

html5

HTML5介绍

HTML5是用来描述网页的一种语言，被称为超文本标记语言。用HTML5编写的文件，后缀以.html结尾

HTML是一种标记语言，标记语言是一套标记标签。标签是由尖括号包围的关键字，例如：`<html>`

标签有两种表现形式：

双标签，例如：`<html></html>`

单标签，例如：``

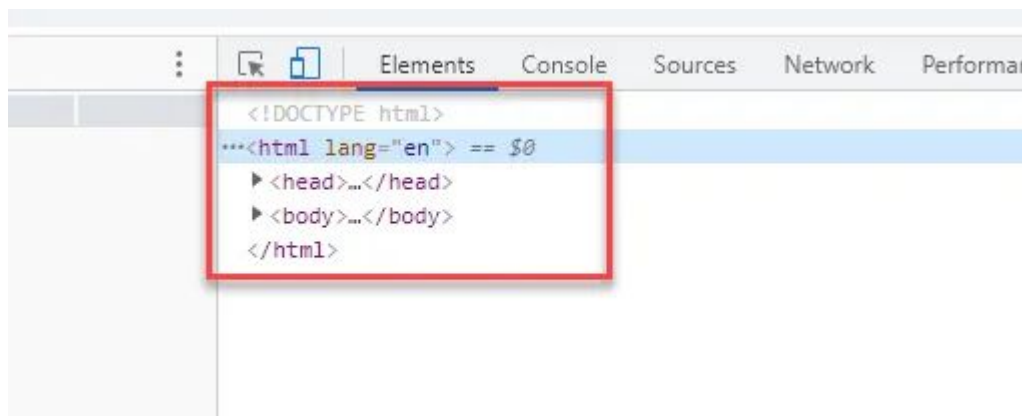
HTML5的DOCTYPE声明

DOCTYPE是document type(文档类型)的缩写。`<!DOCTYPE html >`是H5的声明位于文档的最前面，处于标签之前。他是网页必备的组成部分，避免浏览器的怪异模式。

HTML, XML

```
1 <!DOCTYPE html>
```

HTML5基本骨架



html标签

定义 HTML 文档，这个元素我们浏览器看到后就明白这是个HTML文档了，所以你的其它元素要包裹在它里面，标签限定了文档的开始点和结束点。

Plain Text

```
1 <!DOCTYPE html>
2 <html>
```

```
3 </html>
```

head标签

head标签用于定义文档的头部。文档的头部描述了文档的各种属性和信息，包括文档的标题、在Web中的位置以及和其他文档的关系等。绝大多数文档头部包含的数据都不会真正作为内容显示给读者。

Plain Text

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4     </head>
5 </html>
```

body标签

body元素定义文档的主体。

body元素包含文档的所有内容（比如文本、超链接、图像、表格和列表等等。）

它会直接在页面中显示出来，也就是用户可以直观看到的内容

Plain Text

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4     </head>
5     <body>
6         我会显示在浏览器中
7     </body>
8 </html>
```

title标签

1. 可定义文档的标题。
2. 它显示在浏览器窗口的标题栏或状态栏上。
3. `<title>` 标签是 `<head>` 标签中唯一必须要求包含的东西，就是说写head一定要写title
4. `<title>`的增加有利于SEO优化

SEO是搜索引擎优化的英文缩写。通过对网站内容调整，满足搜索引擎的排名需求

Plain Text

```
1 <!DOCTYPE html>
2 <html>
3     <head>
4         <title>第一个页面</title>
5     </head>
6     <body>
7         我会显示在浏览器中
8     </body>
9 </html>
```

meta标签

meta标签用来描述一个HTML网页文档的属性，关键词等，例如：`charset="utf-8"`是说当前使用的是utf-8编码格式，在开发中我们经常会看到utf-8，或是gbk，这些都是编码格式，通常使用utf-8。

Plain Text

```
1 <!DOCTYPE html>
2 <html lang="en">
3     <head>
4         <meta charset="UTF-8">
5         <title>itbaizhan</title>
6     </head>
7     <body>
8     </body>
9 </html>
```

标题介绍与应用

标题（Heading）是通过`<h1>` - `<h6>`标签进行定义的。

`<h1>`定义最大的标题 `<h6>`定义最小的标题

Plain Text

```
1 <h1>一级标题</h1>
2 <h2>二级标题</h2>
```

```
3 <h3>三级标题</h3>
4 <h4>四级标题</h4>
5 <h5>五级标题</h5>
6 <h6>六级标题</h6>
```

生成h1~h6快捷键：h\$*6

VSCode插件

快速打开浏览器

扩展 -> 搜索open in browser -> 点击安装

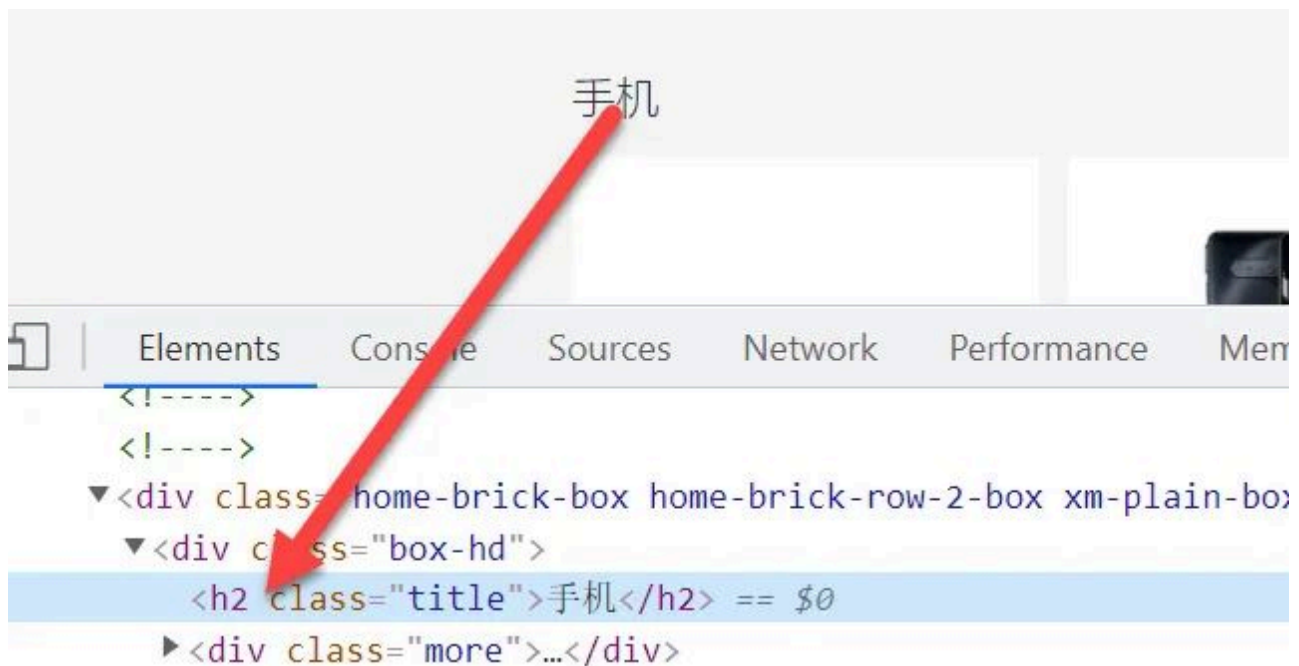
正确使用标题

请确保将 HTML 标题标签只用于标题。

不要仅仅是为了生成粗体或大号的文本而使用标题。

正确使用标题有益于SEO

应该将< h1>用作主标题（最重要的），其后是<h2>（次重要的），再其次是<h3>，以此类推



标题标签位置摆放

在标签中添加属性：align="left | center | right"默认居左

三级标题

标签之段落

段落是通过<p>标签定义的

Plain Text

- 1 <p>这是一个段落 </p>
- 2 <p>这是另一个段落</p>

价格说明:

京东价: 京东价为商品的销售价,是您最终决定是否购买商品的依据。

划线价: 商品展示的划线价为参考价,并非原价,该价格可能是品牌专柜标价、商品吊牌价或由品牌的价格;由于地区、时间的差异性和市场行情波动,品牌专柜标价、商品吊牌价等可能会与您购物时展示

折扣: 如无特殊说明,折扣指销售商在原价、或划线价(如品牌专柜标价、商品吊牌价、厂商指导价、厂商建议零售价)等某

异常问题: 商品促销信息以商品详情页“促销”栏中的信息为准;商品的具体售价以订单结算页价格为准;



换行

如果您希望在不产生一个新段落的情况下进行换行(新行),请使用

元素是一个空的 HTML 元素。

Plain Text

- 1 <p>这个
段落
演示了分行的效果</p>

这个
段落
演示了分行的效果



水平线

`<hr/>` 标签在 HTML 页面中创建水平线

Plain Text

```
1 <hr color="" width="" size="" align=""/>
```

属性：

1. color：设置水平线的颜色
2. width：设置水平线的宽度
3. size：设置水平线的高度
4. align：设置水平线的对齐方式（默认居中），可取值left|right

网站中最多的元素

网站中最多的元素毋庸置疑，一定是图片

`` 标签定义 HTML 页面中的图像

Plain Text

```
1 <img src="" alt="" title="" width="" height="">
```

注意事项

`` 是单标签，不需要进行闭合操作

属性：

1. src：路径（图片地址与名字）
2. alt：规定图像的替代文本

3. width: 规定图像的宽度
4. height: 规定图像的高度
5. title: 鼠标悬停在图片上给予提示

绝对路径

绝对路径是电脑的盘符存储与访问的具体地址

Plain Text

```
1 E:\itbaizhanCode\1.jpg
```

Plain Text

```
1 
```

相对路径

两者相对关系，两者在同一路径下可以直接访问

1. 子级关系: /
2. 父级关系: ../
3. 同级关系: ./ (可以省略)

网络路径

具体的网络地址:<http://iwenwiki.com/api/newworld/images/n1.png>

超链接描述

HTML使用标签<a>来设置超文本链接

超链接可以是一个字，一个词，或者一组词，也可以是一幅图像，您可以点击这些内容来跳转到新的文档

Plain Text

```
1 <a href="url">链接文本</a>
```

超链接属性

在标签中使用了href属性来描述链接的地址

默认情况下，链接将以，以下形式出现在浏览器中：

- 1. 一个未访问过的链接显示为蓝色字体并带有下划线。
- 2. 访问过的链接显示为紫色并带有下划线。
- 3. 点击链接时，链接显示为红色并带有下划线。

超链接表现

当您把鼠标指针移动到网页中的某个链接上时，箭头会变为一只小手



常用文本标签

标签	描述
<code></code>	定义着重文字
<code></code>	定义粗体文本
<code><i></code>	定义斜体字
<code></code>	定义加重语气
<code></code>	定义删除字
<code></code>	元素没有特定的含义

特别提示

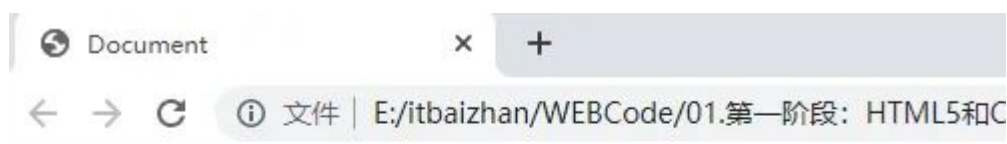
常用文本标签和段落是不同的，段落代表一段文本，而文本标签一般表示文本词汇

有序列表

有序列表是一列项目，列表项目使用数字进行标记。有序列表始于``标签。每个列表项始于``标签。

Plain Text

```
1 <ol>
2     <li>尚学堂</li>
3     <li>百战程序员</li>
4 </ol>
```



1. 尚学堂
2. 百战程序员

type属性

``的属性type 拥有的选项

1. 1 表示列表项目用数字标号 (1,2,3...)
2. a 表示列表项目用小写字母标号 (a,b,c...)
3. A 表示列表项目用大写字母标号 (A,B,C...)
4. i 表示列表项目用小写罗马数字标号 (i,ii,iii...)
5. I 表示列表项目用大写罗马数字标号 (I,II,III...)

有序列表嵌套

列表是可以进行嵌套的

Plain Text

```
1 <ol>
```

```
2     <li>尚学堂</li>
3     <li>
4         <ol>
5             <li>阿里</li>
6             <li>京东</li>
7         </ol>
8     </li>
9     <li>百战程序员</li>
10 </ol>
```

无序列表实现

无序列表是一个项目的列表，此列项目使用粗体圆点（典型的小黑圆圈）进行标记

无序列表始于标签。每个列表项始于标签。

Plain Text

```
1 <ul>
2     <li>尚学堂</li>
3     <li>百战程序员</li>
4 </ul>
```

type属性

的属性type 拥有的选项

- disc 默认实心圆
- circle 空心圆
- square 小方块
- none 不显示

无序列表嵌套

列表是可以进行嵌套的

Plain Text

```
1 <ul>
2     <li>尚学堂</li>
3     <li>
4         <ul>
5             <li>阿里</li>
```

```
6         <li>京东</li>
7     </ul>
8 </li>
9     <li>百战程序员</li>
10 </ul>
```

常见应用场景

1. 无序的列表效果
2. 导航效果

导航效果

Plain Text

```
1 <ul>
2     <li>Xiaomi手机</li>
3     <li>Redmi 红米</li>
4     <li>电视</li>
5     <li>笔记本</li>
6 </ul>
```

服务 | 天星数科 | 有品 | 小爱开放平台 | 企业团购 | 资质证照 | 协议规则 | 下载app | 智能生活 | Select Location

Xiaomi手机 | Redmi 红米 | 电视 | 笔记本 | 平板 | 家电 | 路由器 | 智能硬件



快捷键

快速生成ul+li的布局：ul>li*3（数字根据自己的需要的li数量修改）

表格展示效果

表格在数据展示方面非常简单，并且表现优秀



表格组成与特点

行、列、单元格

单元格特点：同行等高、同列等宽。

表格标签

表格：<table>

行：<tr>

单元格(列)：<td>

Plain Text

```
1 <table>
2   <tr>
3     <td>尚学堂</td>
4     <td>百战程序员</td>
5   </tr>
6   <tr>
7     <td>阿里</td>
8     <td>京东</td>
9   </tr>
10 </table>
```

快捷键

快速生成表格结构：table>tr*2>td{单元格}

表格属性

- 1. border：设置表格的边框
- 2. width：设置表格的宽度
- 3. height：设置表格的高度

单元格合并属性

单元格		单元格	单元格
单元格	单元格	单元格	
	单元格		
单元格	单元格	单元格	单元格

- 水平合并：colspan
- 垂直合并：rowspan

Plain Text

```
1 <table border="1" width="500px" height="200px">
2   <tr>
3     <td colspan="3">单元格1单元格2单元格3</td>
4     <td>单元格4</td>
5     <td>单元格5</td>
6   </tr>
7   <tr>
8     <td rowspan="2">单元格6-11</td>
9     <td>单元格7</td>
10    <td rowspan="3">单元格81318</td>
11    <td colspan="2" rowspan="2">单元格9101415</td>
12  </tr>
13  <tr>
14    <td>单元格12</td>
15  </tr>
16  <tr>
17    <td>单元格16</td>
18    <td>单元格17</td>
19    <td>单元格19</td>
20    <td>单元格20</td>
21  </tr>
22 </table>
```

Form表单

表单在 Web 网页中用来给用户填写信息，从而能采用户信息，使网页具有交互的功能。

所有的用户输入内容的地方都用表单来写，如登录注册、搜索框

表单是由容器和控件组成的,一个表单一般应该包含用户填写信息的输入框,按钮等，这些输入框,按钮叫做控件,表单就是容器,它能够容纳各种各样的控件

Plain Text

```
1 <form action="url" method="get|post" name="myform"></form>
```

属性说明

action服务器地址

name表单名称

method中Get和Post的区别

1. 数据提交方式，get把提交的数据url可以看到，post看不到
2. get一般用于提交少量数据，post用来提交大量数据

表单元素

一个完整的表单包含三个基本组成部分：表单标签、表单域、表单按钮

1. 表单标签
2. 表单域
3. 表单按钮

Plain Text

```
1 <form>
2     <input type="text">
3     <input type="submit">
4 </form>
```

表单元素

文本框

文本域通过<input type="text">标签来设定，当用户要在表单中键入字母、数字等内容时，就会用到文本域

Plain Text

```
1 <form>
2     First name: <input type="text" name="firstname">
3     <br>
4     Last name: <input type="text" name="lastname">
5 </form>
```

First name:

Last name:

密码框

密码字段通过标签 `<input type="password">` 来定义

Plain Text

```
1 <form>
2     Password: <input type="password" name="pwd">
3 </form>
```

Password:

温馨提示

密码字段字符不会明文显示，而是以星号或圆点替代

提交按钮

当用户单击确认按钮时，表单的内容会被传送到另一个文件。表单的动作属性定义了目的文件的文件名。由动作属性定义的这个文件通常会对接收到的输入数据进行相关的处理

Plain Text

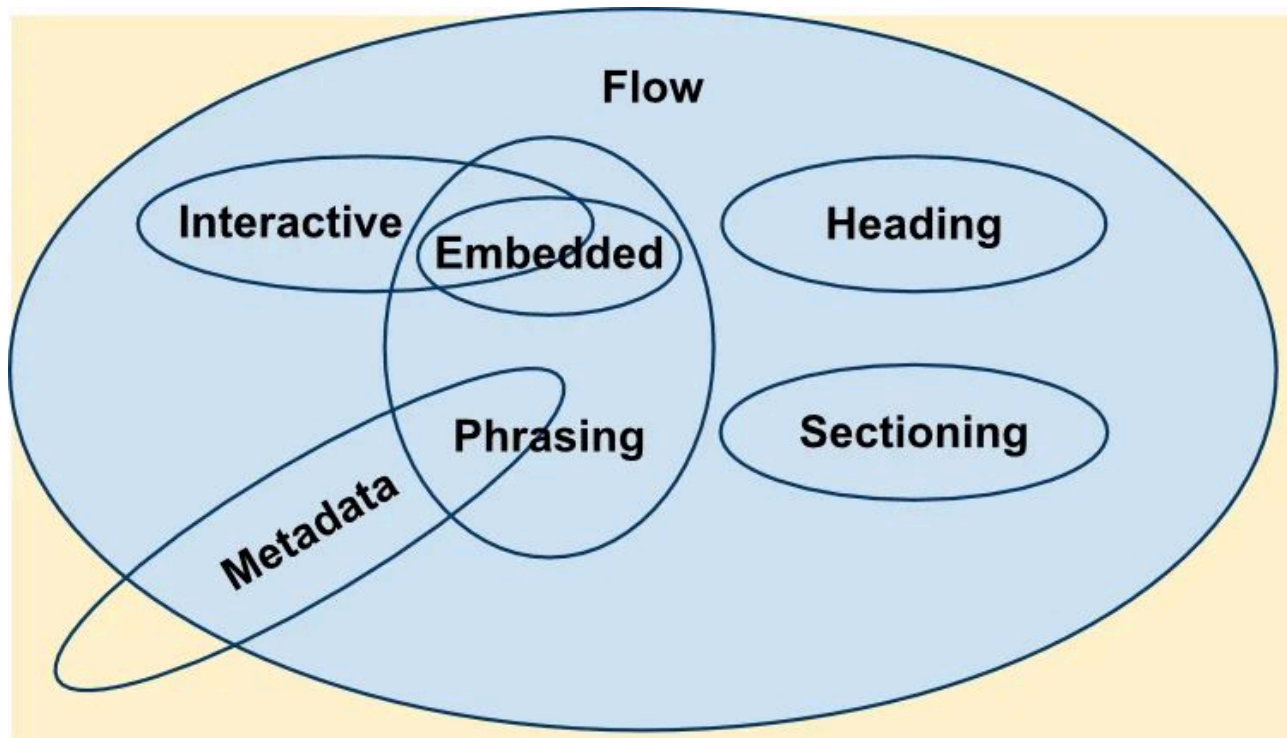
```
1 <form name="input" action="url" method="get">
2     Username: <input type="text" name="user">
3     <input type="submit" value="Submit">
4 </form>
```

Username:

Submit

块元素与行内元素（内联元素）

HTML5出现之前，经常把元素按照块级元素和内联元素来区分。在HTML5中，元素不再按照这种方式来区分，而是按照内容模型来区分，分为元数据类型(metadata content)、区块型(sectioning content)、标题型(heading content)、文档流型(flow content)、语句型(phrasing content)、内嵌型(embedded content)、交互型 (interactive content)。元素不属于任何一个类别，被称为穿透的，元素可能属于不止一个类别，称为混合的



详细参考地址：https://developer.mozilla.org/zh-CN/docs/Web/Guide/HTML/Content_categories

虽然到了HTML5的版本，元素分类更细致了，但是这对初学者并不友好,所以我们仍然按照块元素和内联元素做区分，这对我们的布局起到了至关重要的作用

内联元素和块级元素的区别

块级元素	内联元素
块元素会在页面中独占一行（自上向下垂直排列）	行内元素不会独占页面中的一行，只占自身的大小

可以设置width, height属性	行内元素设置width, height属性无效
一般块级元素可以包含行内元素和其他块级元素	一般内联元素包含内联元素不包含块级元素

常见块级元素

div、form、h1~h6、hr、p、table、ul、等

常见内联元素(行内元素)

a、b、em、i、span、strong等

行内块级元素（特点：不换行、能够识别宽高）

button、img、input等

HTML5新增标签

HTML5是HTML最新的修订版本，2014年10月由万维网联盟（W3C）完成标准制定

在HTML5出现之前，我们一般采用DIV+CSS布局我们的页面。但是这样的布局方式不仅使我们的文档结构不够清晰，而且不利于搜索引擎爬虫对我们页面的爬取。为了解决上述缺点，HTML5新增了很多新的语义化标签

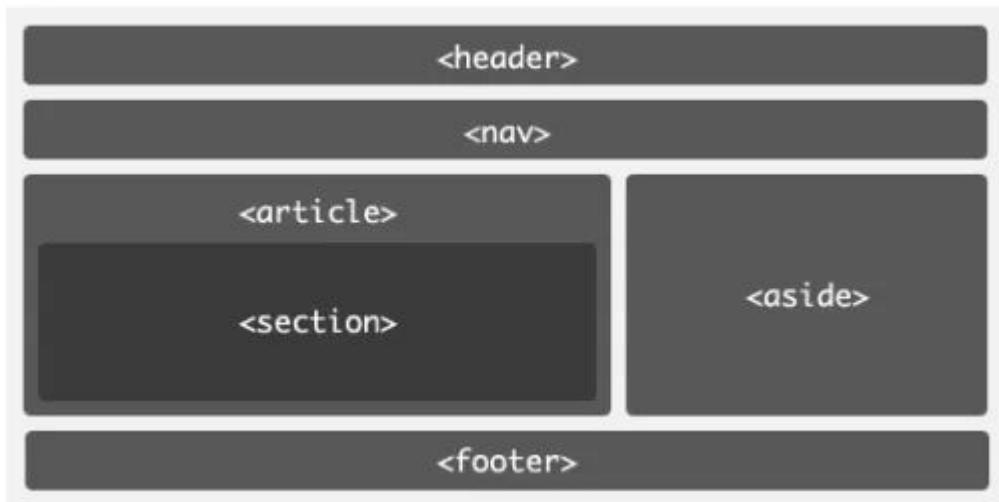
扩展知识

div容器元素，也是页面中见到的最多的元素

div实现



H5新标签实现



H5新标签

1. `<header></header>` 头部
2. `<nav></nav>` 导航
3. `<section></section>` 定义文档中的节,比如章节、页眉、页脚
4. `<aside></aside>` 侧边栏
5. `<footer></footer>` 脚部
6. `<article></article>` 代表一个独立的、完整的相关内容块,例如一篇完整的论坛帖子,一篇博客文章,一个用户评论等