

W3C 代码标准规范

W3C通过设立领域（Domains）和标准计划（Activities）来组织W3C的标准活动，围绕每个标准计划，会设立相关的W3C工作组织（包括工作组、社区组、商务组等）。W3C会根据产业界的标准需求调整Domains和Activity的设置及相关的工作组设置。截至2014年7月，W3C共设立5个技术领域，开展24个标准计划。查看所有工作组列表（英文）。

w3cinteraction domain 交互技术标准领域

主要关注Web与用户交互的接口和技术标准，包括(X)HTML及其他用于Web的标记语言（如CSS, MathML, SMIL, SVG, XForms等），同时提供技术途径将这些技术集成到客户端环境中。下设10个标准计划：

- 图形（Graphics Activity）
- HTML（HTML Activity）
- 国际化（Internationalization Activity）
- 数学表示（Math Activity）
- 富Web客户端（Rich Web Client Activity）
- 样式（Style Activity）级联样式单（CSS）
- Web视频（Video in the Web Activity）媒体标记（Annotations）、媒体片段、时间序列文本（Timed Text）。
- XForms（XForms Activity），电子表单（Forms）。
- Web字体（Fonts Activity），Web字体（WebFonts）。
- Web测试（Web Testing Activity），浏览器测试和工具、Web测试兴趣。

w3cubiquitous web domain 普适Web标准领域

主要关注让任何人在任何地点、任何时间，通过任何设备访问Web内容的技术和标准，包括如何通过移动重点及其他消费电子、交互式电视、车载终端等获取Web内容。2013年11月，W3C调整技术领域分工，可扩展标记语言（XML）相关标准化工作划入信息与知识技术领域。目前，该领域下设5个标准计划：

- 移动Web计划（Mobile Web Initiative Activity），语音API社区组、响应式图片社区组及核心移动，新设立Web和移动。
- 多模式交互（Multimodal Interaction Activity），多模式交互。
- 普适Web应用（Ubiquitous Web Applications Activity），地理定位（Geolocation）、系统应用、基于模型的用户交互（Model-based UI）、设备API及策略、Web实时通讯（WebRTC）、近场通讯。
- 语音浏览器（Voice Browser Activity），语音浏览器。
- Web和电视（Web and TV Activity），Web和电视。

w3cTechnology and Society Domain 技术与社会标准领域

主要解决Web技术和公共策略相关的技术问题，2013年11月，W3C调整了技术与社会（Technology and Society）技术领域的覆盖范围，重点推进Web的隐私保护、安全，以及未来与社会化Web、数字营销（Digital Marketing）等相关的标准工作。鉴于Web服务相关标准制定工作已经完成，W3C关闭了原来的Web服务标准计划。2014年7月，W3C在该技术领域下新设立社会网络标准计划（Social Activity）。该技术领域由Wendy Seltzer负责，目前下设4个标准计划：

- 专利策略（Patent Policy Activity），专利和标准。

- 隐私 (Privacy Activity) , 追踪保护、隐私。
- 安全 (Security Activity) , XML安全、Web应用安全、Web加密、Web安全新。
- 社交网络 (Social Activity) : 2014年7月新设立, 下设社交网络、社交网络。

w3c 信息与知识 (Information & Knowledge) 标准领域

2013年11月, W3C正式将原来的语义Web、Web数据 (Data on the Web)、数字出版, 以及XML相关的标准化工作合并, 设立新的信息与知识技术领域 (Information and Knowledge), 并任命Ralph Swick担任该技术领域的负责人。Ralph同时还兼任了W3C的首席运营官。

- 数据 (Data Activity) , 整合了原来的语义Web标准计划, 及电子政府标准计划, 数据标准计划协调、RDFa、语义Web健康和生命科学、语义Web、RDF工作组、链接数据平台工作组、政府链接数据工作组, 此外, 新设立了Web数据最佳实践工作组 (Data on the Web Best Practices WG)、Web上的逗号分隔值 (CSV on the Web) 。
- 数字出版 (Digital Publishing Activity) , 数字出版。
- 可扩展标记语言XML (Extensible Markup Language Activity) , 高效XML交换、XML协调组、XML核心、XML处理模型、XML查询、XML大纲、XSLT。

w3cwai Web信息无障碍计划标准

主要解决残障人士方便的获得Web内容相关的技术问题并形成解决方案, 包括Web信息无障碍的基本原则、实现技术、评估方法等。下设2个标准计划:

- Web信息无障碍技术 (WAI Technical Activity) , 协议和格式 (PF)、Web内容可访问性原则 (WCAG)、编写工具可访问性原则、用户代理可访问性、评估和修复工具 (ERT) , 及独立用户界面 (Indie UI) 。
- WAI国际计划办公室 (WAI International Program Office Activity) , 教育和推广 (EO)、研究和开发 (RD)、WAI、WAI。

一、目的: 为什么要遵循标准

我们作为生产者实际上只是位于中游, 既不是上游的浏览器制造商, 他们是标准的真正制定者, 也不算是下游, 他们是浏览器的终端使用者。这个角色就意味着我们位于一个接口的位置, 我们需要想办法满足下游的用户使用上游不同浏览器时看到的是同样的内容。用一个程序的语言说, 我们是转换器.....adapter, 我们想方设法让我们的页面、我们的程序能够支持所有的浏览器、能够满足尽可能多的用户。我们要满足所有的客户, 即使做不到, 我们也要满足我们技术范围之内所有用户。

二、目标

一个标准制作的网站, 让你压根感觉不到跟标准有关。

三、受众

所有ui设计师、技术工程师、运营维护人员。

四、标准内容

抛弃声明: 以后我们将抛弃font标签, 新的页面中不应该再出现如, 已经存在的老的页面也应该在修改时尽量替代, 替代方法: 。

一个标准XHTML头信息格式如下:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "
```

```
<html xmlns="http://www.w3.org/1999/xhtml" lang="gb2312">
<head>
  <meta charset="utf-8" />
  <title>W3Cschoo1 - 学技术查资料, 从w3cschool开始!</title>
  <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=:" />
  <meta name="renderer" content="webkit" />
  <meta name="viewport" content="width=device-width, initial-:" />
  <meta name="keywords" content="w3cschool,w3cschool在线教程,技
  <meta name="description" content="w3cschool是一个专业的编程入门
</head>
```

1、什么是DOCTYPE

DOCTYPE是document type(文档类型)的简写, 用来说明你用的XHTML或者HTML是什么版本。其中的DTD(例如xhtml1-transitional.dtd)叫文档类型定义, 里面包含了文档的规则, 浏览器就根据你定义的DTD来解释你页面的标识, 并展现出来。要建立符合标准的网页, DOCTYPE声明是必不可少的关键组成部分; 除非你的XHTML确定了一个正确的DOCTYPE, 否则你的标识和CSS都不会生效。

XHTML 1.0 提供了三种DTD声明可供选择:

i) 过渡的(Transitional): 要求非常宽松的DTD, 它允许你继续使用HTML4.01的标识(但是要符合xhtml的写法)。

完整代码如下:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "
```

ii) 严格的(Strict): 要求严格的DTD, 你不能使用任何表现层的标识和属性, 例如
。

完整代码如下:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http:,"
```

iii) 框架的(Frameset): 专门针对框架页面设计使用的DTD, 如果你的页面中包含有框架, 需要采用这种DTD。

完整代码如下:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http:
```

注: DOCTYPE声明必须放在每一个XHTML文档最顶部, 在所有代码和标识之上。

2、名字空间 namespace

```
<html xmlns="http://www.w3.org/1999/xhtml" lang="gb2312">
```

通常我们HTML4.0的代码只是<html>, 这里的"xmlns"是什么呢?

这个“xmlns”是XHTML namespace的缩写, 叫做“名字空间”声明。XHTML是HTML向XML过渡的标识语言, 它需要符合XML文档规则, 因此也需要定义名字空间。又因为XHTML1.0不能自定义标识, 所以它的名字空间都相同, 就是“http://www.w3.org/1999/xhtml”。目前阶段我们只要照抄代码就可以了。

3、定义语言编码

<meta http-equiv="Content-Type" content="text/html; charset=gb2312" />

为了被浏览器正确解释和通过W3C代码校验，所有的XHTML文档都必须声明它们所使用的编码语言，我们一般使用gb2312(简体中文)，制作多国语言页面也有可能用Unicode、ISO-8859-1等，根据你的需要定义。

注：如果忘记了定义语言编码，可能就会出现，你在DW(dreamweaver)做完一个页面，第二次打开时所有的中文变成了乱码。

4、Javascript定义

Js必须要用<script language="javascript" type="text/javascript">来开头定义，而不是原来的<script language=javascript>或干脆直接<script>，并且需要加个注释符<!-- -->，以保证不在不支持js的浏览器上直接显示出代码来。

例如：

```
<script language="javascript" type="text/javascript">
//
function show_layout(selObj){
var n = selObj.options[selObj.selectedIndex].value;
document.getElementById('stylesheet').href = n;
}
//]]&gt;
&lt;/script&gt;</pre></div><div data-bbox="57 476 276 496" data-label="Text"><p>注：具体参考js规范。</p></div><div data-bbox="57 510 187 528" data-label="Section-Header"><h4>5、CSS定义</h4></div><div data-bbox="57 526 949 577" data-label="Text"><p>CSS必须要用&lt;style type="text/css"&gt;开头来定义，而不是原来的直接&lt;style&gt;，也不建议直接写在内容代码里如：&lt;div style="padding-left:20px;"&gt;&lt;/div&gt;，并需要加个注释符&lt;!-- --&gt;</p></div><div data-bbox="57 576 119 595" data-label="Text"><p>例如：</p></div><div data-bbox="57 612 943 724" data-label="Text"><pre>&lt;style type="text/css" media="screen"&gt;
&lt;!--
body {margin:0px;padding:0px;font-size:12px;text-align:center}
--&gt;
&lt;/style&gt;</pre></div><div data-bbox="57 737 763 775" data-label="Text"><p>为保证各浏览器的兼容性，在写CSS时请都写上数量单位，例如：错误：.space_10{padding-left:10} 正确：.space_10 {padding-left:10px}</p></div><div data-bbox="57 788 372 807" data-label="Section-Header"><h4>6、不要在注释内容中使用“--”</h4></div><div data-bbox="57 805 879 824" data-label="Text"><p>--只能发生在XHTML注释的开头和结束，也就是说，在内容中它们不再有效。</p></div><div data-bbox="57 821 732 841" data-label="Text"><p>例如下面的代码是无效的：&lt;!--这里是注释-----这里是注释--&gt;</p></div><div data-bbox="57 838 925 873" data-label="Text"><p>正确的应用等号或者空格替换内部的虚线。&lt;!--这里是注释=====这里是注释--&gt;</p></div><div data-bbox="57 887 568 907" data-label="Section-Header"><h4>7、所有标签的元素和属性的名字都必须使用小写</h4></div><div data-bbox="57 904 876 923" data-label="Text"><p>与HTML不一样，XHTML对大小写是敏感的，&lt;title&gt;和&lt;TITLE&gt;是不同的标签。</p></div><div data-bbox="57 921 905 957" data-label="Text"><p>XHTML要求所有的标签和属性的名字都必须使用小写。例如：&lt;BODY&gt;必须写成&lt;body&gt;。大小写夹杂也是不被认可的，通常dreamweaver自动生成的属性名</p></div>
```

字"onMouseOver"也必须修改成"onmouseover"。

8、所有的属性必须用引号""括起来

在HTML中，你可以不需要给属性值加引号，但是在XHTML中，它们必须被加引号。

例如：<height=80>必须修改为：<height="80">。

特殊情况，你需要在属性值里使用双引号，你可以用"，单引号可以使用'，例如：<alt="say'hello'">

9、把所有<和&特殊符号用编码表示

任何小于号 (<)，不是标签的一部分，都必须被编码为 <

任何大于号 (>)，不是标签的一部分，都必须被编码为 >

任何与号 (&)，不是实体的一部分的，都必须被编码为 &

错误：

http://club.china.alibaba.com/forum/thread/search_forum.html?

action=SearchForum&doSearchForum=true&main=1&catcount=10&keywords=mp3

正确：

http://club.china.alibaba.com/forum/thread/search_forum.html?

action=SearchForum&doSearchForum=true&main=1&catcount=10&keyw

10、给所有属性赋一个值

XHTML规定所有属性都必须有一个值，没有值的就重复本身。例如：

<td nowrap><input type="checkbox" name="shirt" value="medium" checked>必须修改为：

<td nowrap="nowrap"><input type="checkbox" name="shirt" value="medium" checked="checked" />

11、所有的标记都必须要有个相应的结束标记

以前在HTML中，你可以打开许多标签，例如<p>和而不一定写对应的</p>和来关闭它们。但在XHTML中这是不合法的。XHTML要求有严谨的结构，所有标签必须关闭。如果是单独不成对的标签，在标签最后加一个"/"来关闭它。

例如：

特殊结束标记

错误：

Document.write("<td width=\"300\">ok</td>");

正确：

Document.write("<td width=\"300\">ok<\&a><\&td>");

在js中，原已结束的标签需要再转义再结束。

12、所有的标记都必须合理嵌套

同样因为XHTML要求有严谨的结构，因此所有的嵌套都必须按顺序，以前我们这样写的代码：

<p></p>必须修改为：<p></p>

就是说，一层一层的嵌套必须是严格对称。

错误：

<table><tr><form><td></td></form></tr></table>

正确：

<form><table><tr><td></td></tr></table></form>

13、图片添加有意义的alt属性

例如：

尽可能的让作为内容的图片都带有属于自己的alt属性。

同理：添加文字链接的title属性。

新闻新闻...，在一些限定字数的内容展示尤为重要，帮助显示未完成的内容显示完整，而不用考虑页面会因此而撑大。

14、在form表单中增加lable，以增加用户友好度
例如：

```
<form action="http://somesite.com/prog/adduser" method="post">
  <label for="firstname">first name: </label>
  <input type="text" id="firstname" />
  <label for="lastname">last name: </label>
  <input type="text" id="lastname" />
</form>
```

附录一

1、XHTML校验

校验网址：<http://validator.w3.org/>

校验方式：网址校验、文件上传校验

一般选择"Show Source"和"Verbose Output"可以帮助你找到错误代码所在行和错误原因。

XHTML校验常见错误原因对照表

- 1.No DOCTYPE Found! Falling Back to HTML 4.01 Transitional--未定义DOCTYPE。
 - 2.No Character Encoding Found! Falling back to UTF-8--未定义语言编码。
 - 3.end tag for "img" omitted, but OMITTAG NO was specified--图片标签没有加"/"关闭。
 - 4.an attribute value specification must be an attribute value literal unless SHORTTAG YES is specified--属性值必须加引号。
 - 5.element "DIV" undefined---DIV标签不能用大写，要改成小写div。
 - 6.required attribute "alt" not specified---图片需要加alt属性。
 - 7.required attribute "type" not specified---JS或者CSS调用的标签漏了type属性。
- 其中最最常见的错误就是标签的大小写问题了。通常这些错误都是关联的，比如忘记了一个其他标签都会报错，所以不要看到一堆的错误害怕，通常解决了一个错误，其他的错误也都没有了。

附录二

2、CSS2校验

校验网址：<http://jigsaw.w3.org/css-validator/>

校验方式：网址校验、文件上传校验、直接贴入代码校验

校验成功，会显示"恭喜恭喜，此文档已经通过样式表校验！"。校验失败，会显示两类错误：错误和警告。错误表示一定要修正，否则无法通过校验；警告表示有代码不被W3C推荐,建议修改。

同样，通过检验后，可以放置一个CSS校验通过图标。

CSS2校验常见错误原因对照表

- 1.(错误)无效数字：color909090不是一个color值，909090---十六进制颜色值必须加"#"号，即#909090
- 2.(错误)无效数字：margin-topUnknown dimension: 6pixels ---pixels不是一个单位值，正确写法6px
- 3.(错误)属性：scrollbar-face-color 不存在：#eeeeee --- 定义滚动条颜色是非标准的属性
- 4.(错误)值：cursorhand不存在：hand是非标准属性值，修改为cursor:pointer

5.(警告)Line:0 font-family: 建议你指定一个种类族科作为最后的选择 --W3C建议字体定义的时候，最后以一个类别的字体结束，例如"sans-serif"，以保证在不同操作系统下，网页字体都能被显示。

6.(警告)Line:0 can't find the warning message for otherprofile --表示在代码中有非标准属性或值，校验程序无法判断和提供相应的警告信息。

w3c标准验证链接：

- [验证 HTML](#)
- [验证 CSS](#)
- [验证 XHTML](#)
- [验证 XML](#)
- [验证 WML](#)

W3Cscool (www.w3cschool.cn) 最大的技术知识分享与学习平台

此篇内容来自于[w3cschool.cn](#)网站用户上传并发布。