

# HTML基础

---

## 基本结构

---

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <title>test</title>
6   </head>
7   <body>
8     Hello world!
9   </body>
10 </html>
```

## 专业术语

---

### 标签

HTML标签是HTML的基本组成,一般由一对<>组成,比如 `<p>hello world</p>` 对于只有一个<>的标签,我们叫单标签,单标签最后需要加 `/` 表示结束,比如 `<img/>`

### 属性

每一个标签的开头,都可以加属性,比如 `<a href=""></a>`,这个href就是属性,每一个元素都有一些自己独有的属性,而script 和style属性则是每个标签都有的.

### 文档

HTML文档从 `<html>` 开始,在 `</html>` 结束,这里面包裹的内容就是HTML文档,文档头由 `<head>` 定义,而文档主体由 `<body>` 定义.

## 标记省略

---

HTML的标签并不是都要写结束标签的。根据对标签省略的力度来划分，标签可以分为3种：

1. 不允许写结束标签  
比如 `meta`、`input` 等
2. 可以省略结束标签  
`li`、`dt`、`p`、`thead`等
3. 可以完全省略  
`html`、`head`、`body`、`colgroup`

## 标签属性

---

如果属性为boolean类型，那么属性有三种写法来开启属性。

```
1
2 <!-- 只写属性，不写属性值 -->
3 <input type="checkbox" checked>
4
5 <!-- 属性值==属性名 -->
6 <input type="checkbox" checked="checked">
7
8
9 <!-- 属性值==空串 -->
10 <input type="checkbox" checked="">
```

属性两边对于字符串并不敏感，可以加双引号、单引号，也可以不加。

```
1 <input type="checkbox">
2 <input type='checkbox'>
3 <input type=checkbox>
```

## 元素

- 置换元素 (replaced element)：指的是元素的内容不由文档直接表示，例如 `<img>`。
- 非置换元素 (nonreplaced element)：指的是在元素自身生成的框中显示的元素。比如 `<span>`，绝大多数元素都是非置换元素。

## 文档相关标签

### !DOCTYPE

文档类型声明标签

其实!DOCTYPE并不属于HTML的标签，他的功能是声明文档的类型：表明了本HTML究竟采用哪一版html语言写的,因为除了HTML5之外还有XHTML,HTML 4.01 Strict、HTML 4.01 Transitional等等。不过现在一般都用的是HTML5.

`<!DOCTYPE>` 声明必须是 HTML 文档的第一行，位于 `<html>` 标签之前。之后写上你需要的版本属性

```
1 <!-- 声明HTML5 -->
2 <!DOCTYPE html>
```

## html

---

标志文档起点与终点，还用于定义文档的语言。

- lang="zh-CN"

规定文档所使用的语言（自然语言）

## meta

---

meta可以提供有关页面的元信息,就是说,可以给搜索引擎提供一个简历,以确保被搜索到.

- charset="utf-8"

告诉浏览器web显示页面的时候使用什么字符集，但是要注意浏览器首先会从HTTP响应头来确定字符集，之后才会从文档中查看。但是最好还是写上

## link

---

一般常用于引用外部样式,或者JavaScript文件。

还可以用于图标 的修改。

- rel

规定当前文档与被链接文档之间的关系。

- type

引用的类型

- href

引用的路径

```
1 <link rel="stylesheet" type="text/css" href="引用的路径" />
2 <link rel="shortcut icon" href="引用的路径" />
```

## script

---

# 文档结构

---

## main

---

<main> 标签规定文档的主要内容。

元素中的内容对于文档来说应当是唯一的。它不应包含在文档中重复出现的内容，比如侧栏、导航栏、版权信息、站点标志或搜索表单。

**注释：** 在一个文档中，不能出现一个以上的 `<main>` 元素。`<main>` 元素不能是以下元素的后代：`<article>`、`<aside>`、`<footer>`、`<header>` 或 `<nav>`。

## section

定义文档中的节（section、区段）。比如章节、页眉、页脚或文档中的其他部分。

## body

## header

## footer

## nav

navigation

## section

## aside

## article

## p

paragraph

定义段落

1 | `<p>`块级元素的标签`</p>`

## 文档内容

## figure和figcaption

用于文档的插图

```
1 | <figure>
2 |   
3 |   <figcaption>图片描述</figcaption>
4 | </figure>
```

## details和summary

设置本地折叠功能

- open

如果设置了这个属性，那么就会展开

```
1 <details>
2   <summary>总结</summary>
3   <p>1</p>
4   <p>2</p>
5   <p>3</p>
6 </details>
```

► 总结

## 基础标签

### div

division的缩写，表示分割。

```
1 <div>块级元素</div>
```

### span

span表示跨度。

```
1 <span>行内元素的标签</span>
```

### hr

```
1 <p>
2   hr在换行后加水平线<hr/>就像这样
3 </p>
```

上面就是水平线了

### br

```
1 <p>
2   br仅仅表示换行<br>就像这样
3 </p>
```

### 注释

```
1 <!-- 我是注释 -->
```

## 链接

## a

anchor（锚点）的缩写。

a标签一共有5种链接：

- 外部链接

```
1 | <a href="超链接连接的网址">外部链接</a>
```

- 内部链接

```
1 | <a href="index.html">内部链接</a>
```

- 下载链接：如果链接的是一个zip文件或者其他文件，点击就会下载该文件

```
1 | <a href="game.zip">下载链接</a>
```

- 锚点链接：可以在页面内跳转，跳转到指定的id。

```
1 | <a href="#pos">点我定位到指定位置!</a>
2 | <p id="pos">定位点</p>
```

- 空连接：没有指定id的锚点，会默认跳到页面最上面。

```
1 | <a href="#">返回首部</a>
```

a标签还有一个target属性，用来表示目标窗口的弹出方式。

```
1 | <a href="#" target="_self">默认打开方式，在当前页面打开</a>
2 | <a href="#" target="_blank">新窗口打开</a>
```

## 列表

### ul,li

无序列表

```
1 | <ul>
2 |   <li>我</li>
3 |   <li>是</li>
4 |   <li>无序列表</li>
5 | </ul>
```

- 我
- 是
- 无序列表

# ol,li

顺序列表

```
1 <ol>
2   <li>我</li>
3   <li>是</li>
4   <li>有序列表</li>
5 </ol>
```

- 1. 我
- 2. 是
- 3. 有序列表

属性	值	描述
reversed	reversed	规定列表顺序为降序。(9,8,7...)
start	数字	规定有序列表的起始值。
type	1 A a l i	规定在列表中使用的标记类型。

# dl,dt,dd

```
1 <dl>
2   <dt>定义列表</dt>
3   <dd>是专门写定义的列表,如下</dd>
4   <dt>定义:</dt>
5   <dd>我是内容</dd>
6 </dl>
```

定义列表  
是专门写定义的列表,如下  
定义:  
我是内容

# 表单

## form

form可以规定表单的范围，也就是所谓的**表单域**。

一个表单由表单域、控件和提示信息三部分组成。

控件就是输入框那些，提示信息就是姓名，性别那些信息。

```

1 <form action="/demo/demo_form.asp">
2   form表示一个表单，里面本身啥都没有，但是可以放各种表单的标签
3   <br>
4   我是一个input:
5   <input type="text" name="firstname" value="我是一个text
input">
6   <br>
7   <button type="button">我是一个button</button>
8 </form>

```

## input

input下有很多属性

- number

```
1 <input type="number" min="14" max="27" step="3">
```

- search

```
1 <input type="search">
```

- date

输入日期

```
1 <input type="search">
```

- time

- text

value里面输入的值就是默认值，maxlength可以来规定最大的输入字符数。

```

1 <form>
2 输入框: <input type="text" value = "默认值" maxlength="6"
placeholder="提示文字">
3 </form>

```

输入框:

- password

```
1 <input type="password">
```

- file

文件限制中间用逗号隔开

```
1 <input type="file" name="img" accept="image/gif,
image/jpeg">
```



- reset

点这个按钮会自动把所有的控件信息**初始化**。如果把控件信息用JS清空，也会默认初始化。

```
1 | <input type="reset">
```

重置

- button

```
1 | <input type="button" value="按钮">
```

按钮

- submit

```
1 | <input type="submit">
```

提交

- url
- radio,checkbox

value属性的值是发送给后端的。name是用于分组的，只有同一组name下，控件才能生效。

```
1 | <input type="radio" name="sex" checked>男  
2 | <input type="radio" name="sex">女
```

☒ 男  
☐ 女

```
1 | <input type="checkbox" name="vehicle">姐姐<br>  
2 | <input type="checkbox" name="vehicle">妹妹
```

☐ 姐姐  
☐ 妹妹

- range

```
1 | <input type="range" min="1" max="99" step=".5"  
   value="70">
```

- color

```
1 | <input type="color" >
```

required

如果加了这个属性，那么这个表单必须填写之后才能提交

autofocus

自动聚焦

pattern=""

使用正则表达式来匹配内容

## fieldset, legend

```
1 <form>
2   <fieldset>
3     <legend>我是legend，我是这个表单的边框信息</legend>
4     周围这一圈框框就是fieldset控制的
5     <br>
6     输入框: <input type="text" />
7   </fieldset>
8 </form>
```

## button

```
1 <button type="button">button就是一个按钮</button>
```

## select, optgroup, option

select定义选择列表（下拉列表），option就是里面的选项，optgroup则是选项的分组

```
1 <select>
2   <optgroup label="我是optgroup，我是第一组选项">
3     <option value="被送去服务器的值">option1</option>
4     <option>选项2</option>
5   </optgroup>
6   <optgroup label="我也是optgroup，我是第二组选项">
7     <option>option3</option>
8     <option>选项4</option>
9   </optgroup>
10 </select>
```

option1 ▼

## label

label标签可以让你点到某个文字就能选中选项，可以提高用户体验，这个标签需要搭配id才能使用。

```
1 <input type="radio" name="type" id="萝莉"><label for="萝莉">萝莉
  </label>
2 <input type="radio" name="type" id="御姐"><label for="御姐">御姐
  </label>
```

## textarea

cols代表每行的字符数

rows代表显示的行数

```
1 <textarea cols="30" rows="10"></textarea>
```



## datalist

在文本输入的时候，可以作为建议文本

```
1
2 <input type="text" list="test">
3 <datalist id="test">
4   <option value="1">1</option>
5   <option value="2">2</option>
6   <option value="3">3</option>
7   <option value="4">4</option>
8 </datalist>
```

## 表格

### table,td,tr,th

table用于定义表格,tr表示这是一行,td表示是一个元素.有几个tr就有几行,一个tr里面几个td就代表这行有几列.

要注意的是,第一行一般都是表格的头,所以用th修饰

```
1 <table>
2   <tr>
3     <th>th表示表格头</td>
```

```

4      <th>我也是表头</th>
5      <th>我也是</th>
6  </tr>
7  <tr>
8      <td>table表示一个表格，里面虽然啥都没有，但是可以放表格标签</td>
9      <td>一个tr就用来表示一行表格</td>
10     <td>td就表示每个单元格</td>
11 </tr>
12 <tr>
13     <td>我是第2个tr，表示第二行</td>
14 </tr>
15 </table>

```

th表示表格头	我也是表头	我也是
table表示一个表格，里面虽然啥都没有，但是可以放表格标签	一个tr就用来表示一行表格	td就表示每个单元格
我是第2个tr，表示第二行		

**注意:!!!!!!表格默认是没有边框的,这个之所以有,是因为typora自动给我加了样式.**

table的属性:

属性	值	描述
border	<i>pixels</i>	规定表格边框的宽度。
cellpadding	<ul style="list-style-type: none"> <li><i>pixels</i></li> <li>%</li> </ul>	就是表格文字和格子边框的距离
cellspacing	<ul style="list-style-type: none"> <li><i>pixels</i></li> <li>%</li> </ul>	每个单元格之间的距离
frame	<ul style="list-style-type: none"> <li>void</li> <li>above</li> <li>below</li> <li>hsides</li> <li>lhs</li> <li>rhs</li> <li>vsides</li> <li>box</li> <li>border</li> </ul>	规定外侧边框的哪个部分是可见的。
rules	<ul style="list-style-type: none"> <li>none</li> <li>groups</li> <li>rows</li> <li>cols</li> <li>all</li> </ul>	规定内侧边框的哪个部分是可见的。
width	<ul style="list-style-type: none"> <li>%</li> <li><i>pixels</i></li> </ul>	规定表格的宽度。

td,th的属性

属性	值	描述
align	<ul style="list-style-type: none"> <li>left</li> <li>right</li> <li>center</li> <li>justify</li> <li>char</li> </ul>	规定单元格内容的水平对齐方式。
axis	<i>category_name</i>	对单元格进行分类。
char	<i>character</i>	规定根据哪个字符来进行内容的对齐。
charoff	<i>number</i>	规定对齐字符的偏移量。
colspan	<i>number</i>	设置单元格可横跨的列数。
headers	<i>idrefs</i>	由空格分隔的表头单元格 ID 列表，为数据单元格提供表头信息。
rowspan	<i>number</i>	规定单元格可横跨的行数。
scope	<ul style="list-style-type: none"> <li>col</li> <li>colgroup</li> <li>row</li> <li>rowgroup</li> </ul>	定义将表头数据与单元数据相关联的方法。
valign	<ul style="list-style-type: none"> <li>top</li> <li>middle</li> <li>bottom</li> <li>baseline</li> </ul>	规定单元格内容的垂直排列方式。

## thead,tbody,tfoot

刚才那个表格其实不是完整版的,虽然也能用.但是实际开发中,能多严谨就要多严谨.

- `thead`用于表示表头
- `tbody`表示表格主题内容
- `tfoot`用于总结表的内容,当然你也随便写点啥
- **注意:这三个的顺序是thead,tfoot,tbody**

```

1  <table>
2      <thead>
3          <tr>
4              <th>妹子</th>
5              <th>特征</th>
6          </tr>
7      </thead>
8      <tfoot>
9          <tr>
10             <th>总体评价</th>
11             <th>都是我老婆</th>
12          </tr>
13      </tfoot>
14      <tbody>
15          <tr>
16              <td>牛顿</td>
17              <td>金毛双马尾</td>

```

```
18         </tr>
19         <tr>
20             <td>哈雷</td>
21             <td>大胸软萌</td>
22         </tr>
23         <tr>
24             <td>拉瓦锡</td>
25             <td>腹黑</td>
26         </tr>
27     </tbody>
28 </table>
```

妹子	特征
牛顿	金毛双马尾
哈雷	大胸软萌
拉瓦锡	腹黑
总体评价	都是我老婆

## caption

表示表格的标题

```
1  <table>
2    <caption>牛顿与苹果树鉴赏表</caption>
3    <thead>
4        <tr>
5            <th>妹子</th>
6            <th>特征</th>
7        </tr>
8    </thead>
9    <tbody>
10       <tr>
11           <td>牛顿</td>
12           <td>金毛双马尾</td>
13       </tr>
14       <tr>
15           <td>哈雷</td>
16           <td>大胸软萌</td>
17       </tr>
18       <tr>
19           <td>拉瓦锡</td>
20           <td>腹黑</td>
21       </tr>
22   </tbody>
23 </table>
```

牛顿与苹果树鉴赏表

妹子	特征
牛顿	金毛双马尾
哈雷	大胸软萌
拉瓦锡	腹黑

# 文本

## kbd

用于描述键盘上的按键

```
1 | <kbd>quit</kbd>
```



## abbr

用于定义缩写,并且鼠标滞留时还可以有提示

```
1 | 镜5莲华下海了,<abbr title="爷的青春结束了">爷青结</abbr>.
```

镜5莲华下海了,爷青结.

## blockquote

```
1 | blockquote是一个长引用，
2 | <blockquote>
3 |     它里面的内容会根据编译器自动排版，表示一个段落的引用
4 | </blockquote>
```

blockquote是一个长引用，

它里面的内容会根据编译器自动排版，表示一个段落的引用

## cite

```
1 | <cite>cite语义表示引用，可以表示书名，电影什么的，表现为斜体</cite>
```

cite语义表示引用，可以表示书名，电影什么的，表现为斜体

## ruby, rt, rb

```
1 ruby用来表示
2 <ruby>注<rt>zhù</rt></ruby>
3 <ruby>音<rt>yīn</rt></ruby>。
4 ruby里面表示要注音的内容，rt里面表示拼音<br>
5 若浏览器不支持，就需要用rp了，<br>
6 rp标签在 ruby 注释中使用，以定义不支持 ruby 元素的浏览器所显示的内容。<br>
7 我是<ruby>注音<rt><rp>(</rp>zhù yīn<rp></rp></rt></ruby>
```

ruby用来表示<sup>zhù yīn</sup>注音。

ruby里面表示要注音的内容，rt里面表示拼音,若浏览器不支持，就需要用rp了，rp标签在 ruby 注释中使用，以定义不支持 ruby 元素的浏览器所显示的内容。

我是注音zhù yīn.

## ins, u

```
1 <ins>这个表示下划线，填空题经常会看到</ins>
2 <u>u也有同样的下划线</u>
```

这个表示下划线，填空题经常会看到

u也有同样的下划线

## del

```
1 <del>del里面的内容会被加入删除线，语义上就是删掉的内容</del>
```

~~del里面的内容会被加入删除线，语义上就是删掉的内容~~

## em, i

```
1 <em>em语义上表示强调语气，表现为斜体</em>
2 <i>i一般表示成语，一些关键术语之类的</i>
```

*em语义上表示强调语气，表现为斜体*

*i一般表示成语，一些关键术语之类的*

## pre

```
1 <pre>
2 pre 元素可定义预格式化的文本。被包围在 pre 元素中的文本通常会保留空格和换行符。而文本也会呈现为等宽字体。
3 说白了就是，pre里面怎么写，浏览器就怎么显示，因为一般浏览器会忽略多个回车和空格
4 int main()
5 {
6     return 0;
7 }
8 但是注意标签不能放进去，有的不会显示
9 </pre>
```



**pre** 元素可定义预格式化的文本。被包围在 **pre** 元素中的文本通常会保留空格和换行符。而文本也会呈现为等宽字体。  
说白了就是，**pre**里面怎么写，浏览器就怎么显示，因为一般浏览器会忽略多个回车和空格

```
int main()
{
    return 0;
}
```

但是注意标签不能放进去,有的不会显示

## q

```
1 | <q>q标签表示短引用，一般语义上表示说话之类的，表现为在标签里面的内容，自动被
   | 加了双引号 </q>
2 | <p>比如说，子曰:<q>知之为知之。</q></p>
3 | <p>这个引号并不是HTML加的，所以鼠标也不能选中</p>
```

"q标签表示短引用，一般语义上表示说话之类的，表现为在标签里面的内容，自动被加了双引号"

比如说，子曰:"知之为知之。"

这个引号并不是HTML加的，所以鼠标也不能选中

## strong

```
1 | <strong>strong表示强烈强调，语气上非常激动，表现为黑体</strong>
```

**strong表示强烈强调，语气上非常激动，表现为黑体**

## bdo

可以将文本反向，dir设置rtl就是从右到左，ltr就是从左到右

```
1 | <bdo dir="rtl">12345</bdo>
```

54321

## bdi

bdi内的文本可以脱离父元素的文本方向。

```
1 | <bdo dir="rtl">01234<bdi>56789<bdi></bdo>
   | <br/>
3 | <bdo dir="rtl">01234<span>56789<span></bdo>
```

第一行首先从右开始，发现是bdi标签，里面的内容脱离父元素的文本方向，然后正常输出56789，之后逆向输出01234

第二行则一直逆向输出。

98765432105678943210

## mark

可以让被mark的文本高亮显示

```
1 | <p><mark>高亮</mark>显示</p>
```

高亮显示

## sub

```
1 | sub表示 <sub>下标</sub>, 经常用于x<sub>1</sub>之类的数学符号
```

sub表示 下标, 经常用于 $x_1$ 之类的数学符号

## sup

```
1 | sup可以用来表示 <sup>上标</sup>, 最常见就是y=x<sup>2</sup>
```

sup可以用来表示 上标, 最常见就是 $y=x^2$

## progress

超级强大的标签, 可以显示进度。

max代表最大值, value代表目前的进度。不用加单位, 直接写数字就行了。

```
1 | <progress value="90" max="100"></progress>
```



## small

通常用于免责声明、条款等。

# 媒体

## audio

- src: 音乐的相对路径, 可以是本地的url也可以是服务器的地址
- controls: 显示播放器
- autoplay: 自动播放, 必须加上muted才能自动播放
- preload: 何时进行预加载
  - auto: 浏览器自动决定
  - metadata: 浏览器不会缓存, 但是对于音轨和时长这样的元数据会进行预加载
  - none: 不进行预加载, 直到用户激活控件
- muted: 设置静音
- loop: 循环播放
- poster 设置封面图像

```
1 <audio src="001.mp3" controls autoplay loop preload="auto">您的浏览器不支持 audio 标签。</audio>
```

有的格式浏览器并不支持mp3，所以需要使用source标签来匹配多种格式的音源

```
1 <audio controls preload="metadata" >
2   <source src="./audio/bgm/晨曦.mp3">
3   <source src="./audio/bgm/晨曦.ogg">
4 </audio>
```

## img

- src: 图片的相对路径
- alt: 如果图片加载失败，就会显示里面的文字

```
1 <img src="" alt="加载失败!">
```

## 事件属性

一般都放在body里面。

```
1 <body onload="load()">
```

## 自定义属性

在H5中，如果想要设置自定义属性，需要用 data- 开头，而且必须赋值

## 通用属性

- hidden  
让标签隐藏，但是要注意，这个玩意实际上就是display: none
- contenteditable  
让文档可以编辑
- draggable="true"  
让元素可以拖拽

