# 尚马教育 JAVA WEB课程

# Html

文档编号：B01

创建日期： 2017-06-13

最后修改日期：2019-09-23

版 本 号：V3.0

电子版文件名：尚马教育-第二阶段-1.html.docx

**文档修改记录：**

|  |  |  |  |
| --- | --- | --- | --- |
| 更新日期 | 更新作者 | 更新说明 | 版本号 |
| 2017-07-30 | 张元林 | 初始版本 | V1.0 |
| 2018-08-01 | 王绍成 | Html内容更新 | V2.0 |
| 2019-09-13 | 郭雪岩 | Html内容更新 | V3.0 |

**主讲人：**

**郭雪岩**

目录

[1. WEB技术介绍](#_Toc11927_WPSOffice_Level1) [3](#_Toc11927_WPSOffice_Level1)

[1.1. 认识web技术](#_Toc15421_WPSOffice_Level2) [3](#_Toc15421_WPSOffice_Level2)

[1.2. 认识html](#_Toc20014_WPSOffice_Level2) [3](#_Toc20014_WPSOffice_Level2)

[1.3. Html基本结构](#_Toc28581_WPSOffice_Level2) [4](#_Toc28581_WPSOffice_Level2)

[2. Html详解](#_Toc15421_WPSOffice_Level1) [5](#_Toc15421_WPSOffice_Level1)

[2.1. Head标签](#_Toc8273_WPSOffice_Level2) [5](#_Toc8273_WPSOffice_Level2)

[2.2. html中特殊编写方式：](#_Toc15941_WPSOffice_Level2) [5](#_Toc15941_WPSOffice_Level2)

[2.3. Body标签](#_Toc18156_WPSOffice_Level2) [6](#_Toc18156_WPSOffice_Level2)

[ 标签的闭合特性](#_Toc15421_WPSOffice_Level3) [6](#_Toc15421_WPSOffice_Level3)

[ 标签的属性描述](#_Toc20014_WPSOffice_Level3) [7](#_Toc20014_WPSOffice_Level3)

[ 属性的声明语法](#_Toc28581_WPSOffice_Level3) [7](#_Toc28581_WPSOffice_Level3)

[ 标签的公用属性：所有标签都具备的属性。](#_Toc8273_WPSOffice_Level3) [7](#_Toc8273_WPSOffice_Level3)

[ Body子标签功能分类](#_Toc15941_WPSOffice_Level3) [7](#_Toc15941_WPSOffice_Level3)

[2.3.1. 布局标签](#_Toc18156_WPSOffice_Level3) [8](#_Toc18156_WPSOffice_Level3)

[2.3.2. 文本标签](#_Toc28193_WPSOffice_Level3) [11](#_Toc28193_WPSOffice_Level3)

[2.3.3. 超链接标签](#_Toc15094_WPSOffice_Level3) [11](#_Toc15094_WPSOffice_Level3)

[2.3.4. 图片标签img](#_Toc15174_WPSOffice_Level3) [14](#_Toc15174_WPSOffice_Level3)

[2.3.5. 表格标签](#_Toc15194_WPSOffice_Level3) [14](#_Toc15194_WPSOffice_Level3)

[2.3.6. 表单标签](#_Toc22467_WPSOffice_Level3) [16](#_Toc22467_WPSOffice_Level3)

[2.4. 页面嵌套](#_Toc28193_WPSOffice_Level2) [19](#_Toc28193_WPSOffice_Level2)

[2.4.1. Frameset frame](#_Toc3469_WPSOffice_Level3) [19](#_Toc3469_WPSOffice_Level3)

[2.4.2. Iframe标签](#_Toc26701_WPSOffice_Level3) [20](#_Toc26701_WPSOffice_Level3)

## WEB技术介绍

### 认识web技术

* HTML ：超文本标记语言 (Hyper Text Markup Language)，用于描述网页中存在哪些网页素。
* css ：层叠样式表(Cascading Style Sheets)是一种用来表现HTML元素显示方式的语言，包含元素的美观修饰、位置定位等。
* javascript：通俗的叫法是js，浏览器可以解析并执行的语言，可以控制网页上元素，从而形成各种动态效果。

注意：由浏览器读取 并展示效果

html 决定有哪些元素

css 给元素进行修饰 有漂亮的外观

javascript（js） 给元素加上动态效果（动作）

### 认识html

HTML 超文本标记语言 (Hyper Text Markup Language)，在浏览器中配合CSS样式表显示网

超文本

指页面内可以包含图片、链接，甚至音乐、程序等非文字元素

标记语言

描述html的语法格式，采用标签的方式描述网页内的元素

### Html基本结构

|  |
| --- |
| <html>  <head>  </head>  <body>  hello html！  </body>  </html> |

使用文本编辑器编辑 通过浏览器读取运行

## Html详解

### Head标签

* 头标签head：头标签包含当前网页的元数据描述，如搜索关键字、编码、引入外部文等，头标签内的内容不会出现在网页显示区域。
* 头标签包含以下内容
  + title标签：用来声明网页的标题
  + meta标签：HTML网页文档的属性，例如作者、日期、网页描述、关键词等。
* 其他还有一些后边会用到
  + style标签：用来内嵌css样式
  + link标签：用来引入css样式文件
  + script标签：用来引入javascript程序。
* Meta标签（元数据标签）主要用于对页面的配置 主要会使用charset属性设置字符编码来解决乱码问题

|  |
| --- |
| <head>  <meta charset="utf-8">  </head> |

### html中特殊编写方式：

* Html注释

<!-- 注释内容 -->

注意：html的注释 没有显示效果 但是用户在浏览器source中可以看到

* Html转义符（特殊字符）

特殊字符：需要使用其他字符代替的字符，主要有如下两种使用场景

使用频率较少且难以直接输入的符号，比如数学符号；单位符号；制表符等。

需要输入的符号是html的关键字符号如单引号、双引号、尖括号等。

|  |
| --- |
|  |

### Body标签

* Body标签：网页内容的容器，Body标签的子标签用来渲染网页显示，所有标签都是忽略大小写的，但推荐小写。
* 标签的闭合特性
* 闭合标签(双标签)：由开始标签和结束标签组成。开始标签是被括号包围的元素名， 结束标签是被括号包围的斜杠和元素名。

例如：<p></p><a></a>

闭合标签经常用作为容器使用 需要在内部包含（标签嵌套）其他标签

例如：<p><a></a></p>

* 非闭合标签(单标签)：是被括号包围的元素名和斜杠。

非闭合标签一般都具有特殊的显示效果 不在内部包含其他标签

例如：<input/><img />

注意：html中有大量双标签 所以这种包含结构类似树状结构

* 标签的属性描述

每个标签都具备一组公用或当前标签独有的属性，属性的作用是描述标签的外观行为或识别当前标签的标识。

* 属性的声明语法

<标签名 属性名 = “属性值“ 属性名 = “属性值“ >xxx </标签名>

属性可以用来设置标签的一些特性 每个标签有一些公用的属性也有不同的特性属性

* 标签的公用属性：所有标签都具备的属性。
* class：规定当前元素所引用的样式类名称。
* id：规定当前元素在网页上的唯一标识，不允许重复，一般在css、js中会根据id检索当前元素。
* name：当前元素在文档上的名称。
* style：规定当前元素的显示样式，取值为css样式表。
* Body子标签功能分类
* 布局标签：控制子元素所在位置，并不独立显示。
* 文本标签：用来显示文本的标签。
* 框架标签：在网页内嵌入其他子网页的标签。
* 超链接标签：超链接标签。
* 图片以及图像映射：用来显示图片、图像的标签。
* 表格标签：用来按照一定长度、宽度显示的表格标签。
* 表单类标签：用来控制用户输入的一组标签。

#### 布局标签

* 标题标签

|  |
| --- |
| 1. <BODY> 2. <H1>一级标题</H1> 3. <H2>二级标题</H2> 4. <H3>三级标题</H3> 5. <H4>四级标题</H4> 6. <H5>五级标题</H5> 7. <H6>六级标题</H6> 8. </BODY> |

<H#> ... </H#> #=1, 2, 3, 4, 5, 6

说明：<H1>到<H6>字体大小显示效果依次递减

* 段落标签：

|  |
| --- |
| 1. <body> 2. <p>这是一个段落</p> 3. </body> |

段落（paragraph）的缩写，包裹的内容为一个段落的文字，默认自带内边距。

* div标签：一般配合css对网页内大块区域进行布局。

|  |
| --- |
| 1. <body> 2. <div>这是一个div 空白块元素</div> 3. </body> |

* span标签：一般用来包裹非特殊显示文本进行

|  |
| --- |
| 1. <body> 2. <span>这是一个span 空白行内元素</span> 3. </body> |

* 列表标签：是这一组样式相同，功能相同的元素的集合，列表不同于表格（多行多列）。

|  |
| --- |
|  |

* 列表标签的分类
* 自定义列表
* 有序列表
* 无序列表
* 自定义列表：用来表示一组无序的、具备标题的列表数据。

|  |
| --- |
| <dl>  <dt>标题1</dt>  <dd>这里是一段内容</dd>  <dt>标题1</dt>  <dd>这里是一段内容</dd>  </dl>  列表容器标签  列表标题  列表内容 |

* 有序列表：用来显示有顺序的列表数据信息，可显示顺序号。

|  |
| --- |
| <ol type="a">  <li>用户管理</li>  <li>部门管理</li>  <li>岗位管理</li>  </ol>  列表容器  列表明细，块级  序号类型 |

* 无序列表：用来显示无顺序的列表数据信息，可以设置列表符号。

|  |
| --- |
| <ul type="circle">  <li>用户管理</li>  <li>部门管理</li>  <li>岗位管理</li>  </ul>  列表容器  列表明细，块级  明细符号 |

#### 文本标签

* 常用文本标签
* u标签：内容默认以下划线方式显示。

<u>下划线</u>

* strong标签：内容默认以强调文字方式显示。

<strong>粗体</strong>

* em标签：内容默认以一般强调文字显示。

<em>斜体</em>

#### 超链接标签

* A标签：又称为超链接标签，通过a标签可以访问（链接）到其他网页。
* 基本语法：<a href="http://www.baidu.com">百度一下</a>
* 显示效果：

|  |
| --- |
| 按照是否被访问过显示链接 |

* href的取值可以为如下两种类型。
* 绝对路径：完整的网页访问路径，与当前网页位置无关

|  |
| --- |
| <a href="http://192.168.20.141:8080/myproj/test.html">我的页面</a>  https:// XXXX : port /XX/XX.html  协议名 地址 端口 资源路径 |

* 相对路径：需要访问的网页资源与当前网页资源路径的相对位置。

|  |
| --- |
| "."--代表目前所在的目录。  <a href="../chapter5/iframe.html">  使用相对路径访问本域内的网页资源  </a>  ".."--代表上一层目录。  "/"--代表根目录。 |

* 锚点：超链接的另一种用法，又叫命名锚记。在网页中的作用是快速定位器。

|  |
| --- |
| 点击后迅速定位到网页指定位置。 |

* 锚点的声明

|  |
| --- |
| <a href="#jump">点我看看</a>  ---<p id="jump">我是该区域的相应内容</p>  ---<p name="jump">我是该区域的相应内容</p> |

使用id与name声明锚点的区别

在老版本浏览器中最初都是使用name作为锚点位置的声明方式。

网页中name属性的取值不唯一，因此造成锚点定位不准，所以后期浏览器都支持使用id作为锚点声明位置。

Html5规范中已经声明取消name的锚点声明方式，建议使用id。

超链接中经常嵌套图片标签<img>

此时点击该图片也可以有跳转效果 让链接变的更美观（各种广告）

* 超链接的打开方式：被访问的网页以哪种方式打开，使用target属性设置。

|  |
| --- |
| <a href="xxx" target=”xxx”>点我看看</a> |

target属性的取值范围

* \_blank:在新窗口中打开链接文档
* \_self:默认，在当前浏览器窗口中打开被链接文档。
* \_parent:在父框架中打开被链接文档。
* \_top:在当前网页的最顶层框架中打开被链接文档。
* framename：在指定的框架中打开被链接文档。

#### 图片标签img

<img src="coder.jpg" width="100px" height="100px" alt="图片"/>

|  |
| --- |
| Src 图片路径  Width 图片宽度  Height 图片高度  Alt 图片说明 |

* src与href的区别：实际效果类似 内部处理方式不同 主要区别如下
* src（ source ）的值是外部资源的访问路径，在请求src资源时会将其指向的资源下载并应用到当前文档中，此时外部资源作为当前文档的一部分（引入），一般用作非文本引入方式。
* href 表示超文本引用（hypertext reference），在使用href请求外部资源时，会下载外部资源，同时当前网页读取外部资源的内容（引用）。一般用作文本引入方式

注意：Link、a标签使用href，其余使用src

#### 表格标签

* 常见标签

Table 表格

Tr 行

Td 单元格（列）

Th 列头

Thead 表区域（头）

Tbody 表区域（体）

Tfoot 表区域（脚）

Caption 表名

表格标签主要用于直观展示数据库中的数据 与数据库数据格式类似

|  |
| --- |
| 图片包含 屏幕截图, 室内, 天空  已生成极高可信度的说明 |

* 表格属性设置

|  |
| --- |
| 1. **border(边框的厚度)**   **内容**  **内容**  **内容**  **内容**  **cellpadding**   1. **(单元格填充)** 2. **cellspacing** 3. **(单元格间距)** |

* 跨行跨列

Rowspan 跨行

Cellspan 跨列

#### 表单标签

* 表单的典型应用
* 注册用户
* 收集信息
* 反馈信息
* 为网站提供搜索

|  |
| --- |
| **注册**  **收集信息** |

* 常用标签

|  |
| --- |
| * Input标签   <input type=”xxxx”/> 常用表单元素  type的属性值：   * Text 文本框 * Password 密码框 * Radio 单选按钮 * Checkbox 多选按钮 * File 文件框 * Color 颜色框 * Date 日期框 * Button 空白按钮 * Submit 提交按钮 * Reset 重置按钮 * Select标签 下拉菜单   <select>  <option>选项1</option>  <option>选项1</option>  </select>   * Textarea标签 多行文本框   <textarea></textarea>   * Form表单（用户填写的数据需要通过表单标签提交）   <form action=“http://www.sohu.com” method=“post”> </form>  Action属性 代表提交的地址  Method属性 代表提交的方式get post |

|  |  |  |
| --- | --- | --- |
|  | **GET** | **POST** |
| 后退按钮/刷新 | 无害 | 数据会被重新提交（浏览器应该告知用户数据会被重新提交）。 |
| 书签 | 可收藏为书签 | 不可收藏为书签 |
| 缓存 | 能被缓存 | 不能缓存 |
| 编码类型 | application/x-www-form-urlencoded | application/x-www-form-urlencoded 或 multipart/form-data。为二进制数据使用多重编码。 |
| 历史 | 参数保留在浏览器历史中。 | 参数不会保存在浏览器历史中。 |
| 对数据长度的限制 | 是的。当发送数据时，GET 方法向 URL 添加数据；URL 的长度是受限制的（URL 的最大长度是 2048 个字符）。 | 无限制。 |
| 对数据类型的限制 | 只允许 ASCII 字符。 | 没有限制。也允许二进制数据。 |
| 安全性 | 与 POST 相比，GET 的安全性较差，因为所发送的数据是 URL 的一部分。  在发送密码或其他敏感信息时绝不要使用 GET ！ | POST 比 GET 更安全，因为参数不会被保存在浏览器历史或 web 服务器日志中。 |
| 可见性 | 数据在 URL 中对所有人都是可见的。 | 数据不会显示在 URL 中。 |

### 页面嵌套

Html页面可以通过一些方式把多个页面组合成一个页面

#### Frameset frame

|  |
| --- |
|  |

通过frameset 可以把页面拆分成多份 通过frame在每一份引入一个单独的页面

|  |
| --- |
| **<FRAMESET rows="20%,\*" frameborder="0">**  **<FRAME src="top.html" name="topframe“ scrolling="no" noresize="noresize">**  **<FRAMESET cols="20%,\*">**  **<FRAME src="left.html" noresize="noresize“ scrolling="no" name="leftframe“ >**  **<FRAME src="right.html" name="rightframe">**  **</FRAMESET>**  **</FRAMESET>**  **设置无边框**  **不显示滚动条**  **禁止调整框架大小**  **框架名称，便于超文本链接锚标签target属性所引用** |

* 可以通过超链接的target属性 指定到跳转哪个frame

<frame src=”xxxx” name=“窗口名”>

<a href=”xxx” target=“窗口名”>

#### Iframe标签

#### 在页面中直接嵌入页面 用法与frame类似

<iframe src="xxx"></iframe>