td>
5     <td th:text="${'$' + #numbers.formatDecimal(product.price, 1, 2)}">$123</td>
6     <td th:text="${#dates.format(product.availableFrom, 'yyyy-MM-dd')}">2014-12-01</td>
7 </tr>
8 <tr>
9     <td>White table</td>
10    <td>$200</td>
11    <td>15-Jul-2013</td>
12 </tr>
13 </tbody>

```

条件判断

```

1 <span th:if="${product.price lt 100}" class="offer">Special offer!</span>
2     不能用"<", ">"等符号, 要用"lt"等替代
3 <!-- 当gender存在时, 选择对应的选项; 若gender不存在或为null时, 取得customer对象的名字 -->
4 <td th:switch="${customer.gender?.name()}">
5     
6     
8     <span th:case="*">Unknown</span>
9 </td>

```

公共页面包含

```
1 <head th:include="header::headername" th:with="title='主页标题'">
2     <meta charset="UTF-8">
3     <title>Title</title>
4 </head>
```

```
1 <!DOCTYPE html>
2 <html lang="cn" xmlns:th="http://www.thymeleaf.org">
3 <head th:fragment="headername">
4     <meta charset="UTF-8">
5     <title th:text="${title}!=null?${title}:'layout'">header的标题</title>
6 </head>
7 <body>
8 </body>
9 </html>
```

js处理

```
1 <script th:inline="javascript">
2     var name="hello,"+[[${name}]];
3     alert(name);
4 </script>
```

H5基础

什么是盒子模型

在网页中，一个元素占有空间的大小由几个部分构成，其中包括元素的内容（content），元素的内边距（padding），元素的边框（border），元素的外边距（margin）四个部分。这四个部分占有的空间中，有的部分可以显示相应内容，而有的部分只用来分隔相邻的区域或区域。4个部分一起构成了css中元素的盒模型

简述同步和异步的区别

同步是阻塞模式，异步是非阻塞模式。同步就是指一个进程在执行某个请求的时候，若该请求需要一段时间才能返回信息，那么这个进程将会一直等待下去，直到收到返回信息才继续执行下去；异步是指进程不需要一直等下去，而是继续执行下面的操作，不管其他进程的状态。当有消息返回时系统会通知进程进行处理，这样可以提高执行的效率。

浏览器的内核分别是什么

IE: trident内核

Firefox : gecko内核

Safari : webkit内核

Opera : 以前是presto内核，Opera现已改用Google Chrome的Blink内核

Chrome : Blink(基于webkit，Google与Opera Software共同开发)

JQuery应用开发

什么是jQuery ?

- jQuery是一个快速，轻量且功能丰富的客户端JavaScript库/框架，
- 它有助于遍历HTML DOM，制作动画，添加Ajax交互，操纵页面内容，更改样式并提供很酷的UI效果。
- 它是最受欢迎的客户端库之一，根据调查，它在每二个网站上就会有一个使用

为什么使用jQuery ?

- 易于使用和学习。
- 易于扩展。
- 跨浏览器支持 (IE 6.0 + , FF 1.5 + , Safari 2.0 + , Opera 9.0+)
- 易于用于DOM操作和遍历。
- 大量内置方法。
- AJAX功能支持。
- 更改或应用CSS，创建动画的方法。
- 事件检测和处理。
- 满足各种需求的大量插件

\$(this) 和 this 关键字在 jQuery 中有何不同

- \$(this) 返回一个 jQuery 对象，你可以对它调用多个 jQuery 方法，比如用 text() 获取文本，用val() 获取值等等
- this代表当前元素，它是 JavaScript 关键词中的一个，表示上下文中的当前 DOM 元素。你不能对它调用 jQuery 方法，直到它被 (this)。

JQuery有几种选择器

- (1)、基本选择器：#id , class,element,*;
- (2)、层次选择器：parent > child , prev + next , prev ~ siblings
- (3)、基本过滤器选择器：:first , :last , :not , :even , :odd , :eq , :gt , :lt
- (4)、内容过滤器选择器：:contains , :empty , :has , :parent
- (5)、可见性过滤器选择器：:hidden , :visible
- (6)、属性过滤器选择器：[attribute] , [attribute=value] , [attribute!=value] , [attribute^=value] , [attribute\$=value] , [attribute*=value]
- (7)、子元素过滤器选择器：:nth-child , :first-child , :last-child , :only-child
- (8)、表单选择器：:input , :text , :password , :radio , :checkbox , :submit 等；
- (9)、表单过滤器选择器：:enabled , :disabled , :checked , :selected

\$(document).ready()方法和window.onload有什么区别

- window.onload方法是在网页中所有的元素(包括元素的所有关联文件)完全加载到浏览器后才执行的
- \$(document).ready() 方法可以在DOM载入就绪时就对其进行操纵，并调用执行绑定的函数。

Ajax的优缺点及工作原理

AJAX = Asynchronous JavaScript and XML (异步的 JavaScript 和 XML)。Ajax 是一种用于创建快速动态网页的技术。Ajax 是一种在无需重新加载整个网页的情况下，能够更新部分网页的技术。

传统的网页（不使用 Ajax）如果需要更新内容，必须重载整个网页页面。

Ajax优点：

- 减轻服务器的负担,按需取数据,最大程度的减少冗余请求
- 局部刷新页面,减少用户心理和实际的等待时间,带来更好的用户体验
- 基于xml标准化,并被广泛支持,不需安装插件等,进一步促进页面和数据的分离

Ajax缺点：

- AJAX大量的使用了javascript和ajax引擎,这些取决于浏览器的支持.在编写的时候考虑对浏览器的兼容性.
- AJAX只是局部刷新,所以页面的后退按钮是没有用的.
- 对流媒体还有移动设备的支持不是太好

AJAX的工作原理

- 创建ajax对象 (XMLHttpRequest / ActiveXObject(Microsoft.XMLHttp))

```
var xhr = new XMLHttpRequest();
```

- 判断数据传输方式(GET/POST)

```
以GET为例： xhr.onload = function() { console.log("READYSTATE:", xhr.readyState);  
console.log(this.responseText); }
```

- 打开链接 open()

```
xhr.open("GET"," user.txt", true);
```

- 发送 send()

```
xhr.send();
```

当ajax对象完成第四步onreadystatechange数据接收完成，判断http响应状态status200-300之间或者304缓存执行回调函数

jquery中.get()提交和.post()提交有区别吗

相同点：都是异步请求的方式来获取服务端的数据；异同点：

- 1、请求方式不同：.post() 方法使用POST方法来进行异步请求的。
- 2、参数传递方式不同：get请求会将参数跟在URL后进行传递，而POST请求则是作为HTTP消息的实体内容发送给Web服务器的，这种传递是对用户不可见的。
- 3、数据传输大小不同：get方式传输的数据大小不能超过2KB 而POST要大的多
- 4、安全问题：GET 方式请求的数据会被浏览器缓存起来，因此有安全问题。

写出一个简单的\$.ajax()的请求方式

```
1 $.ajax({
2     url:'http://www.baidu.com',
3     type:'POST',
4     data:data,
5     cache:true,
6     headers:{},
7     beforeSend: function() {},
8     success: function() {},
9     error: function() {},
10    complete: function() {}
11 });
```

什么是跨域

由于浏览器同源策略，凡是发送请求url的协议、域名、端口三者之间任意一与当前页面地址不同即为跨域。存在跨域的情况：

- 网络协议不同，如http协议访问https协议。
- 端口不同，如80端口访问8080端口。
- 域名不同，如qianduanblog.com访问baidu.com。
- 子域名不同，如abc.qianduanblog.com访问def.qianduanblog.com。
- 域名和域名对应ip,如www.a.com访问20.205.28.90。

跨域请求资源的方法：

(1)、proxy代理

定义和用法：proxy代理用于将请求发送给后台服务器，通过服务器来发送请求，然后将请求的结果传递给前端。

实现方法：通过nginx代理；

注意点：1、如果代理的是https协议的请求，那么proxy首先需要信任该证书（尤其是自定义证书）或者忽略证书检查，否则请求无法成功。

(2)、CORS 【Cross-Origin Resource Sharing】

定义和用法：是现代浏览器支持跨域资源请求的一种最常用的方式。

使用方法：一般需要后端人员在处理请求数据的时候，添加允许跨域的相关操作。如下：

```
1 res.writeHead(200, {
2     "Content-Type": "text/html; charset=UTF-8",
3     "Access-Control-Allow-Origin": 'http://localhost',
4     'Access-Control-Allow-Methods': 'GET, POST, OPTIONS',
5     'Access-Control-Allow-Headers': 'X-Requested-With, Content-Type'
6 });
```

(3)、jsonp

定义和用法：通过动态插入一个script标签。浏览器对script的资源引用没有同源限制，同时资源加载到页面后会立即执行（没有阻塞的情况下）。

特点：通过情况下，通过动态创建script来读取他域的动态资源，获取的数据一般为json格式。

实例如下：

```
1 <script>
2     function testjsonp(data) {
3         console.log(data.name); // 获取返回的结果
4     }
5 </script>
6 <script>
7     var _script = document.createElement('script');
8     _script.type = "text/javascript";
9     _script.src = "http://localhost:8888/jsonp?callback=testjsonp";
10    document.head.appendChild(_script);
11 </script>
```

缺点：

- 1、这种方式无法发送post请求（这里）
- 2、另外要确定jsonp的请求是否失败并不容易，大多数框架的实现都是结合超时时间来判定。

你所了解到的Web攻击技术

- XSS (Cross-Site Scripting , 跨站脚本攻击)：指通过存在安全漏洞的Web网站注册用户的浏览器内运行非法的HTML标签或者JavaScript进行的一种攻击。
- SQL注入攻击
- CSRF (Cross-Site Request Forgeries , 跨站点请求伪造)：指攻击者通过设置好的陷阱，强制对已完成的认证用户进行非预期的个人信息或设定信息等某些状态更新。

web前端开发，如何提高页面性能优化

- 内容方面：
 - 减少 HTTP 请求 (Make Fewer HTTP Requests)
 - 减少 DOM 元素数量 (Reduce the Number of DOM Elements)
 - 使得 Ajax 可缓存 (Make Ajax Cacheable)
- 针对CSS：
 - 把 CSS 放到代码页上端 (Put Stylesheets at the Top)
 - 从页面中剥离 JavaScript 与 CSS (Make JavaScript and CSS External)
 - 精简 JavaScript 与 CSS (Minify JavaScript and CSS)
 - 避免 CSS 表达式 (Avoid CSS Expressions)
- 针对JavaScript：
 - 脚本放到 HTML 代码页底部 (Put Scripts at the Bottom)
 - 从页面中剥离 JavaScript 与 CSS (Make JavaScript and CSS External)
 - 精简 JavaScript 与 CSS (Minify JavaScript and CSS)
 - 移除重复脚本 (Remove Duplicate Scripts)
- 面向图片(Image)：
 - 优化图片
 - 不要在 HTML 中使用缩放图片

- 使用恰当的图片格式
- 使用 CSS Sprites 技巧对图片优化

浏览器是如何渲染页面的

1. 解析HTML文件，创建DOM树。自上而下，遇到任何样式（link、style）与脚本（script）都会阻塞（外部样式不阻塞后续外部脚本的加载）。
2. 解析CSS。优先级：浏览器默认设置<用户设置<外部样式<内联样式<HTML中的style样式；
3. 将CSS与DOM合并，构建渲染树（Render Tree）
4. 布局和绘制，重绘（repaint）和重排（reflow）

LayUI应用开发

layui是一款采用自身模块规范编写的前端 UI 框架，遵循原生 HTML/CSS/JS 的书写与组织形式，门槛极低，拿来即用。其外在极简，体积轻盈，组件丰盈，从核心代码到 API 的每一处细节都经过精心雕琢，非常适合界面的快速开发。

是为后端程序员设计的前端框架。这是一个封装了各种css和js、Ajax等等的前端框架，其封装程度之高，有时甚至对程序员来说不大友好。但对于前端技术一般的人来说，layui不失为一个好的工具。

建议参考文档：https://blog.csdn.net/qq_28379809/article/details/80098653