

HTML 介绍

1. 什么是 HTML

HTML 的全称为 Hyper Text Markup Language，译为超文本标记语言。

超文本，就是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，是一种将文本以及文本相关的其他信息结合起来，展现出关于文档结构和数据处理细节的电脑文字编码。与文本相关的其他信息（包括例如文本的结构和表示信息等）与原来的文本结合在一起，但是使用标记进行标识。

2. 发展历史

1980年，蒂姆·伯纳斯-李为使世界各地的物理学家能够方便的进行合作研究，创建了使用于其系统的 HTML。Tim Berners-Lee 设计的 HTML 以纯文字格式为基础，可以使用任何文本编辑器处理，最初仅有少量标记（TAG）而易于掌握运用。随着 HTML 使用率的增加，人们不满足只能看到文字。

1993年，还是大学生的马克·安德生在他的 Mosaic 浏览器加入 `` 标记，从此可以在 Web 页面上浏览图片。但人们认为仅有图片还是不够，希望可将任何形式的媒体加到网页上。因此 HTML 不断地扩充和发展。

3. 标准的版本历史

- 超文本标记语言（第一版）—— 在 1993 年 6 月作为互联网工程工作小组（IETF）工作草案发布（并非标准）
- HTML 2.0 —— 1995 年 11 月作为 RFC 1866 发布，在 RFC 2854 于 2000 年 6 月发布之后被宣布已经过时
- HTML 3.2 —— 1997 年 1 月 14 日，W3C 推荐标准
- HTML 4.0 —— 1997 年 12 月 18 日，W3C 推荐标准
- HTML 4.01（微小改进）—— 1999 年 12 月 24 日，W3C 推荐标准
- HTML 5 的第一份正式草案已于 2008 年 1 月 22 日公布，仍继续完善

第一个 HTML 页面

4. 示例代码

```
1 <html>
2   <head>
3     <title>01_helloworld.html</title>
4   </head>
5   <body>
6     <!-- 这是第一个页面 helloworld -->
7     <font color="red">hello world!</font>
8   </body>
9 </html>
```

示例代码说明:

- html 标签: 定义 HTML 文档, HTML 文档的根标签。
- head 标签: 定义关于文档的基本信息。
- body 标签: 定义文档的主体, 运行 HTML 文档时显示的内容。
- title 标签: 定义文档的标题。在运行网页时, 页面顶端显示的内容。
- font 标签: 定义文字的字体、尺寸和颜色。

5. HTML元素 (标签)

HTML 标记标签通常被称为 HTML 标签, "HTML 标签" 和 "HTML 元素" 通常都是描述同样的意思。(一个 HTML 元素包含了开始标签与结束标签)

- HTML 标签是由尖括号包围的关键词, 比如 `<html>`
- HTML 标签通常是成对出现的, 比如 `` 和 ``
- 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
- 开始和结束标签也被称为开放标签和闭合标签

值得注意的是:

不要忘记结束标签:

```
1 | <font color="red">hello world!
```

- 即使忘记了使用结束标签, 大多数浏览器也会正确地显示 HTML。
- HTML 文档中, 关闭标签是可选的。
- 不要依赖这种做法。忘记使用结束标签会产生不可预料的结果或错误。

HTML 空元素:

- 没有内容的 HTML 元素被称为空元素。空元素是在开始标签中关闭的。
- `
` 就是没有关闭标签的空元素 (`
` 标签定义换行)。
- 在 XHTML、XML 以及未来版本的 HTML 中, 所有元素都必须被关闭。

- 在开始标签中添加斜杠，比如 `
`，是关闭空元素的正确方法，HTML、XHTML 和 XML 都接受这种方式。
- 即使 `
` 在所有浏览器中都是有效的，但使用 `
` 其实是更长远的保障。

推荐使用小写标签：

- HTML 标签对大小写不敏感：`<P>` 等同于 `<p>`。许多网站都使用大写的 HTML 标签。
- 万维网联盟（W3C）在 HTML 4 中推荐使用小写，而在未来 (X)HTML 版本中强制使用小写。

6. HTML 属性

```
1 | <font color="red">hello world!</font>
```

属性是 HTML 元素提供的附加信息。

- HTML 元素可以设置属性。
- 属性可以在元素中添加附加信息。
- 属性一般描述于开始标签。
- 属性总是以名称/值对的形式出现，比如：`name="value"`。

值得注意的是：

HTML 属性常用引用属性值：

- 属性值应该始终被包括在引号内。
- 双引号是最常用的，不过使用单引号也没有问题。在某些个别的情况下，比如属性值本身就含有双引号，那么必须使用单引号。

推荐使用小写属性：

- 属性和属性值对大小写不敏感。
- 万维网联盟在其 HTML 4 推荐标准中推荐小写的属性/属性值。
- 新版本的 (X)HTML 要求使用小写属性。

7. HTML 注释

```
1 | <!-- 这是一个注释 -->
```

- 开始括号之后（左边的括号）需要紧跟一个叹号，结束括号之前（右边的括号）不需要。
- 浏览器会忽略注释，也不会显示它们。
- 将注释插入 HTML 代码中，这样可以提高其可读性，使代码更易被人理解。
- 合理地使用注释可以对未来的代码编辑工作产生帮助。

8. HTML 文档

```
1 | HTML文档 = 网页
```

- HTML文档描述网页
- HTML文档包含HTML标签和纯文本
- HTML文档也被称为网页
- HTML文档文件的后缀名为“.html”和“.htm”，两种后缀名没有任何区别

9. HTML 基本结构

```
1 | <html>
2 |     <head>
3 |     </head>
4 |     <body>
5 |     </body>
6 | </html>
```

10. <!DOCTYPE> 声明

<!DOCTYPE> 声明有助于浏览器中正确显示网页。网络上有很多不同的文件，如果能够正确声明 HTML 的版本，浏览器就能正确显示网页内容。

- <!DOCTYPE> 声明位于文档中的最前面的位置，处于 标签之前。
- <!DOCTYPE> 声明不是一个 HTML 标签；它是用来告知 Web 浏览器页面使用了哪种 HTML 版本。
- 总是给您的 HTML 文档添加 <!DOCTYPE> 声明，确保浏览器能够预先知道文档类型。

各个HTML版本的<!DOCTYPE> 声明应如何书写：

- HTML 5

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

- HTML 4.01 Strict

这个 DTD 包含所有 HTML 元素和属性，但不包括表象或过时的元素（如 font）。框架集是不允许的。

```
1 | <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.d
```

- HTML 4.01 Transitional

这个 DTD 包含所有 HTML 元素和属性，包括表象或过时的元素（如 font）。框架集是不允许的。

```
1 | <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/h
```

- HTML 4.01 Frameset

这个 DTD 与 HTML 4.01 Transitional 相同，但是允许使用框架集内容。

```
1 | <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4
```

11. HTML 头部

<head> 元素

<head> 元素包含了所有的头部标签元素。在 <head> 元素中你可以插入脚本（scripts），样式文件（CSS），及各种 meta 信息。

可以添加在头部区域的元素标签为：<title>，<style>，<meta>，<link>，<script>，<noscript>，and <base>。

<title> 元素

<title> 标签定义了不同文档的标题。<title> 在 HTML/XHTML 文档中是必须的。

<base> 元素

<base> 标签描述了基本的链接地址/链接目标，该标签作为 HTML 文档中所有的链接标签的默认链接。

<link> 元素

<link> 标签定义了一个文档和外部资源之间的关系。

<style> 元素

<style> 标签定义了HTML文档的样式文件引用地址。

<meta> 元素

meta 标签描述了一些基本的元数据。<meta> 标签提供了元数据。元数据也不显示在页面上，但会被浏览器解析。

meta 元素通常用于指定网页的描述，关键词，文件的最后修改时间，作者，和其他元数据。元数据可以使用浏览器（如何显示内容或重新加载页面），搜索引擎（关键词），或其他 Web 服务。

<meta> 标签使用实例

- 为搜索引擎定义关键词:

```
1 | <meta name="keywords" content="HTML, CSS, XML, XHTML, JavaScript">
```

- 为网页定义描述内容:

```
1 | <meta name="description" content="Free Web tutorials on HTML and CSS">
```

- 定义网页作者:

```
1 | <meta name="author" content="King">
```

- 每30秒中刷新当前页:

```
1 | <meta http-equiv="refresh" content="30">
```

HTML 基本元素

12. 示例代码

```
1 | <!DOCTYPE html>
2 | <html>
3 |   <head>
4 |     <title>06_基本元素实例.html</title>
5 |     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
6 |   </head>
7 |   <body>
8 |     <h1 align="center">春晓</h1>
9 |     <hr align="center" width="30%">
10 |    <p align="center">
11 |      春眠不觉晓, <br>
12 |      处处闻啼鸟。<br>
13 |      夜来风雨声, <br>
14 |      花落知多少。<br>
15 |    </p>
16 |  </body>
17 | </html>
```

13. HTML 标题

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>02_标题元素.html</title>
5     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
6   </head>
7   <body>
8     <h1>这是标题 1</h1>
9     <h2>这是标题 2</h2>
10    <h3>这是标题 3</h3>
11    <h4>这是标题 4</h4>
12    <h5>这是标题 5</h5>
13    <h6>这是标题 6</h6>
14  </body>
15 </html>

```

在 HTML 文档中，标题很重要。

- 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。
- 浏览器会自动地在标题的前后添加空行。
- 用户可以通过标题来快速浏览您网页。

值得注意的是：

- 标题（Heading）在浏览器运行显示的效果并不重要。（后面可以通过 CSS 进行修改）
- 标题（Heading）更重要的在于其语义化。
- 搜索引擎检索 HTML 页面时，`<h1>` 元素仅次于 `<title>` 元素。
- 一般一个 HTML 页面中只包含一个 `<h1>` 元素。

14. 段落

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>03_段落元素.html</title>
5     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
6   </head>
7   <body>
8     <p>这是一个段落。</p>
9     <p>这是一个段落。</p>
10    <p>这是一个段落。</p>
11  </body>
12 </html>

```

- 段落是通过 `<p>` 标签定义的。
- 浏览器会自动地在段落的前后添加空行。

值得注意的是:不要忘记结束标签。

- 即使忘了使用结束标签,大多数浏览器也会正确地将 HTML 显示出来,但不要依赖这种做法。
- 忘记使用结束标签会产生意想不到的结果和错误。
- 在未来的 HTML 版本中,不允许省略结束标签。

15. 换行

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>04_换行元素.html</title>
5     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
6   </head>
7   <body>
8     <p>This is<br>a para<br>graph with line breaks</p>
9   </body>
10 </html>
```

- 在不产生一个新段落的情况下进行换行（新行），请使用 `
` 标签。
- `
` 元素是一个空的 HTML 元素。
- 由于关闭标签没有任何意义，因此它没有结束标签。

16. 水平线

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>05_水平线元素.html</title>
5     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
6   </head>
7   <body>
8     <p>hr 标签定义水平线: </p>
9     <hr />
10    <p>这是段落。</p>
11    <hr />
12    <p>这是段落。</p>
13    <hr />
14    <p>这是段落。</p>
15  </body>
16 </html>
```


- `<hr>` 标签在 HTML 页面中创建水平线。
- `hr` 元素可用于分隔内容。

列表

17. 有序列表

有序列表也是一列项目，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。

`` 标签定义了一个有序列表. 列表排序以数字来显示。

| 属性名 | 值 | 描述 |
|-------|-----------|--|
| start | number | HTML5不支持，不赞成使用。 请使用样式取代它。规定列表中的起始点。 |
| type | 1 A a l i | 规定列表的类型。 |

`` 标签定义列表项目。

| 属性名 | 值 | 描述 |
|-------|-----------|--|
| value | number | HTML5不支持，不赞成使用。 请使用样式取代它。规定列表项目的数字。 |
| type | 1 A a l i | 规定列表的类型。 |

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>13_有序列表.html</title>
5     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
6   </head>
7   <body>
8     <h4>Numbered list:</h4>
9     <ol>
10      <li>Apples</li>
11      <li>Bananas</li>
12      <li>Lemons</li>
13      <li>Oranges</li>
14    </ol>
15    <h4>Letters list:</h4>
16    <ol type="A">
17      <li>Apples</li>
18      <li>Bananas</li>
19      <li>Lemons</li>
20      <li>Oranges</li>
21    </ol>
22    <h4>Lowercase letters list:</h4>
23    <ol type="a">
24      <li>Apples</li>
25      <li>Bananas</li>
26      <li>Lemons</li>
27      <li>Oranges</li>
28    </ol>
29    <h4>Roman numbers list:</h4>
30    <ol type="I">
31      <li>Apples</li>
32      <li>Bananas</li>
33      <li>Lemons</li>
34      <li>Oranges</li>
35    </ol>
36    <h4>Lowercase Roman numbers list:</h4>
37    <ol type="i">
38      <li>Apples</li>
39      <li>Bananas</li>
40      <li>Lemons</li>
41    </ol>
42  </body>
43 </html>
```

18. 无序列表

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。

**** 标签定义无序列表。

| 属性名 | 值 | 描述 |
|-------|--------------------|--|
| start | number | HTML5不支持，不赞成使用。 请使用样式取代它。规定列表中的起始点。 |
| type | disc/square/circle | 规定列表的类型。 |

**** 标签定义列表项目。

| 属性名 | 值 | 描述 |
|-------|-----------|--|
| value | number | HTML5不支持，不赞成使用。 请使用样式取代它。规定列表项目的数字。 |
| type | 1 A a l i | 规定列表的类型。 |

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>14_无序列表.html</title>
5     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
6   </head>
7   <body>
8     <h4>Disc bullets list:</h4>
9     <ul style="list-style-type:disc">
10      <li>Apples</li>
11      <li>Bananas</li>
12      <li>Lemons</li>
13      <li>Oranges</li>
14    </ul>
15
16    <h4>Circle bullets list:</h4>
17    <ul style="list-style-type:circle">
18      <li>Apples</li>
19      <li>Bananas</li>
20      <li>Lemons</li>
21      <li>Oranges</li>
22    </ul>
23
24    <h4>Square bullets list:</h4>
25    <ul style="list-style-type:square">
26      <li>Apples</li>
27      <li>Bananas</li>
28      <li>Lemons</li>
29      <li>Oranges</li>
30    </ul>
31  </body>
32 </html>
```

19. 嵌套列表

有序嵌套列表

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>15_嵌套列表.html</title>
5     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
6   </head>
7   <body>
8     <h4>有序嵌套列表:</h4>
9     <ol>
10      <li>咖啡</li>
11      <li>茶
12        <ol type="a">
13          <li>红茶</li>
14          <li>绿茶
15            <ol type="i">
16              <li>中国</li>
17              <li>非洲</li>
18            </ol>
19          </li>
20        </ol>
21      </li>
22      <li>牛奶</li>
23    </ol>
24  </body>
25 </html>
```

无序嵌套列表

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>15_嵌套列表.html</title>
5     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
6   </head>
7   <body>
8     <h4>无序嵌套列表:</h4>
9     <ul>
10      <li>咖啡</li>
11      <li>茶
12        <ul>
13          <li>红茶</li>
14          <li>绿茶
15            <ul>
16              <li>中国</li>
17              <li>非洲</li>
18            </ul>
19          </li>
20        </ul>
21      </li>
22      <li>牛奶</li>
23    </ul>
24  </body>
25 </html>
```

20. 定义列表

定义列表不仅仅是一列项目，而是项目及其注释的组合。定义列表以 `<dl>` 标签开始。每个定义列表项以 `<dt>` 开始。每个定义列表项的定义以 `<dd>` 开始。

- `<dl>` 标签定义一个描述列表。
- `<dt>` 标签定义一个描述列表的项目/名字。
- `<dd>` 标签被用来对一个描述列表中的项目/名字进行描述。

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>16_自定义列表.html</title>
5     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
6   </head>
7   <body>
8     <dl>
9       <dt>
10        平时爱好:
11      </dt>
12      <dd>- 抽烟</dd>
13      <dd>- 喝酒</dd>
14      <dd>- 烫头</dd>
15      <dt>
16        喜欢的游戏:
17      </dt>
18      <dd>- 魔兽世界</dd>
19      <dd>- 反恐精英</dd>
20      <dd>- 红色警戒</dd>
21    </dl>
22  </body>
23 </html>
```

链接元素

HTML 使用超级链接与网络上的另一个文档相连。几乎可以在所有的网页中找到链接。点击链接可以从一张页面跳转到另一张页面。

- 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或者当前文档中的某个部分。
- 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。
- 在标签 `<a>` 中使用了 href 属性来描述链接的地址。

值得注意的是：

- 如果没有使用 href 属性，则不能使用 hreflang、media、rel、target 以及 type 属性。
- 通常在当前浏览器窗口中显示被链接页面，除非规定了其他 target。
- 请使用 CSS 来改变链接的样式。

`<a>` 元素的属性列表：

| 属性名 | 值 | 描述 |
|--------|---------------------------|--------------------------------|
| href | URL | 规定链接的目标 URL。 |
| target | <i>blank/parent/_self</i> | 规定在何处打开目标 URL。仅在 href 属性存在时使用。 |

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>12_链接元素.html</title>
5     <meta http-equiv="content-type" content="text/html; charset=UTF-8">
6   </head>
7   <body>
8     <a name="top">这是顶端</a><br>
9     <a href="mailto:82328769@qq.com" >联系我们</a><br>
10    <br>
11    <a name="middle">这是中间</a><br>
12    <a href="#top" >回到顶部</a><br>
13    <a href="#middle" >回到中间</a><br>
14  </body>
15 </html>
```

版权说明

本笔记的内容免费开源，任何人都可以免费学习、分享，甚至可以进行修改。但需要注明作者及来源，并且不能用于商业。

本笔记采用[知识共享署名-非商业性使用-禁止演绎 4.0 国际许可协议](#)进行许可。

联系方式



可以问职业问题。
不要问技术问题。



可以问职业问题。
可以问技术问题。



可以问职业问题。
可以问技术问题。



