**1. 一定要闭合HTML标签**       
     在以往的页面源代码里，经常看到这样的语句：

1. <li>Some text here.
2. <li>Some new text here.
3. <li>You get the idea.

复制代码

也许过去我们可以容忍这样的非闭合HTML标签，但在今天的标准来看，这是非常不可取的，是必须百分百避免的。一定要注意闭合你的HTML标签，否则将无法通过验证，并且容易出现一些难以预见的问题。  
  
      最好使用这样的形式：

1. <ul>
2. <li>Some text here. </li>
3. <li>Some new text here. </li>
4. <li>You get the idea. </li>
5. </ul>

复制代码

**2. 声明正确的文档类型( DocType )**  
      笔者早先曾加入过许多CSS论坛，在那里，如果有用户遇到问题，我们会建议他首先做两件事：

* 1. 验证CSS文件，解决所有可见的错误
* 2. 加上文档类型 Doctype

      DOCTYPE 定义在HTML标签出现之前，它告诉浏览器这个页面包含的是HTML，XHTML，还是两者混合出现，这样浏览器才能正确的解析标记。

**通常有四种文档类型可供选择**：

1. 1. <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
2. "[url=http://www.w3.org/TR/html4/strict.dtd]http://www.w3.org/TR/html4/strict.dtd[/url]">
3. 2. <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "[url=http://www.w3.org/TR/html4/loose.dtd]http://www.w3.org/TR/html4/loose.dtd[/url]">
4. 3. <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "[url=http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd]http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd[/url]">
5. 4. <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
6. "[url=http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd]http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd[/url]">

复制代码

关于该使用什么样的文档类型声明，一直有不同的说法。通常认为使用最严格的声明是最佳选择，但研究表明，大部分浏览器会使用普通的方式解析这种声明，所以很多人选择使用HTML4.01标准。选择声明的底线是，它是不是真的适合你，所以你要综合考虑来选择适合你得项目的声明。  
  
**3. 不要使用嵌入式CSS样式**  
      当你在埋头写代码时，可能会经常顺手或偷懒的加上一点嵌入式css代码，就象这样：

1. <p style="color: red;">W3CFuns.com</p>

复制代码

这样看起来即方便又没有问题，但是它会在你得代码中产生问题。  
       
     在你开始写代码时，最好是在内容结构完成之后再开始加入样式代码。

这样的编码方式就像打游击，是一种很山寨的做法。——Chris Coyier

        更好的做法是，把这个P的样式定义在样式表文件里：

1. [code]someElement > p {
2. color: red;
3. }

复制代码

**4. 在页面head标签中引入所有的样式表文件**理论上讲，你可以在任何位置引入CSS样式表，但HTML规范建议在网页的head标记中引入，这样可以加快页面的渲染速度。

     在雅虎的开发过程中，我们发现，在head标签中引入样式表，会加快网页加载速度，

因为这样可以使页面逐步渲染。 —— ySlow团队

1. <head>
2. <title>My Favorites Kinds of Corn</title>
3. <link rel="stylesheet" type="text/css" media="screen" href="path/to/file.css" />
4. <link rel="stylesheet" type="text/css" media="screen" href="path/to/anotherFile.css" />
5. </head>

复制代码

**5. 在页面底部引入javascript文件**  
      要记住一个原则，就是让页面以最快的速度呈现在用户面前。当加载一个脚本时，页面会暂停加载，直到脚本完全载入。所以会浪费用户更多的时间。  
  
      如果你的JS文件只是要实现某些功能，（比如点击按钮事件），那就放心的在body底部引入它，这绝对是最佳的方法。  
 **举例**：

1. <p>And now you know my favorite kinds of corn. </p>
2. <script type="text/javascript" src="path/to/file.js"></script>
3. <script type="text/javascript" src="path/to/anotherFile.js"></script>
4. </body>
5. </html>

复制代码

**6. 不要使用嵌入式JavaScript，这都21世纪了！**许多年以前，还存在一种这样的方式，就是直接将JS代码加入到HTML标签中。尤其是在简单的图片相册中非常常见。本质上讲，一个“onclick”事件是附加在标签上的，其效果等同于一些JS代码。不需要讨论太多，非常不应该使用这样的方式，应该把代码转移到一个外部JS文件中，然后使用“ addEventListener / attachEvent ”加入时间侦听器。或者使用jquery等框架，之需要使用其“clock”方法。

1. $('a#moreCornInfoLink').click(function() {
2. alert('Want to learn more about corn?');
3. });

复制代码

**7. 开发中随时进行标准验证**很多人并不真正理解标准验证的意义和价值，笔者在一篇博客中详细分析了这个问题。一句话，标准验证是为你服务的，不是给你找麻烦的。  
  
     如果你刚开始从事网页制作，那强烈建议你下载这个[网页开发工具条](https://addons.mozilla.org/en-US/firefox/addon/60" \t "_blank) ，并在编码过程中随时使用”HTML标准验证”和“CSS标准验证”。如果你认为CSS是一种非常好学的语言，那么它会把你整的死去活来。你的不严谨的代码会让你的页面漏洞百出，问题不断，一个好的方法就是—— 验证，验证，再验证。  
 **8. 下载Firebug**  
      Firebug是当之无愧的网页开发最佳插件，它不但可以调试JavaScript，还可以直观的让你了解页面标记的属性和位置。不用多说， [下载它！](https://addons.mozilla.org/cn-ZH/firefox/addon/1843)  
 **9. 使用Firebug！**  
     据笔者观察，大部分的使用者仅仅使用了Firebug 20%的功能，那真是太浪费了，你不妨花几个小时的时间来系统学习这个工具，相信会让你事半功倍。  
 **Firebug教程**：

* [Overview of Firebug](http://michaelsync.net/2007/09/08/firebug-tutorial-overview-of-firebug)
* [Debug Javascript With Firebug – video tutorial](http://www.digitalmediaminute.com/screencast/firebug-js/)  
    
  **10. 使用小写的标记**  
       理论上讲，你可以像这样随性的书写标记：

1. <DIV>
2. <P>Here's an interesting fact about corn. </P>
3. </DIV>

复制代码

最好不要这样写，费力气输入大些字母没有任何用处，并且会让代码很难看，这样子就很好：

1. <div>
2. <p>Here's an interesting fact about corn. </p>
3. </div>

复制代码

**11.使用H1 – H6标签**  
      笔者建议你在网页中使用其中全部六种标记，虽然大部分人只会用到前四个，但使用最多的H会有很多好处，比如设备友好、搜索引擎友好等，不妨把你的P标签都替换成H6。  
  
**12. 如果是博客，那把H1留给文章标题**  
     今天笔者在Twitter上发起一次讨论：是该把H1定义到LOGO上还是定义到文章标题上，有80%的人选择了后者。  
  
     当然具体如何使用要看你的需求，但我建议你在建立博客的时候，将文章题目定为H1，这对搜索引擎优化(seo)是非常有好处的。  
 **13. 下载ySlow**在过去几年里，雅虎的团队在前端开发领域做了许多伟大的工作。前不久，它们发布了一个叫ySlow的Firebug扩展，它会分析你的<网页，并返回 一个“成绩单”，上面细致分析了这个网页的方方面面，提出需要改进的地方，虽然它有点苛刻，但它绝对会对你有所帮助，强烈推荐—— [ySlow！](http://developer.yahoo.com/yslow/" \t "_blank)  
  
**14. 使用UL列表布局导航菜单**通常网站都会有导航菜单，你可以用这样的方式定义：

1. <div id="nav">
2. <a href="#">Home </a>
3. <a href="#">About </a>
4. <a href="#">Contact </a>
5. </div>

复制代码

如果你想书写优美的代码，那最好不要用这种方式，

为什么要用UL布局导航菜单？  
——因为UL生来就是为定义列表准备的

    最好这样定义：

1. <ul id="nav"><li><a href="#">Home</a></li><li><a href="#">About</a></li><li><a href="#">Contact</a></li></ul>

复制代码

**15. 学会怎样对付IE**  
    IE一直以来都是前端开发人员的噩梦！  
  
    如果你的CSS样式表基本定型了，那么可以为IE单独建立一个样式表，然后这样仅对IE生效：

1. <!--[if lt IE 7]>
2. <link rel="stylesheet" type="text/css" media="screen" href="path/to/ie.css" />
3. <![endif]-->

复制代码

这些代码的意思是：如果用户浏览器是IE6及以下，那这段代码才会生效。如果你想把IE7也包含进来，那么就把“[if lt IE 7]”改为“[if lt IE 7]”。