



# 前端开发规范文档

**2.0**

文档修改历史记录

日 期	版本号	更改人	更改内容
04-22-2012	2.0	黄小东	创建文档

# 目录

基本准则与说明.....	4
文件命名.....	4
第一章    标准与优化.....	5
W3C标准.....	5
HTML优化.....	5
HTML技巧与常识.....	5
CSS优化.....	6
CSS技巧与常识.....	6
JS技巧与常识.....	7
第二章    约定与规范.....	8

## 基本准则与说明

符合 w3c 标准, 语义化 html 标签, 结构表现行为分离, 兼容性优良。代码简洁有序, 尽可能减小服务器负载以及服务器请求数, 保证最快的解析速度和最优的页面性能。做最优于用户的 web 开发

说明: 本文档适用于浏览器 IE7+、firefox、Chrome、opera、safari

## 文件命名

说明: 一般使用驼峰法则 (如 :docDownload.html) 或者下划线命名方式 (如: doc\_download.html)、根据本站目前情况使用后者

**html 文件命名:** 英文命名\_后缀.htm (如: product\_down.html), 不使用拼音开头字母、无意义英文数字或者只有作者自己懂的命名等

**css 文件命名:** 英文命名\_后缀.css (如: global\_zh.css)

**Js 文件命名:** 英文命名\_后缀.js。

**Image 文件命名:** 英文命名\_后缀.扩展名

# 第一章 标准与优化

## W3C标准

1. 所有的 HTML 标签使用小写英文字母，且必须闭合。如：<div></div>、<br />
2. 所有的 HTML 标签的属性值必须用双引号包括。如：<a href="#"></a>
3. 所有的 HTML 文件必须声明文档类型以及编码
4. 所有的 META 标签都应该放在 TITLE 标签的前面，且顺序依次是：  
content-type,content-language,robots,key-words,description

更多规范参照：[www.w3school.com.cn](http://www.w3school.com.cn)

## HTML优化

1. 语义化的 HTML 标签：如：段落使用 p、列表使用 ul li、表格使用 table、标题使用 h1、h2、h3 等
2. 尽量少的无意义标签嵌套，如需通过标签实现效果，自己把握实现代价
3. 页面上尽量少的使用 style 定义样式（并非不可使用，例如特别的页面）
4. 无意义的装饰用图片，尽量作为背景图写到 CSS 中，而不是使用 img 标签
5. 页面上的 img 需要加入 alt 属性，以便在加载失败的时候，用户知道这个图片是什么
6. 背景图尽量使用 sprite 技术（就是把许多常用小图，整合成一个大图，通过背景定位来显示），减小 http 请求。

## HTML技巧与常识

1. 图片使用 gif、jpg、以及 png 格式，其中 PNG 格式支持通道透明（半透明），GIF 格式图片支持全透明，IE6 不支持 PNG 通道透明，透明图片优先使用 GIF 格式，PNG 图片尽量使用 PNG-8 标准格式
2. HTML 模块化开发，良好的代码缩进以及分类习惯，先想好怎么实现，再动手

3. 所有需要居中的图片还有层的宽度必须为偶数像素(如: 980px), 高度尽量也使用偶数。
4. 块状结构以及非块状结构的标签, 应该更准确的被使用(还是需要对标签理解深刻啊, 经验啊)
5. 所有的图标都使用 16x16、32x32、48x48 大小, 不要图方便直接切出来, 将来不通用
6. 尽量从程序的角度考虑 html 代码实现, 比如: 新闻列表, 那么文字就要做超出隐藏, 如果是一段话的弹窗, 那么就要做宽高自适应等(这个需要特别注意)

## CSS优化

1. 所有的 CSS 统一常用样式提成一个公共布局文件, 视情况尽量公用这个布局 css,
2. 其他 css 按整站大功能模块分类, 同一模块下使用尽量少的 CSS 文件(一般只用一个), 以减少服务器请求数。
3. CSS 尽量使用继承, 少使用独立样式, 来减少代码量(但是在自己 CSS 能力不强, 权重理解不是很深的情况下, 少用继承)
4. 尽量使用 CSS 的权重替换来做一些相似效果的切换, 尽量减少程序以及 JS 工作量以及 html 代码量, 不要把所有效果寄希望于 JS 和程序, 最优的搭配才是最好的效率(还是经验问题)
5. 在权重可控的情况下尽量不使用 ID(把 ID 留给程序使用), 命名尽量使用短英文
6. 在对 CSS 理解还好的情况下, 尽量不要使用 hack, 基本上都不需要 hack 即可达到兼容效果

## CSS技巧与常识

1. 关于权重顺序: 写在标签上的 style 权重最高, 依次是 id, class, 标签
2. 关于权重叠加(只有 IE6 有部分问题), 如: #header .logo{}的权重低于#header div .logo{}
3. 相同命名的 CSS 权重, 写在后面的 CSS, 高于写在前面的
4. 良好的文本超出隐藏习惯, 不要所有控制字数的工作都交给程序
5. 尽量给外部提供现成的 CSS 接口, 比如 JS 只需控制他的 CLASS 变可以实现效果转变, JS 中应该尽量少的出现直接控制 css 样式的情况

## JS技巧与常识

1. 选择器尽量直接选中 DOM，减少 html 树的遍历数，如：`$("#header")` 明显比 `$("div#header")` 效率高
2. for 循环内部使用的选择器，尽量在 for 循环之前，定义成一个变量
3. 尽量多使用 `this`
4. 少使用全局变量，即使要使用也要定义一个不易重复的前缀
5. 尽量少使用不同的 JS 类库，以免造成冲突（推荐 JQuery）
6. 常用方法最好提成一个通用方法

## 第二章 约定与规范

说明：以下仅仅为约定大部分情况，特殊情况按照实际情况处理

1. 默认使用“宋体”，正文字体大小 12 像素，标题 14 像素
2. 所有的图标使用 16x16、32x32 等规格(严禁 12x16 什么的)
3. 所有 select 下拉框以及 input 文件上传框是不控制高度以及样式，因为各个浏览器表现不一，为提高页面效率尽量使用相同同的输入框样式以及弹窗样式。
4. A 标签文本链接默认颜色为# 0085c5，无下划线，鼠标停留颜色不变，出下划线
5. 页面宽度为 1004 像素
6. 所有文件统一编码为 UFT-8
7. 所有间距使用 5 像素的整数倍如 5px、10px、15px
8. CSS 注释约定 （等级越高注释线越长）

一级模块分割注释:/\*=====注释=====\*/

二级模块分割注释:/\*-----注释-----\*/

三级模块分割注释:/\* \*\*\*\*\*注释\*\*\*\*\* \*/

默认注释: /\*\*/

HTML 的注释同理

9. Css 文件需要在页头注释最后修改的时间、修改人、联系方式、以及版权和作者信息等
10. 开发过程中，大区块 DIV 结束标签必须注释名字，如 <div class="header"></div><!--header-->
11. 网页使用的图片，严格按照偶数宽高来定。宽高比例可以为：1:1，9:6，6:9 等，特殊情况除外
12. 所有标题的高度都必须是偶数宽高
13. 图片样式设计，能使用 CSS 的而且对效果影响不大的，少使用阴影。（因为有时候实现代价大，又没有特别的意义）