**使用正确的文档类型**

文档类型声明位于HTML文档的第一行：

<!DOCTYPE html>

如果你想跟其他标签一样使用小写，可以使用以下代码：

<!doctype html>

**使用小写元素名**

HTML5 元素名可以使用大写和小写字母。

推荐使用小写字母：

* 混合了大小写的风格是非常糟糕的。
* 开发人员通常使用小写 (类似 XHTML)。
* 小写风格看起来更加清爽。
* 小写字母容易编写。

**不推荐:**

<SECTION>   
  <p>这是一个段落。</p>  
</SECTION>

**非常糟糕:**

<Section>   
  <p>这是一个段落。</p>  
</SECTION>

**推荐:**

<section>   
  <p>这是一个段落。</p>  
</section>

**关闭所有 HTML 元素**

在 HTML5 中, 你不一定要关闭所有元素 (例如 <p> 元素)，但我们建议每个元素都要添加关闭标签。

不推荐:

<section>  
  <p>这是一个段落。  
  <p>这是一个段落。  
</section>

推荐:

<section>  
  <p>这是一个段落。</p>  
  <p>这是一个段落。</p>  
</section>

**关闭空的 HTML 元素**

在 HTML5 中, 空的 HTML 元素也不一定要关闭：

我们可以这么写：

<meta charset="utf-8">

也可以这么写：

<meta charset="utf-8" />

在 XHTML 和 XML 中斜线 (/) 是必须的。

如果你期望 XML 软件使用你的页面，使用这种风格是非常好的。

**使用小写属性名**

HTML5 属性名允许使用大写和小写字母。

我们推荐使用小写字母属性名:

* 同时使用大小写是非常不好的习惯。
* 开发人员通常使用小写 (类似 XHTML)。
* 小写风格看起来更加清爽。
* 小写字母容易编写。

不推荐：

<div CLASS="menu">

推荐：

<div class="menu">

**属性值**

HTML5 属性值可以不用引号。

属性值我们推荐使用引号:

* 如果属性值含有空格需要使用引号。
* 混合风格不推荐的，建议统一风格。
* 属性值使用引号易于阅读。

以下实例属性值包含空格，没有使用引号，所以不能起作用:

<table class=table striped>

以下使用了双引号，是正确的：

<table class="table striped">

**图片属性**

图片通常使用 **alt** 属性。 在图片不能显示时，它能替代图片显示。

<img src="html5.gif" **alt="HTML5"** style="width:128px;height:128px">

定义好图片的尺寸，在加载时可以预留指定空间，减少闪烁。

<img src="html5.gif" alt="HTML5" **style="width:128px;height:128px**">

**空格和等号**

等号前后可以使用空格。

<link rel = "stylesheet" href = "styles.css">

但我们推荐少用空格:

<link rel="stylesheet" href="styles.css">

**避免一行代码过长**

使用 HTML 编辑器，左右滚动代码是不方便的。

每行代码尽量少于 80 个字符。

**空行和缩进**

不要无缘无故添加空行。

为每个逻辑功能块添加空行，这样更易于阅读。

缩进使用两个空格，不建议使用 TAB。

比较短的代码间不要使用不必要的空行和缩进。

**不必要的空行和缩进:**

<body>  
  
  <h1>菜鸟教程</h1>  
  
  <h2>HTML</h2>  
  
  <p>  
    菜鸟教程，学的不仅是技术，更是梦想。  
    菜鸟教程，学的不仅是技术，更是梦想。  
   菜鸟教程，学的不仅是技术，更是梦想,  
    菜鸟教程，学的不仅是技术，更是梦想。  
  </p>  
  
</body>

**推荐:**

<body>  
  
<h1>菜鸟教程</h1>  
  
<h2></h2>  
<p>菜鸟教程，学的不仅是技术，更是梦想。  
菜鸟教程，学的不仅是技术，更是梦想。  
菜鸟教程，学的不仅是技术，更是梦想。  
菜鸟教程，学的不仅是技术，更是梦想。</p>  
  
</body>

**表格实例:**

<table>  
  <tr>  
    <th>Name</th>  
    <th>Description</th>  
  </tr>  
  <tr>  
    <td>A</td>  
    <td>Description of A</td>  
  </tr>  
  <tr>  
    <td>B</td>  
    <td>Description of B</td>  
  </tr>  
</table>

**列表实例:**

<ol>  
  <li>London</li>  
  <li>Paris</li>  
  <li>Tokyo</li>  
</ol>

**省略 <html> 和 <body>?**

在标准 HTML5 中， <html> 和 <body> 标签是可以省略的。

以下 HTML5 文档是正确的:

**实例:**

<!DOCTYPE html>  
<head>  
  <title>页面标题</title>  
</head>  
  
<h1>这是一个标题</h1>  
<p>这是一个段落。</p>

**不推荐省略 <html> 和 <body> 标签。**

<html> 元素是文档的根元素，用于描述页面的语言：

<!DOCTYPE html>  
<html lang="zh">

声明语言是为了方便屏幕阅读器及搜索引擎。

省略 <html> 或 <body> 在 DOM 和 XML 软件中会崩溃。

省略 <body> 在旧版浏览器 (IE9)会发生错误。

**省略 <head>?**

在标准 HTML5 中， <head>标签是可以省略的。

默认情况下，浏览器会将 <body> 之前的内容添加到一个默认的 <head> 元素上。

**实例**

<!DOCTYPE html>  
<html>  
<title>页面标题</title>  
  
<body>  
  <h1>这是一个标题</h1>  
  <p>这是一个段落。</p>  
</body>  
  
</html>

|  |  |
| --- | --- |
| **Note** | 现在省略 head 标签还不推荐使用。 |

**元数据**

HTML5 中 <title> 元素是必须的，标题名描述了页面的主题:

<title>菜鸟教程</title>

标题和语言可以让搜索引擎很快了解你页面的主题:

<!DOCTYPE html>  
<html lang="zh">  
<head>  
  <meta charset="UTF-8">  
  <title>菜鸟教程</title>  
</head>

**HTML 注释**

注释可以写在 <!-- 和 --> 中:

<!-- 这是注释 -->

比较长的评论可以在 <!-- 和 --> 中分行写：

<!--   
  这是一个较长评论。 这是 一个较长评论。这是一个较长评论。  
  这是 一个较长评论 这是一个较长评论。 这是 一个较长评论。  
-->

长评论第一个字符缩进两个空格，更易于阅读。

**样式表**

样式表使用简洁的语法格式 ( type 属性不是必须的):

<link rel="stylesheet" href="styles.css">

短的规则可以写成一行:

p.into {font-family: Verdana; font-size: 16em;}

长的规则可以写成多行:

body {  
  background-color: lightgrey;  
  font-family: "Arial Black", Helvetica, sans-serif;  
  font-size: 16em;  
  color: black;  
}

* 将左花括号与选择器放在同一行。
* 左花括号与选择器间添加一个空格。
* 使用两个空格来缩进。
* 冒号与属性值之间添加一个空格。
* 逗号和符号之后使用一个空格。
* 每个属性与值结尾都要使用分号。
* 只有属性值包含空格时才使用引号。
* 右花括号放在新的一行。
* 每行最多 80 个字符。

|  |  |
| --- | --- |
| **Note** | 在逗号和冒号后添加空格是常用的一个规则。 |

**在 HTML 中载入 JavaScript**

使用简洁的语法来载入外部的脚本文件 ( type 属性不是必须的 ):

<script src="myscript.js">

**使用 JavaScript 访问 HTML 元素**

一个糟糕的 HTML 格式可能会导致 JavaScript 执行错误。

以下两个 JavaScript 语句会输出不同结果:

**实例**

var obj = getElementById("Demo")  
  
var obj = getElementById("demo")

HTML 中 JavaScript 尽量使用相同的命名规则。

**使用小写文件名**

大多 Web 服务器 (Apache, Unix) 对大小写敏感： london.jpg 不能通过 London.jpg 访问。

其他 Web 服务器 (Microsoft, IIS) 对大小写不敏感： london.jpg 可以通过 London.jpg 或 london.jpg 访问。

你必须保持统一的风格，我们建议统一使用小写的文件名。

**文件扩展名**

HTML 文件后缀可以是 **.html** (或 **.htm**)。

CSS 文件后缀是 **.css** 。

JavaScript 文件后缀是 **.js** 。

**.htm 和 .html 的区别**

.htm 和 .html 的扩展名文件本质上是没有区别的。浏览器和 Web 服务器都会把它们当作 HTML 文件来处理。

区别在于：

.htm 应用在早期 DOS 系统，系统现在或者只能有三个字符。

在 Unix 系统中后缀没有特别限制，一般用 .html。

**技术上区别**

如果一个 URL 没有指定文件名 (如 http://www.runoob.com/css/), 服务器会返回默认的文件名。通常默认文件名为 index.html, index.htm, default.html, 和 default.htm。

如果服务器只配置了 "index.html" 作为默认文件，你必须将文件命名为 "index.html", 而不是"index.htm"。

但是，通常服务器可以设置多个默认文件，你可以根据需要设置默认文件名。

不管怎样，HTML 完整的后缀是 ".html"。