

**HTML**

**1.学习线路**

|  |  |  |  |
| --- | --- | --- | --- |
| **HTML5基础** | **CSS3基础** | **H5C3加强** | **项目开发** |
| 任务：常见HTML5 标签 | 任务：CSS3样式，完成传统PC端的网 页渲染 | 任务：HTML5+CSS3新增标签 和样式 | 综合利用HTML+CSS进行项目 开发 |

**2.HTML简介**

本章学习线路：

网页的基本组成

什么是HTML

常用的浏览器

Web标准的三大组成部分

**2.1.什么是网页**

**网页**指在Internet根据一定规则，使用HTML制作的用于展示特定内容相关的网页集合。

**网页**是网站中的一页，通常是**HTML格式的文件**，它要通过浏览器来进行阅读。

**网页是构成网站的基本元素**，它通常由图片，超链接，文字，声音，视频等元素组成。通常我们看到的网页以**.htm**和**.html**后缀结尾的文

件，因此将其俗称为**HTML文件**。

**2.2.什么是HTML**

**HTML**是指**超文本标记语言**(**H**yper **T**ext **M**arkup **L**anguage)，它是用来描述网页的一种语言。

**HTML**不是编译型的语言，而是一种**标记语言** (markup language)。

标记语言是一套标记标签 (markup tag)。

**所谓超文本，有两种含义：**

它可以加入图片、声音、动画、多媒体等内容（超越了文本限制）；

它还可以从一个文件跳转到另一个文件，与世界各地主机的文件连接（超级链接文本）。

**2.3.网页的形成**

网页是由网页元素组成的，这些元素是利用HTML标签描述出来的，然后通过浏览器解析来先显示给用户。



**前端人员开发代码 ----> 浏览器显示代码（解析、渲染） -----> 生成最后的 Web 页面**

**2.4.常用浏览器**   
浏览器是网页显示、运行的平台。常用的浏览器有 IE、火狐（Firefox）、谷歌（Chrome）、Safari和Opera等。平时称为五大浏览器。



查看浏览器市场份额：<http://tongji.baidu.com/data/browser>  
**浏览器内核**   
浏览器内核（渲染引擎）：负责读取网页内容，整理讯息，计算网页的显示方式并显示页面。

|  |  |  |
| --- | --- | --- |
| **浏览器** | **内核** | **备注** |
| IE、Edge | Trident | IE,猎豹，360，百度 |
| firefox | Gecko | 火狐浏览器内核 |
| Safari | Webkit | 苹果浏览器内核 |
| chrome/Opera | Blink | chorme/Opera浏览器内核，Bink属于Webkit的一个分支 |

目前国内一般浏览器都会采用 Webkit/Blink 内核，如 360、UC、QQ、搜狗等。

**2.5.Web标准（重点）**  
**Web 标准**是由**W3C 组织**和其他标准化组织制定的一系列标准的集合。**W3C（万维网联盟）**是国际最著名的标准化组织。

**2.5.1.为什么需要Web标准**   
浏览器的不同，他们显示页面或者排版就有些许差异。

Web 标准是由 W3C 组织和其他标准化组织制定的一系列标准的集合。W3C（万维网联盟）是国际最著名的标准化组织。

遵循 Web 标准除了可以让不同的开发人员写出的页面更标准、更统一外，还有以下优点：

让 Web 的发展前景更广阔。

内容能被更广泛的设备访问。

更容易被搜寻引擎搜索。

降低网站流量费用。

使网站更易于维护。

提高页面浏览速度。

**2.5.2.Web标准的构成**

主要包括**结构（Structure）**、**表现（Presentation**）和**行为（Behavior）**三个方面。

|  |  |
| --- | --- |
| **标准** | **说明** |
| 结构 | 结构用于网页元素进行整理和分类，主要为HTML |
| 表现 | 表现用于设计网页元素的排版、颜色、大小等外观样式，主要指CSS |
| 行为 | 行为指网页模型的定义及**交互**的编写，主要指JavaScript |

Web 标准提出的最佳体验方案：结构、样式、行为相分离。

简单理解：结构写到 HTML 文件中，表现写到 CSS 文件中，行为写到 JavaScript 文件中。



**3.HTML的标签**

|  |  |
| --- | --- |
| 本章 | 学习线路  标签的书写注意规范  HTML的骨架标签  超链接标签  图片标签和alt、tittle的区别  相对路径的三种形式 |

**3.1.基础语法概述**   
 1. HTML 标签是由**尖括号包围的关键词**，例如。

2. HTML 标签通常是**成对出现**的，例如和，我们称为**双标签。**标签对中的第一个标签是**开始标签**，第二个标签是**结束标签**。

3. 有些特殊的标签必须是单个标签（极少情况），例如  
 ，我们称为**单标签。**

**3.2.标签的关系**   
双标签的关系分为两种：**包含关系**和**并列关系**。



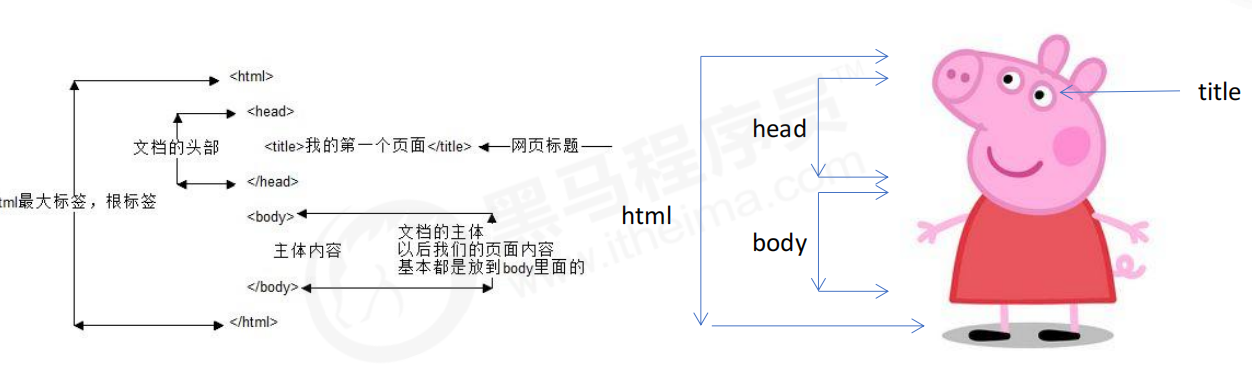
**3.3.第一个HTML网页**   
每个网页都会有一个基本的结构标签（也称为骨架标签），页面内容也是在这些基本标签上书写。HTML页面也称为 HTML 文档.

|  |
| --- |
| <html>   <head>   <title>我的第一个页面</title>  </head>  <body>   大学生活绚烂多彩，网页设计有趣得很。  </body>  </html> |

|  |  |  |
| --- | --- | --- |
| **标签名** | **定义** | **说明** |
|  | HTML定义标签 | 页面中最大的标签，我们称之为跟标签 |
|  | 文档的头部 | 注意在标签内我们必须要设置的标签是 |
|  | 文档的标题 | 让页面有一个网页标题 |
|  | 文档的主体 | 元素包含文档所有的内容，页面内容 |

HTML 文档的的后缀名必须是 .html 或 .htm ，浏览器的作用是读取 HTML 文档，并以网页的形式显示出它们。此时，用浏览器打开这个网页，我们就可以预览我们写的第一个 HTML 文件了。

**3.4.基本结构的总结**

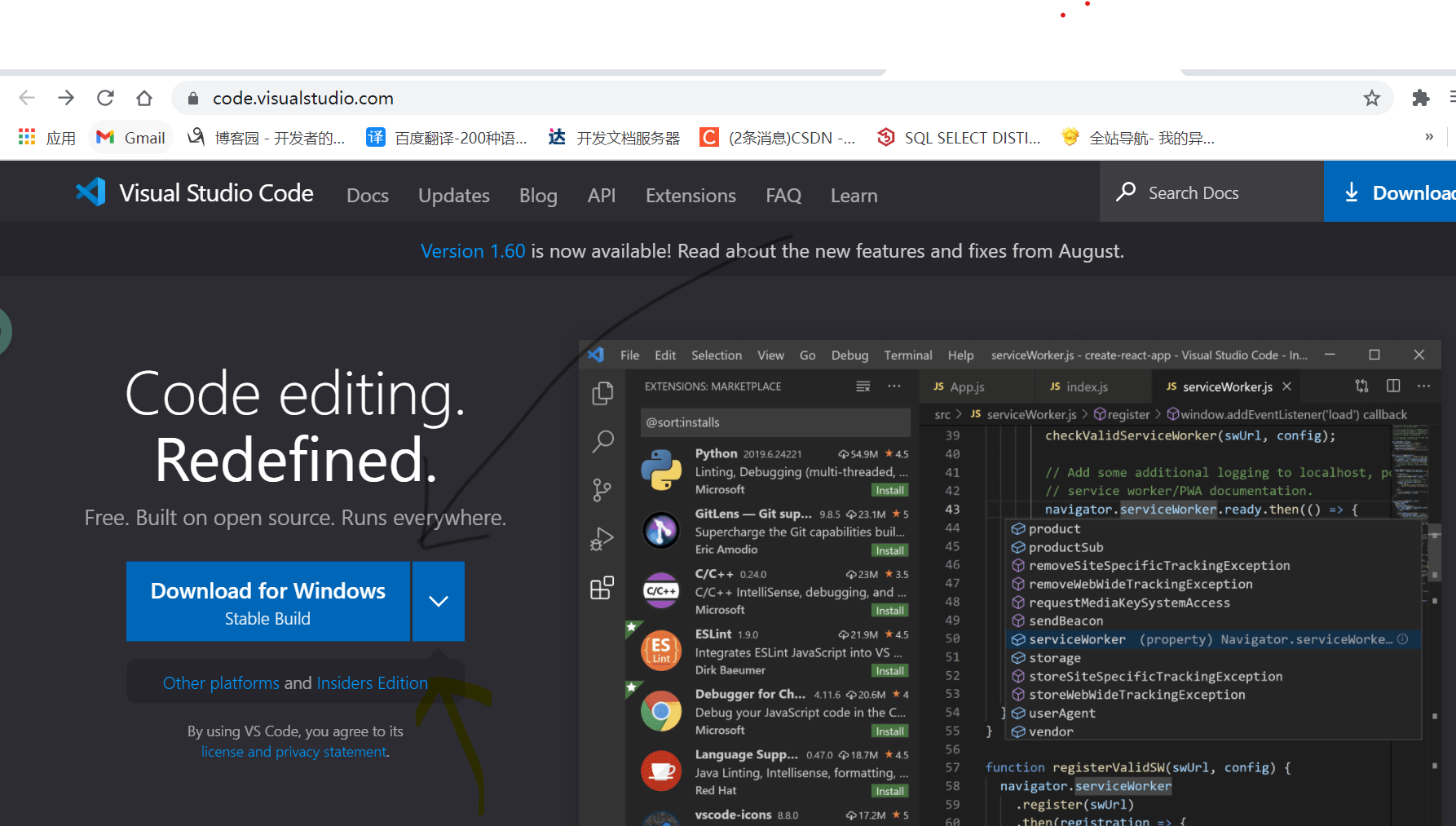


**4.开发工具**



**4.1.VScode的安装和使用**

**4.1.1.VScode的下载**



**4.1.2.VScode的安装**

直接双击安装，直接下一步即可。

安装好的VScode界面全是英文界面，为了方便我们尽快入门VScode，我们可以选择将VScode汉化。汉化步骤：

1. 使用快捷键【Ctrl+Shift+P】，然后搜索【Configure Display Language】；  
2. 安装【Install】中文插件【Chinese （Simplified）Language Pack for Visual Studio Code】，将原本的en改为zh-CN; 3. 重启VScode.

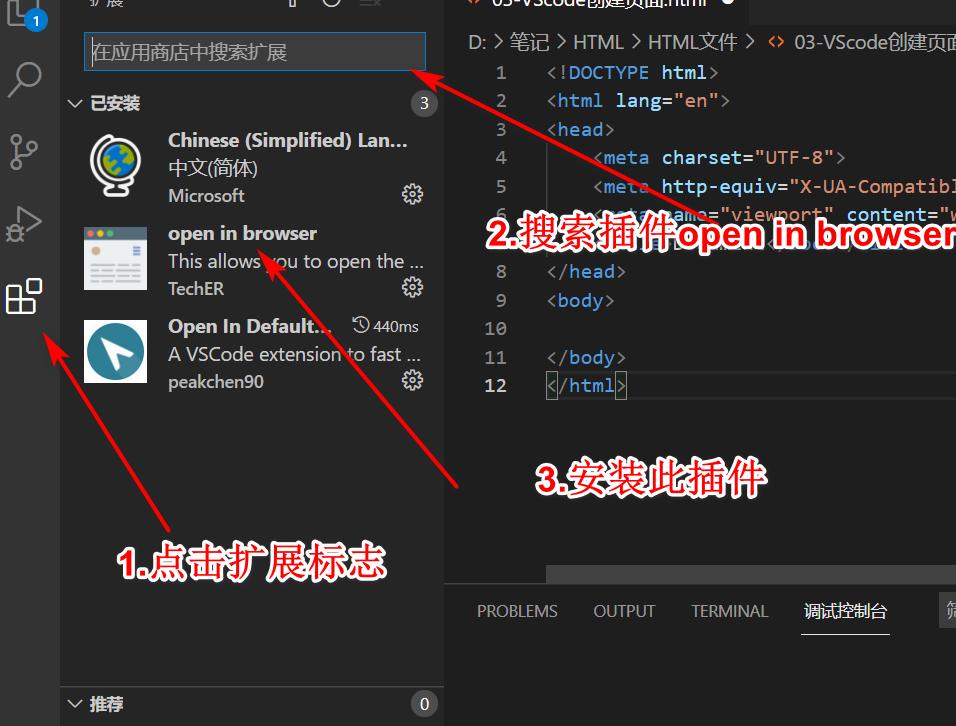
**4.1.3.创建HTML文件**

打开VScode后，只需要点击文件(File),新建文件(Ctrl+N)便可以新建一个文件，将这个新创建的文件保存(Ctrl+S)为html的文件，此时我们只 需要输入**!**选择第一个(或者按住键盘tab键)就可以生成一个html文件。

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="UTF-8">  <meta http-equiv="X-UA-Compatible" content="IE=edge">  <meta name="viewport" content="width=device-width, initial-scale=1.0"> <title>Document</Doc></title>  </head>  <body>  </body>  </html> |

创建好HTML文件后，我们就可以点击运行---启动调试，这是HTML就会在浏览器中打开。但为了方便，我们可以安装插件，一边快速打开 HTML网页。安装插件方式如下：

1. 点击左侧菜单栏的扩展程序：  
2. 搜索【open in browser】，安装此插件；  
3. 这时在HTML页面单机鼠标右键选择open in default browser就可快速在浏览器中打开。



扩展：如果想对文件文字大小进行调整，可以Ctrl+加号放大字体，Ctrl+减号缩小字体。

**4.1.4.VScode插件推荐**

|  |  |
| --- | --- |
| **插件** | **作用** |
| Chinese （Simplified）Language Pack for Visual Studio Code | 中文语言包 |
| Open In Browser | 右键选择在浏览器中打开html文件 |
| JS-CSS-HTML Formatter | 每次保存都会自动格式化js,css,html代码 |
| Auto Rename Tag | 自动重命名配对的HTML或XML标签 |
| CSS Peek | 追踪至样式 |

**5.HTML骨架代码理解**

**5.1.文档类型声明标签**

文档类型声明，作用就是告诉浏览器使用哪种HTML版本来显示网页。

|  |
| --- |
| <!DOCTYPE html> |

**这句代码的意思是: 当前页面采取的是 HTML5 版本来显示网页.**

注意：

 声明位于文档中的最前面的位置，处于标签之前；

不是一个 HTML 标签，它就是文档类型声明标签。

**5.2.lang语言种类**

用来定义当前文档显示的语言：

1. en定义语言为英语；  
2. zh-CN定义语言为中文。  
3. fr定义语言为法文。。。。

简单来说en就是英文网页，zh-CN就是中文网页。

其实对于文档显示来说，定义成en也可以显示中文，定义zh-CN的文档也可以显示英文。

这个属性对浏览器和搜索引擎(百度.谷歌等)还是有作用的

**meta标签的作用：**meta标签是head部的一个辅助性标签，提供关于 HTML 文档的元数据。它并不会显示在页面上，但对于机器是可读 的。可用于浏览器（如何显示内容或重新加载页面），搜索引擎（SEO），或其他 web 服务。

meta标签里的数据是**供机器解读的**，其主要作用有：搜索引擎优化（SEO），定义页面使用语言，自动刷新并指向新的页面，实现网页转 换时的动态效果，控制页面缓冲，网页定级评价，控制网页显示的窗口等等。

关于meta标签和其属性的阅读：<https://www.deanhan.cn/html-meta.html>  
**5.3.字符集**   
**字符集 (Character set)**是多个字符的集合。以便计算机能够识别和存储各种文字。  
在标签内，可以通过标签的 charset 属性来规定 HTML 文档应该使用哪种字符编码。

|  |
| --- |
| <meta charset=" UTF-8" /> |

charset 常用的值有：GB2312 、BIG5 、GBK 和 UTF-8，其中 UTF-8 也被称为万国码，基本包含了全世界所 有国家需要用到的字符.

**注意：\***\*上面语法是必须要写的代码，否则可能引起乱码的情况。一般情况下，统一使用“UTF-8”编码，尽量\*\* **统一写成标准的 "UTF-8"，不要写成 "utf8" 或 "UTF8"。**

各国编码对照：

|  |  |  |
| --- | --- | --- |
| **语言** | **字符集** | **正式名称** |
| 英语、西欧语 | ASiCII，ISO-8859-1 | MBCS 多字节 |
| 简体中文 | GB2312 | MBCS 多字节 |
| 繁体中文 | BIG5 | MBCS 多字节 |
| 简繁中文 | GBK | MBCS 多字节 |
| 中文、日文及朝鲜语 | GB18030 | MBCS 多字节 |
| 各国语言 | UNICODE，UCS | DBCS 宽字节 |

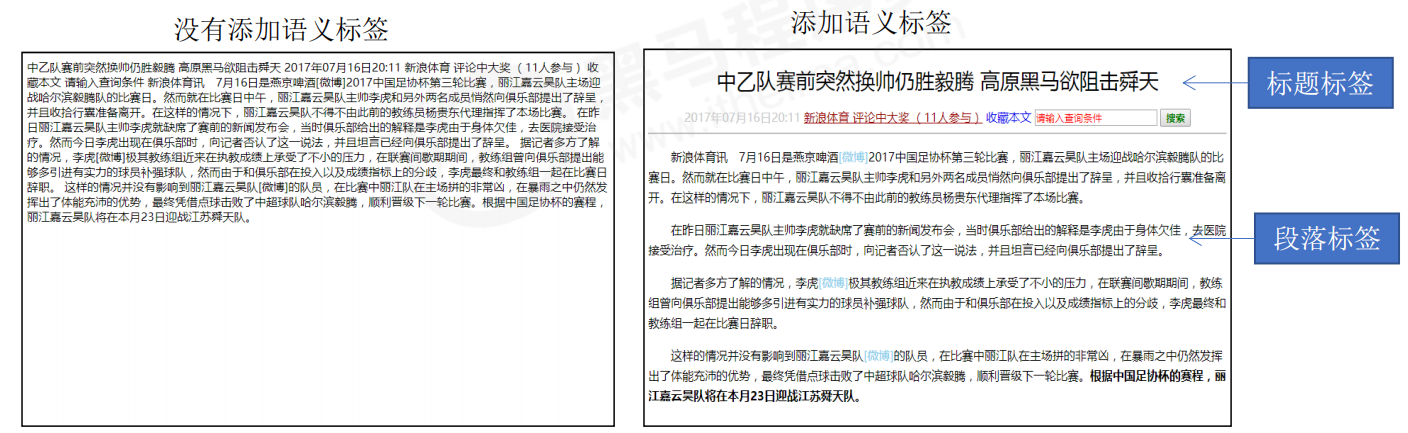
关于字符编码集的相关阅读：<https://www.cnblogs.com/lzmrex/articles/9906102.html>  
**拓展：**  
在HTML中有时候需要一些注释让其他的开发人员明白你所写的代码的含义，所以往往需要添加一些注释。而这些注释只能用于对代码的解 释，不能被加载到网页中，所以需要特殊的标志引入这些注释。

HTML的注释标识为：

|  |
| --- |
| <!-- 注释 --> |

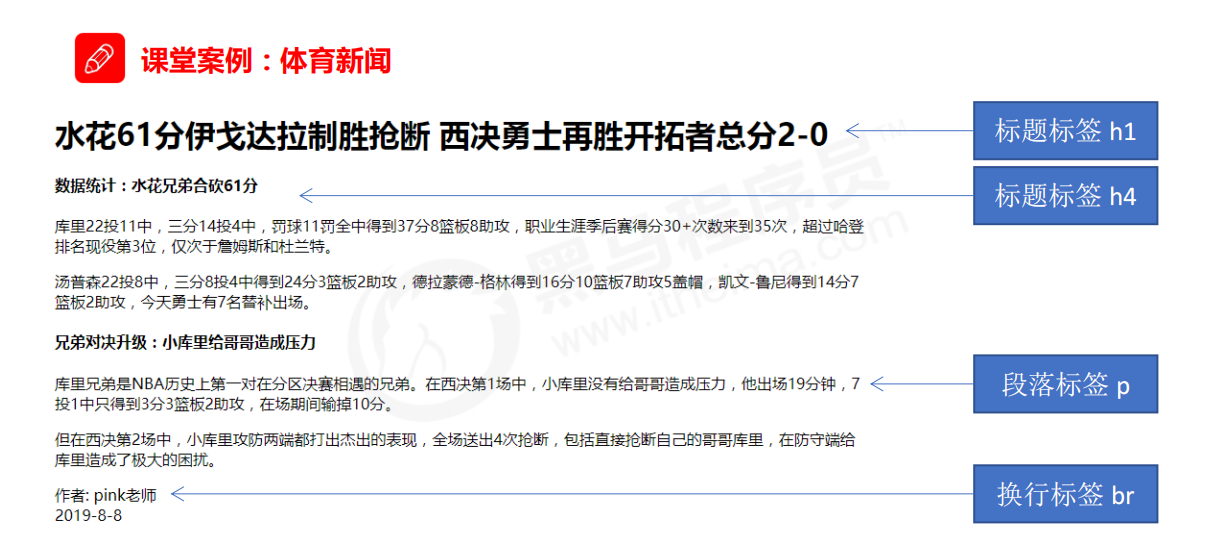
在VS Code中注释可以用Ctrl+/的快捷键生成   
**6.标签的语义**

学习标签是有技巧的，重点是记住每个标签的语义。简单理解就是指标签的含义，即这个标签是用来干嘛的。**根据标签的语义，在合适的地方给一个最为合理的标签，可以让页面结构更清晰。**



**6.1.标题标签--**  
**-**

为了使网页更具有语义化，我们经常会在页面中用到标题标签。HTML 提供了 6 个等级的网页标题，即  
**-**

**。**

|  |
| --- |
| <h1> 我是一级标题 </h1> |

**单词 head 的缩写，意为头部、标题。**

**标签语义：作为标题使用，并且依据重要性递减。**

**特点：**

|  |  |
| --- | --- |
|  | **加了标题标签的文字会变得加粗，字号也会依次变大；每个标题都是独占一行。** |

|  |
| --- |
| <h1>标题一共六级选,</h1>  <h2>文字加粗一行显。</h2>  <h3>由大到小依次减，</h3>  <h4>从重到轻随之变。</h4>  <h5>语法规范书写后，</h5>  <h6>具体效果刷新见。</h6> |

**6.2.段落和换行标签**   
**6.2.1.段落标签---**  
在网页中，要把文字有条理地显示出来，就需要将这些文字分段显示。在 HTML 标签中，  
标签用于定义段   
落，它可以将整个网页分为若干个段落。

|  |
| --- |
| <p> 我是一个段落标签 </p> |

单词 paragraph [ˈpærəgræf] 的缩写，意为段落。  
标签语义：可以把 HTML 文档分割为若干段落。

**特点：**

文本会在一个段落中根据浏览器窗口的大小自动换行；

段落和段落之间保有空隙。

####   
**6.2.2.换行标签**   
在 HTML 中，一个段落中的文字会从左到右依次排列，直到浏览器窗口的右端，然后才自动换行。如果希望 某段文本强制换行显示，就需要使用换行标签

|  |
| --- |
| <br/> |

单词 break 的缩写，意为打断、换行。  
标签语义：强制换行。

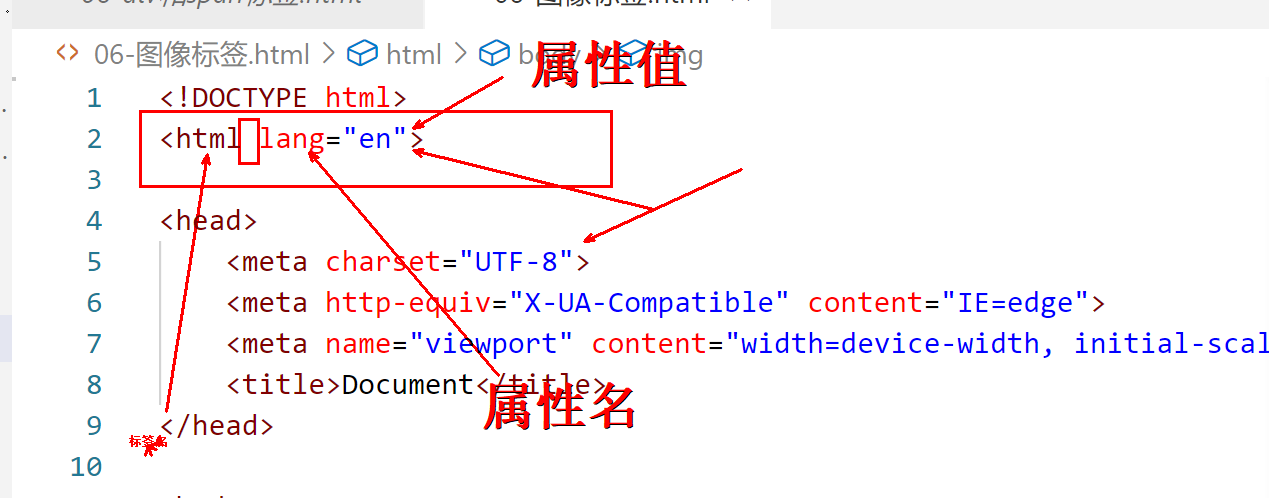
**特点：**



是一个单标签；



标签只是简单地开始新的一行，跟段落不一样，段落之间会插入一些垂直的间距。  
**6.3.案例分析**

**6.4.格式化标签**   
在网页中，有时需要为文字设置**粗体**、斜体或下划线等效果，这时就需要用到 HTML 中的文本格式化标签，使 文字以特殊的方式显示。

标签语义: 突出重要性, 比普通文字更重要.

|  |  |  |
| --- | --- | --- |
| **语义** | **标签** | **说明** |
| 加粗 | 或 | 推荐使用,语义更强烈 |
| 倾斜 | 或 | 推荐使用,语义更强烈 |
| 删除线 | 或 | 推荐使用,语义更强烈 |
| 下划线 | 或 | 推荐使用,语义更强烈 |

同学们重点记住**加粗**和倾斜。

**6.5.div和span标签**   
**和是没有语义的，它们就是一个盒子，用来装内容的**

|  |
| --- |
| <div>这是头部</div>  <span>今日价格</span> |

div 是 division 的缩写，表示分割、分区。span 意为跨度、跨距。  
**特点：**

 标签用来布局，但是现在一行只能放一个  
 。大盒子

|  |  |
| --- | --- |
|  | 标签用来布局，一行上可以多个。小盒子 |

|  |
| --- |
| <div>我是一个div标签我一个人单独占一行</div>  <div>我是一个div标签我一个人单独占一行</div>  <span>百度</span>  <span>新浪</span>  <span>搜狐</span> |

**6.6.图像标签和路径(重点)**   
**6.6.1.图像标签**

在 HTML 标签中，标签用于定义 HTML 页面中的图像。

|  |
| --- |
| <img src="图像URL" /> |

单词 image 的缩写，意为图像。

src 是标签的必须属性，它用于指定图像文件的**路径和文件名**。

所谓属性：**简单理解就是属于这个图像标签的特性**。  
图像标签的其他属性：

|  |  |  |
| --- | --- | --- |
| **属性** | **属性值** | **说明** |
| src | 图片路径 | 必须属性 |
| alt | 文本 | 图片不能显示时的替换文本 |
| title | 文本 | 鼠标放在图片上能显示文字，作提示用 |
| width | 像素 | 设置图片的宽度 |
| height | 像素 | 设置图片的高度 |
| border | 像素 | 设置图片的边框粗细 |

**图像标签属性注意点：**

图像标签可以拥有多个属性，必须声明在标签名的后面；

属性之间不分先后顺序，标签名与属性，属性与属性之间均有空格分开；

属性采用键值对的格式，即key="value"的格式：属性="属性值"。

重点掌握：

图像标签的哪个属性必须写；

属性alt和title有什么区别。

**6.6.2.路径**

在实际的工作中，我们的文件不能随便乱放，否则用起来很难快速的找到它们。因此我们需要一个专门的文件夹管理他们。

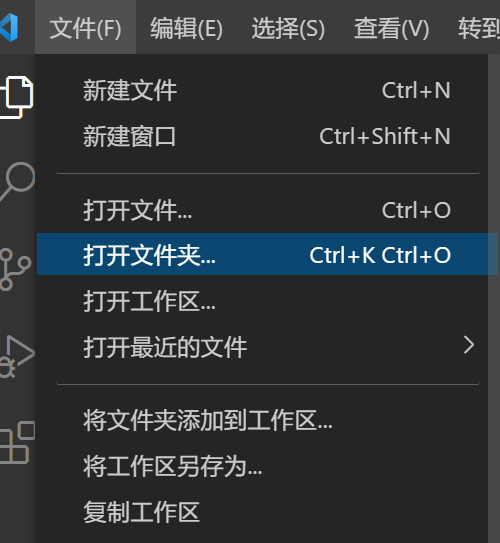
**目录文件夹：**就是普通文件夹，里面存放了我们做页面所需的相关素材，比如html文件，图片等。创建文件夹web存放前端所有文件，里面

可以存放html文件，web下创建images文件夹存放图像。

**根目录：\*\*打开目录文件夹的第一层就是根目录。**

**VSCode中打开目录文件夹：**

文件--打开文件夹--选择目录文件夹



打开目录文件夹的后面就可以新建文件，新建文件夹，刷新和折叠文件夹。

在页面中，我们要使用到的图片会非常的多，通常我们会新建一个文件夹(images)来存放这些图像，这时再查找这些图像，就需要采用路径

的方式来指定图像所在的位置。

路径可分为：

相对路径

绝对路径

**6.6.2.1.相对路径**

以引用**文件所在位置**为参考基础，而建立的目录文件路径。简单来说就是图像相对于HTML页面的位置。

|  |  |  |
| --- | --- | --- |
| **相对路径分类** | **符号** | **说明** |
| 同一级路径 |  | 图像文件位于HTML文件同一级，如 |
| 下一级路径 | / | 图像文件位于HTML文件的下一级，如 |
| 上一级路径 | ../ | 图像文件位于HTML文件的上一级，如 |

相对路径是从代码所在的这个文件出发，去寻找目标文件的，而我们这里所说的上一级、下一级和同一级就是 **图片相对于 HTML 页面的位置**。

**6.6.2.2.绝对路径**   
**绝对路径：**指目录下的绝对位置，直接到达目标位置，通常从盘符开始的位置。

例如：D:\note\HTML\前端学习资料\资料\img 或者完整的网络地址：

|  |
| --- |
| <img src="D:\笔记\HTML\前端学习资料\资料\img\img\1.gif" />  <img src="http://bpic.588ku.com/element\_origin\_min\_pic/18/08/24/05dbcc82c8d3bd356e57436be0922357.jpg"> |

**6.7.超链接标签**   
在HTML中，用标签作为超链接标签，作用是从一个页面跳转到另一个页面或者是另一段文字。

**6.7.1.超链接语法格式：**

|  |
| --- |
| <a href="跳转目标" target="目标窗口的弹出方式"> 文本或图像 </a> |

单词 anchor [ˈæŋkə(r)] 的缩写，意为：锚。  
两个属性的作用：

|  |  |
| --- | --- |
| **属性** | **作用** |
| href | 用于指定链接跳转的url地址，（必须属性）当标签有href属性时，它就具有了跳转功能 |
| target | 用于链接指定的跳转方式，默认值为*self*意为在当前页面打开新链接，blank在新窗口打开 |

**链接的分类：**  
 1. 外部链接: 例如 < a href="https:// [www.baidu.com](http://www.baidu.com/) "> 百度。

2. 内部链接:网站内部页面之间的相互链接. 直接链接内部页面名称即可，例如 < a href="index.html"> 首页。3. 空链接: 如果当时没有确定链接目标时，< a href="#"> 首页。

4. 下载链接: 如果 href 里面地址是一个文件或者压缩包，会下载这个文件。

5. 网页元素链接: 在网页中的各种网页元素，如文本、图像、表格、音频、视频等都可以添加超链接。

6. 锚点链接: 点我们点击链接,可以快速定位到页面中的某个位置.：

在链接文本的 href 属性中，设置属性值为 #名字的形式，如第2集

找到目标位置标签，里面添加一个 id 属性 = 刚才的名字，如：  
 **第2集介绍**

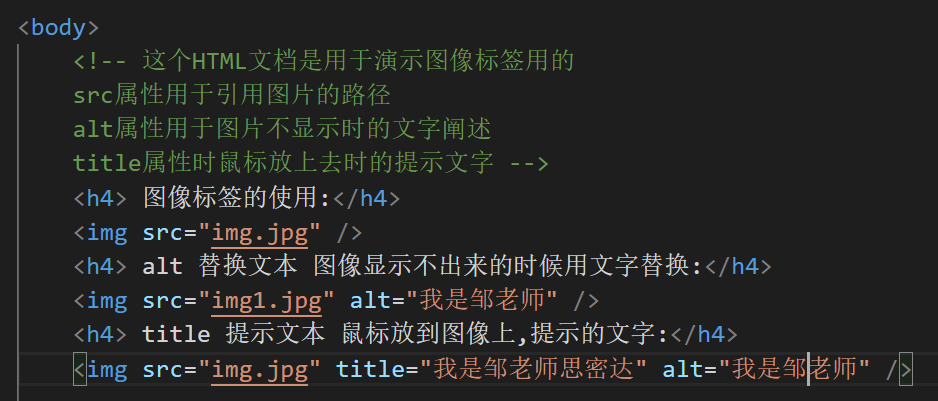
**7.HTML中的注释和特殊字符**

**7.1.注释**   
如果需要在HTML文档中添加一些便于阅读和理解，但又不需要在页面中显示的文字就是**注释**。所以需要注释标签：

|  |
| --- |
| <!--注释文字-->  在VSCode中可以用快捷键 Ctrl + /快捷键生成注释 |

注释标签里的内容是给开发人员开的，不会显示在页面中。

添加注释是方便开发人员阅读，了解代码的含义。网页是不会执行标签里的内容的。



**7.2.特殊字符**   
在HTML页面中，一些特殊的字符很难或者不方便使用，此时我们就可以使用下面的字符代替：

|  |  |  |
| --- | --- | --- |
| **特殊字符** | **描述** | **字符的代替码** |
|  | 空格符 |  |
| < | 小于符 | < |
| > | 大于符 | > |
| & | 与号 | & |
| ￥ | 人民币 | ¥ |
| © | 版权符 | © |
| ® | 注册商标 | ® |
| ° | 度 | ° |
| ± | 正负符 | ± |
| ✖ | 乘号 | × |
| ➗ | 除号 | ÷ |
| ² | 平方2-上标 | ² |
| ³ | 3次方 | ³ |

重点记住：**空格、大于号、小于号**。其余用到的情况可以查看文档。

**8.表格标签**

**8.1.表格的主要作用**   
表格主要是用于显示，展示数据，以为表格可以使数据显示更规整，可读性好。特别使后台展示数据的时候，能够熟练运用表格展示 数据尤为重要。

**8.2.表格的语法**

|  |  |
| --- | --- |
|  | table标签 |

|  |
| --- |
| <table>  <tr>  <td>单元格里的文字</td>  ....  </tr>  ....  </table> |

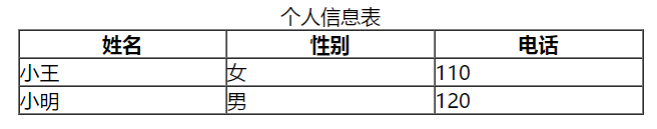
用于定义表格标签；

tr标签标签用于定义表格中的行，必须嵌套在table标签中使用；

td标签用于定义表格中的单元格，必须嵌套在tr标签中使用；

td(table data)指表格中的数据。  
**8.3.表格中的单元格标签**

一般表头单元格位于表格的第一行或第一列，表头单元格中的文字**居中加粗**展示。  
（table head）标签表示HTML表格的表头部分。



|  |
| --- |
| <table>  <tr>  <th>姓名</th>  ...  </tr>  ...  </table> |

**8.4.表格的属性**   
表格的属性在实际开发中并不常用，我们会通过学习css来进行设置。

|  |  |  |
| --- | --- | --- |
| **属性名** | **属性值** | **描述** |
| align | left,center,right | 规定表格相对周围元素的对齐方式 |
| border | 1或"" | 规定表格是否有边框，默认值为""为没有边框 |
| cellspadding | 像素值 | 规定表格单元边沿与内容之间的空白，默认1像素 |
| cellspacing | 像素值 | 规定单元格之间的空白，默认2像素 |
| width | 像素值或百分比 | 规定表格的宽度 |

**8.5.案例练习**   
制作以下小说排行榜：



**先制作表格的结构：**  
1. 第一行里是th表头单元格；  
2. 第二行开始是td单元格里的内容；  
3. 单元格里可以放任何元素，文字链接图片皆可。

**再书写表格的属性：**  
1. 用到宽度高度边框cellpadding 和 cellspacing   
2. 表格浏览器对其align.

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | | |  | | --- | | <table align="center" width="500" height="249" border="1" cellspacing="0"> <thead>  <tr>  <th>排名</th>  <th>关键词</th>  <th>趋势</th>  <th>进入搜索</th>  <th>最近七日</th>  <th>相关链接</th>  </tr>  </thead>  <tbody>  <tr>  <td>1</td>  <td>鬼吹灯</td>  <td><img src="down.jpg"></td> | | |



|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | | |  | | --- | | <td>456</td>  <td>123</td>  <td> <a href="#">贴吧</a> <a href="#">图片</a> <a href="#">百科</a> </td> </tr>  <tr>  <td>1</td>  <td>鬼吹灯</td>  <td><img src="down.jpg"></td>  <td>456</td>  <td>123</td>  <td> <a href="#">贴吧</a> <a href="#">图片</a> <a href="#">百科</a> </td> </tr>  <tr>  <td>3</td>  <td>西游记</td>  <td><img src="up.jpg"></td>  <td>456</td>  <td>123</td>  <td> <a href="#">贴吧</a> <a href="#">图片</a> <a href="#">百科</a> </td> </tr>  <tr>  <td>1</td>  <td>鬼吹灯</td>  <td><img src="down.jpg"></td>  <td>456</td>  <td>123</td>  <td> <a href="#">贴吧</a> <a href="#">图片</a> <a href="#">百科</a> </td> </tr>  <tr>  <td>1</td>  <td>鬼吹灯</td>  <td><img src="down.jpg"></td>  <td>456</td>  <td>123</td>  <td> <a href="#">贴吧</a> <a href="#">图片</a> <a href="#">百科</a> </td> </tr>  <tr>  <td>1</td>  <td>鬼吹灯</td>  <td><img src="down.jpg"></td>  <td>456</td>  <td>123</td>  <td> <a href="#">贴吧</a> <a href="#">图片</a> <a href="#">百科</a> </td> </tr>  </tbody>   </table> | | |

**8.6.表格结构标签**

使用场景：因为表格可能很长，为了更好表示表格的语义，可以将表格分割成表格头部和表格主体两部分。

在表格标签中，可以用标签表格头部区域，标签表示表格主体部分，这样可以更好的表示表格的结构。

：用于定义表格的头部。内部必须。一般是位于第一行。

：用于定义表格的主体，主要用于放数据本体。

以上标签都是放在

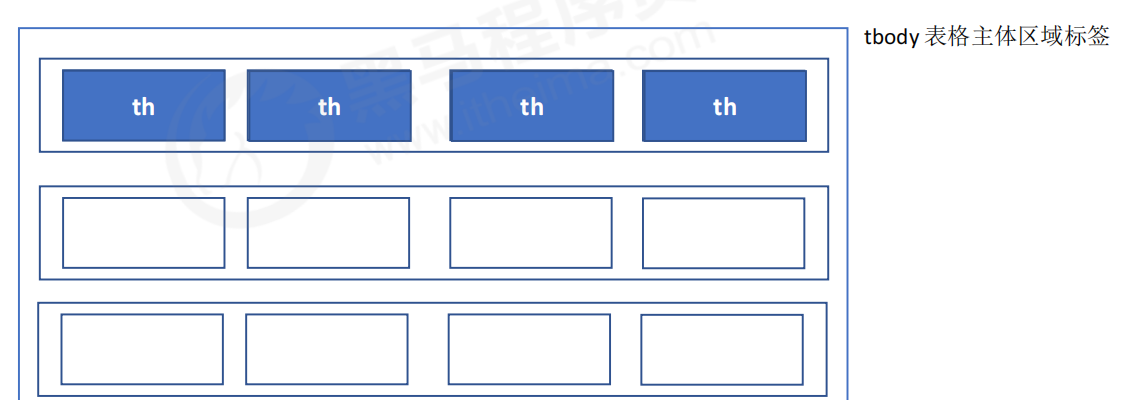
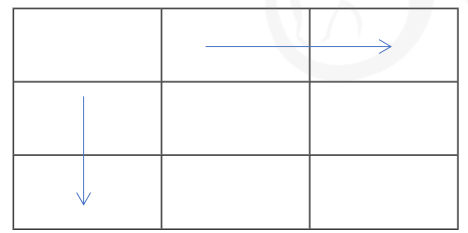
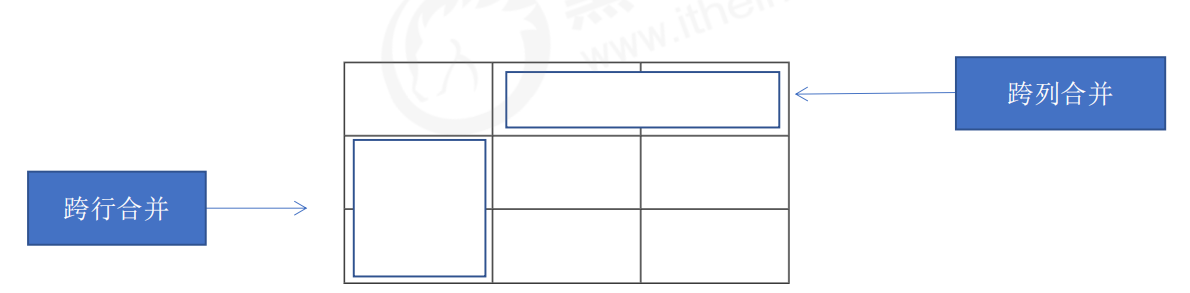
标签中。

**8.7.合并单元格**

特殊情况下，我们可以把多个单元格合并成一个单元格。比如制作简历，统计表格等。

**1.合并单元格的方式：**

|  |  |  |
| --- | --- | --- |
|  | | 跨行合并：rowspan="合并单元格的个数"； |
|  | 跨列合并：colspan="合并单元格的个数"。 | |



|  |
| --- |
| <table width="500" height="249" border="1" cellspacing="0"> <tr>  <td></td>  <td colspan="2"></td>  </tr>  <tr>  <td rowspan="2"></td>  <td></td>  <td></td>  </tr>  <tr>  <td></td>  <td></td>  </tr>  </table> |

**2。目标单元格：**

跨行：最上侧单元格为目标单元格, 写合并代码

跨列：最左侧单元格为目标单元格, 写合并代码

**合并单元格的步骤：**

1. 先确定跨行还是跨列合并；

2. 找到目标单元格，写上合并方式=合并单元格数量。如：；

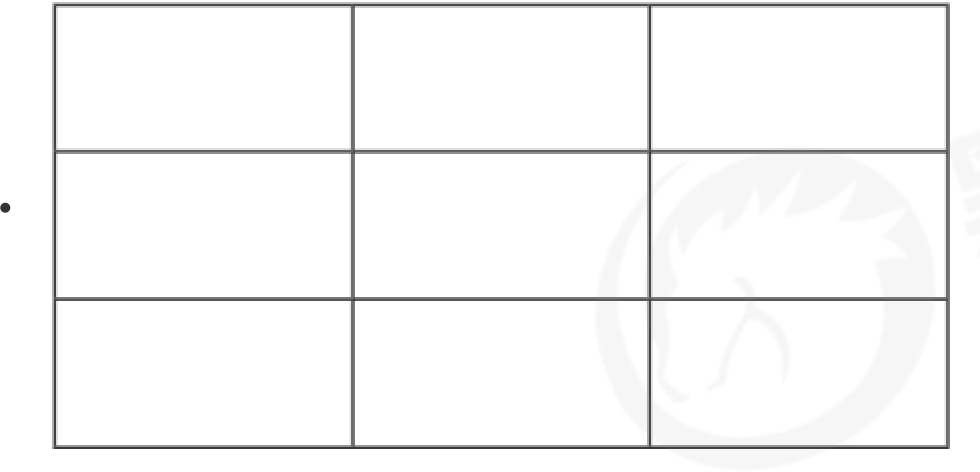
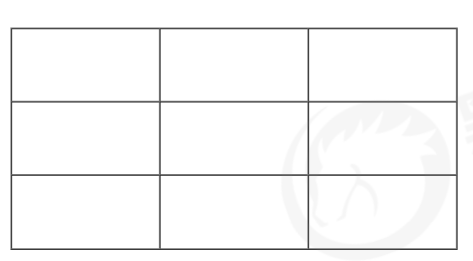
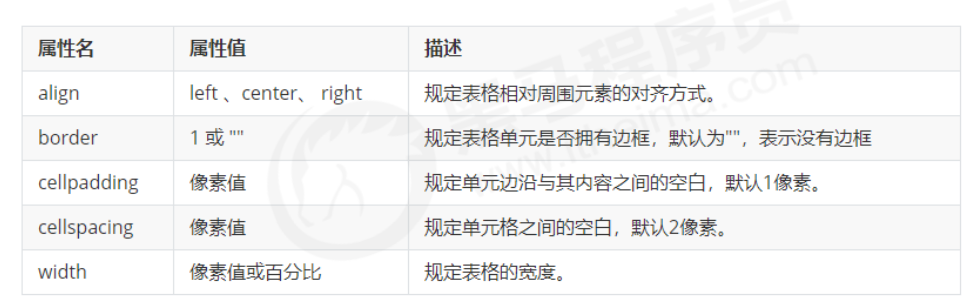
3. 删除多余单元格。

**表格标签总结：**

表格的相关标签；

table 标签 tr 行标签 td 单元格标签 th表头单元格标签 thead表格头部区域标签

|  |  |
| --- | --- |
|  | 表格的相关属性； |



合并单元格。

**9.列表标签**

表格标签是用来展示数据的，而无序标签是**用来布局的**。

列表最大的特点就是整齐，整洁，有序，它作为布局会更多的自由和方便。

根据使用情景不同，列表标签可以分为三大类:**无序标签，有序标签，自定义标签。**

**9.1.无序标签-重点**

ul 标签标识HTML中的无序标签，一般会以项目符号呈现列表项，而列表项使用

|  |  |  |
| --- | --- | --- |
|  | 标签定义。 | |
|  | 格式： |

|  |
| --- |
| <ul>  <li>列表项1</li>  <li>列表项2</li>  <li>列表项3</li>  ...  </ul> |

1. 无序列表的各个列表项之间没有顺序级别之分，是并列的。

2. ul 中只能嵌套 li，直接在 ul 标签中输入其他标签或者文字的做法是不被允许的。3.

4. 与   
 之间相当于一个容器，可以容纳所有元素。

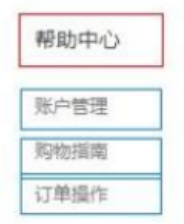
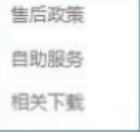
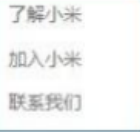
5. 无序列表会带有自己的样式属性，但在实际使用时，我们会使用 CSS 来设置。

**9.2.有序列表-了解**

有序列表即为有排列顺序的列表，其各个列表项会按照一定的顺序排列定义。

在 HTML 标签中，

标签用于定义有序列表，列表排序以数字来显示，并且使用   
1. 标签来定义列表项。



|  |  |
| --- | --- |
|  | 格式： |

|  |
| --- |
| <ol>  <li>列表项1</li>  <li>列表项2</li>  <li>列表项3</li>  ...  </ol> |

1. ol 中只能嵌套li，直接在ol 标签中输入其他标签或者文字的做法是不被允许的。2.

3. 与   
 之间相当于一个容器，可以容纳所有元素。

4. 有序列表会带有自己样式属性，但在实际使用时，我们会使用 CSS 来设置。

**9.3.自定义标签-重点**

自定义列表的使用场景:

自定义列表常用于对术语或名词进行解释和描述，定义列表的列表项前没有任何项目符号。

在 HTML 标签中，

标签用于定义描述列表（或定义列表），该标签会与  
（定义项目/名字）和

（描述每一个项目/名字）一起使用。

|  |  |
| --- | --- |
|  | 格式： |

|  |
| --- |
| <dl>  <dt>名词1</dt>  <dd>名词1解释1</dd>  <dd>名词1解释2</dd>  </dl> |

1. dl 里面只能包含   
 和  
 。

2. 和   
 个数没有限制，经常是一个   
 对应多个  
 。

在 HTML 标签中，

标签用于定义描述列表（或定义列表），该标签会与  
（定义项目/名字）和

（描述每一个项目/名字）一起使用。

|  |  |  |
| --- | --- | --- |
| **标签名** | **定义** | **说明** |
|  | 无序标签 | 里面只能包含li标签，没有是顺序，使用较多。li里面可以包含任何标签 |
|  | 有序标签 | 里面只能包含li标签，有顺序，使用相对较少。li标签里可以包含任何标签 |
|  | 自定义标签 | 里面只能包含dt和dd，dt和dd里面可以放任何标签。 |

**10.表单标签**

现实中，我们可能去银行办理银行卡填写的单子是这样的：



而在网页中，我们也需要使用到类似的表单：



**10.1.为什么我们需要表单**

使用表单的目的就是为了**收集用户的信息。**

在我们网页中，我们也需要跟用户进行交互，收集用户信息，此时我们就需要表单。

**10.2.表单的组成**

在 HTML 中，一个完整的表单通常由**表单域、表单控件（也称为表单元素）和提示信息**3个部分构成。

**表单域**

表单域是一个**包含表单元素的区域**。

在HTML标签中，  
标签用于定义表单域，以实现用户信息的收集和传递。  
form 标签会把它范围内的信息提交给服务器。

格式：

|  |
| --- |
| <form action=“url地址” method=“提交方式” name=“表单域名称"> 各种表单元素控件  </form> |

**常用属性：**

|  |  |  |
| --- | --- | --- |
| **属性** | **属性值** | **作用** |
| action | url地址 | 用于指定接收并处理表单数据的服务器的程序url地址 |
| method | get/post | 用于设置表单提交的方式 |
| name | 名称 | 用于指定表单的名称，以区分页面中多个表单提交 |

|  |  |
| --- | --- |
|  | **表单控件** |

在表单域中可以定义各种表单元素，这些表单元素就是允许用户在表单中输入或者选择的内容控件。内容控件有：**input输入表单元素，select下拉菜单元素，textarea文本域元素。**

1. **input输入表单元素**

在英文单词中，input 是输入的意思，而在表单元素中 **标签用于收集用户信息**。

在 标签中，包含一个 type 属性，根据不同的 type 属性值，输入字段拥有很多种形式（可 以是文本   
字段、复选框、掩码后的文本控件、单选按钮、按钮等）。

|  |
| --- |
| <input type="属性值" />  input标签为单标签  type属性设置不同的属性值用来指定不同的控件类型 |

type的值有：

|  |  |
| --- | --- |
| **属性值** | **描述** |
| button | 定义可点击的按钮 |
| checkbox | 定义复选框 |
| file | 定义输入的字段和'浏览'按钮，供文件上传 |
| hidden | 定义隐藏输入字段 |
| image | 定义图像形式的提交按钮 |
| password | 定义密码字段。该字段字符会被掩饰 |
| radio | 定义单选按钮 |
| reset | 定义重置按钮，重置按钮会清除列表里的所有数据 |
| submit | 定义提交按钮，把表单数据发送到服务器 |
| text | 定义单行的输入字段，默认宽度为20字符，用户可以在其中输入字段 |

除type属性外，input标签还有几个常用属性：

|  |  |  |
| --- | --- | --- |
| **属性** | **属性值** | **描述** |
| name | 用户自定义 | 定义input元素的名称 |
| value | 用户自定义 | 规定input元素的值 |
| checked | checked | input元素首次选中时的默认选项 |
| maxlength | 正整数 | 规定输入字符字段的最大长度 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  | | --- | |  |  |  | | --- | |  |  |  | | --- | |  |  |  | | --- | |  | | name 和value 是每个表单元素都有的属性值,主要给后台人员使用.  name 表单元素的名字, 要求单选按钮和复选框要有相同的name值.  **checked属性主要针对于单选按钮和复选框**, 主要作用一打开页面,就要可以默认选中某个表单元素. maxlength 是用户可以在表单元素输入的最大字符数, 一般较少使用. |

问题：  
 1. 有些表单元素想刚打开页面就默认显示几个文字怎么做?

答: 可以给这些表单元素设置 value 属性=“值”

|  |
| --- |
| 用户名: <input type="text" value="请输入用户名" /> |

2.页面中的表单元素很多，如何区别不同的表单元素?

答: name 属性：当前 input 表单的名字，后台可以通过这个 name 属性找到这个表单。页面中的表单很多，name 的主要作用就是用于区别不同的表单。

|  |
| --- |
| 用户名: <input type="text" value="请输入用户名" name="username" /> |

|  |  |
| --- | --- |
|  | name 属性后面的值，是自定义的  radio (或者checkbox）如果是一组，我们必须给他们命名相同的名字 |

|  |
| --- |
| <input type="radio" name="sex" />男  <input type="radio" name="sex" />女 |

3.如果页面一打开就让某个单选按钮或者复选框是选中状态?

答: checked 属性：表示默认选中状态。用于单选按钮和复选按钮。

|  |
| --- |
| 性别: <input type="radio" name="sex" value="男" checked="checked" />男 <input type="radio" name="sex" value="女" />女 |

4.如何让input表单元素展示不同的形态? 比如单选按钮或者文本框?   
答: type属性：type属性可以让input表单元素设置不同的形态

|  |
| --- |
| <input type="radio" name="sex" value="男" checked="checked" />男 <input type="text" value=“请输入用户名”> |

**10.3.labal标签**   
标签为 input 元素定义标注（标签）。

标签用于绑定一个表单元素, 当点击标签内的文本时，浏览器就会自动将焦点(光标)转到或者   
选择对应的表单元素上,用来增加用户体验.

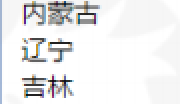
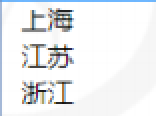
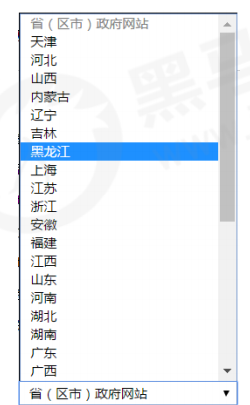
格式：

|  |
| --- |
| <label for="sex">男</label>  <input type="radio" name="sex" id="sex" /> |

核心：  
标签的 for 属性应当与相关元素的 id 属性相同。

**10.4.表单元素**

|  |  |  |
| --- | --- | --- |
| **1.** |  | **标签控件定义下拉列表。** |



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| **在页面中，如果有多个选项让用户选择，并且想要节约页面空间时，我们可以使用** | | | |

<select>

<option>选项1</option>

<option>选项2</option>

<option>选项3</option>

...

</select>

**select 中至少包含一对**

**在**

**<span>标签</span>**

|  |  |
| --- | --- |
| **2.** | **</strong></p></li></ul>** |

<textarea rows="3" cols="20">

文本内容

</textarea>

**<span> 标签可以轻松地**

|  |  |  |
| --- | --- | --- |
|  | **通过** | **创建多行文本输入框。** |