

HTML超文本标记语言

有俩部分：head和body

body中的标签有很多，我们常见的标签是table（表格）和form（表单）

JQUERY是一个快捷简洁的JavaScript框架，它封装JavaScript常用的功能代码，提供一种简便的JavaScript设计模式，优化HTML文档操作，事件处理，设计动画和ajax交互

jQuery的使用：需要下载jQuery文件，解压后将js文件复制到工程中，引入需要使用jQuery的页面中即可

一，*jQuery*

jQuery的常用操作：

1， 页面加载事件：jQuery中的页面加载事件可以共存

格式：`$ (function () { //代码 })`；

优先执行js中的页面加载事件完成后，才会执行jQuery中的事件加载

2， jQuery的基本选择器

2.1： id选择器：`$ (“#id的值”)`

2.2： 类选择器：`$ (“.class的值”)`；

2.3： 元素选择器：`$ (“HTML的标签名”)`；

3， 关于jQuery对象的常用方法：

3.1： attr和prop方法：当仅传递标签的属性名时，就是获取属性值，如果既传属性名和属性值，就是赋值操作

3.2： 操作form表单中表单项的value属性值，有特殊的方法：val（）方法，空参数表示取值，传参数代表赋值

3.3： A.append(B)表示将B对象追加到A对象内部

二，*jsp*：

java server pages: java服务器页面，根本是一个简化的servlet设计，后缀名为.jsp，实现了HTML语法中java扩展，与servlet一样，在服务器端执行，通常返回给客户端的就是一个HTML文本，因此客户端只要有服务器就能访问该页面

注意：使用jQuery选择器获取的对象都是jQuery的对象，只能调用jQuery中规定的方法和属性，使用原生的js代码获取的对象都是js对象，只能调用js中的方法和属性，可以通过\$(js对象)的方式将js对象转换为jQuery对象，反之，可以使用jQuery对象[0]，js对象和jQuery的对象不一样

是一个简化的servlet

一个jsp页面可以分为以下几个部分：

- a, 静态数据，如HTML
- b, jsp指令，如include指令（静态包含）
- c, jsp脚本元素和变量
- d, jsp动作
- e, 用户自定义标签

jsp 的9大内置对象：

- 1, request: HttpServletRequest类的实例
- 2, response: HttpServletResponse类的实例
- 3, out: JspWriter类的实例，用于把结果输出到网页上
- 4, session: HttpSession类的实例
- 5, application: ServletContext类的实例，与应用上下文有关
- 6, config: ServletConfig类的实例
- 7, pageContext: PageContext类的实例，提供对jsp页面所有对象以及命名空间的访问
- 8, page: 类似于java类中的this关键字
- 9, exception: Throwable类的对象，代表发生错误的jsp页面中对应的异常对象

jsp的4个域对象：（就是存值和取值）

- 1, page域: pageContext对象：仅仅在当前页面内可以使用，
- 2, request域: request对象，请求范围内有效
- 3, session域: session对象，一个会话范围内可以使用，多次请求范围可以使用
- 4, application域: application对象，整个工程范围内有效

三, EL表达式:

JSP表达式语言（EL）使得访问存储在javabean中的数据变得非常简单，jsp EL既可以用来创建算术表达式也可以用来创建逻辑表达式，在jsp EL表达式内可以是整型，浮点型，字符串，常量true，false还有null

`$(expr)` 其中，`expr`指的是表达式，在jsp EL中的操作符`.`和`{ }`，这两个操作符通过内嵌的jsp对象访问各种各样的javabean属性

EL操作符：`.`（访问一个bean属性或一个映射条目） `[]` (特殊的属性名)

EL表达式的隐式对象：在`<%` 只能使用jsp的隐式对象`>`，在`#{` 只能使用EL表达式的隐式对象`}`

含有11个隐式对象

- 1, `pagescope`: page作用域
- 2, `requestscope`: request作用域
- 3, `sessionscope`: session作用域
- 4, `applicationscope`: application作用域
- 5, `param`: request对象的参数，字符串
- 6, `paramvalues`: request对象的参数，字符串集合
- 7, `header`: HTTP信息头，字符串
- 8, `headervalues`: HTTP信息头，字符串集合
- 9, `initparam`: 上下文初始化参数
- 10, `cookie`: cookie值
- 11, `pagecontext`: 当前页面的pagecontext，可以在EL表达式内部获取其他8个内置对象

更改jsp页面的编码格式：

<https://jingyan.baidu.com/article/e2284b2b3f4089e2e7118d57.html>

四, *jstl*

jsp标准库jstl是一个jsp标签集合，它封装了jsp应用的通用核心功能，jstl支持通用的，结构化的任务，比如迭代，条件判断，xml文档操作，国际化标签，sql标签，除了这些，它还提供了一个框架来使用集成jstl的自定义标签

要使用jstl需要引入jstl才能使用

下载地址：

官网: <http://archive.apache.org/dist/jakarta/taglibs/standard/binaries/>

菜鸟教程网站下载地址: <http://static.runoob.com/download/jakarta-taglibs-standard-1.1.2.tar.gz>

使用步骤:

- 1, 下载Jakarta-taglibs-standard-1.1.2.zip包并解压
- 2, 将Jakarta-taglibs-standard-1.1.2/lib下的两个jar文件: standard.jar和jstl.jar文件拷贝到/WEB-INF/lib/下
- 3, 在需要使用jstl标签的jsp页面中引入标准标签库即可使用jstl标准标签

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

JSTL 库安装

Apache Tomcat安装JSTL 库步骤如下:

从Apache的标准标签库中下载的二进包(jakarta-taglibs-standard-current.zip).

- 官方下载地址: <http://archive.apache.org/dist/jakarta/taglibs/standard/binaries/>
- 本站下载地址: jakarta-taglibs-standard-1.1.2.zip

下载jakarta-taglibs-standard-1.1.2.zip 包并解压, 将jakarta-taglibs-standard-1.1.2/lib/下的两个jar文件: standard.jar和jstl.jar文件拷贝到/WEB-INF/lib/下

接下来我们在 web.xml 文件中添加以下配置:

功能分为5个类别:

核心标签:

格式化标签:

sql标签:

sml标签:

jstl函数:

jstl 常用标签:

- 1, <c:if>标签: 判断表达式的值, 如果表达式的值为true, 则执行其主体内容

语法格式:

```
<c:if test = "<boolean>" var = "<string>" scope = "<string>">
...
</c:if>
```

其中: test表示条件, var用于存储条件结果的变量, scope表示var属性的作用域

- 2, foreach:

<c:forEach><c:Tokens>标签, 这些标签封装了java中的for, while, do-while循环, 相对而言<c:forEach>标签是更加通用的标签, 因为它迭代一个集合中的对象

ajax是一种在无需重新加载整个页面的情况下，能够更新部分页面技术