

# 多迪标准：前端切图排

2013年10月24日

- (一) 初审标准
- (二) 复审标准
- (三) 附件一：网站制作切图规范
- (四) 附件二：常用的模块命名和css文件命名

## （一） 初审：

存在以下问题任何一项均属不及格。

**A1、**作品与提供的设计效果图某些地方不一致

比如字体大小、颜色、行距、图标及排版等均应严格一致

文字不应写错（除非效果图本身是错字），比如效果图上的“无限极”切图后误写成“无极限”

**A2、**在IE6, IE7, IE8及FIREFOX存在兼容问题，比如变形，JavaScript错误或者失效

**A3、**JavaScript编程是前端工程师的工作职责，除了部分需要结合PHP或者MYSQL才能完成的，其它地方如有未完成则作品不及格，比如下拉菜单，图片切图效果，选项卡等

**A4、**页面全部采用DIV+CSS布局代替TABLE布局

**A5、**CSS和JavaScript代码各自成为独立的外部文件，而不是把代码写在HTML页面

**A6、**列表要考虑程序能够写循环，即写好一行DIV，把它复制N行，否则视为不及格

**A7、**切图时，请主观判断，文字是否有链接，需要链接的文字请加上超级链接，避免开发阶段加上链接后样式错乱，如明显需要链接但无做好链接的

## （二） 复审：

通过初审的作品，才会进入复审。复审是优化内容，参考附件一进得优化

**B1**、Class等命名参考附件二

**B2**、样式属性有适当代码缩写

**B3**、样式代码有适当的注释

**B4**、文件夹及文件名称命名规范

**B5**、首行缩进 `text-indent`

**B6**、背景图优化 `background-position`

**B7**、绝对定位 `position:absolute`

**B8**、网站页面存在样式相同的栏目，均使用相同的DIV和CLASS名称

**B9**、能用HTML完成的文字，尽量都不直接写在图片，比如若干相同样式的搜索按钮，只用一张背景图，文字部分由HTML完成，页面90%以上的地方能做到这样

**C1**、Div居中：有width值，并且左右margin:auto

**C2**、浮动及清理 Float 给高度或者clear:both

**C3**、能够不用图片的地方就不要用，比如线条、色块等应直接用代码来完成

## （三）附件一：网站制作切图规范

为提高团队协作效率，便于后台人员添加功能及前端后期优化维护，输出高质量的文档，特制订此文档。本规范文档一经确认，前端开发人员必须按本文档规范进行前台页面开发。本文档如有不对或者不合适的地方请及时提出，经讨论决定后方可更改。

### 基本准则

符合web标准，语义化html，结构表现行为分离，兼容性优良。页面性能方面，代码要求简洁明了有序，尽可能的减小服务器负载，保证最快的解析速度。

### 文件规范

1. html, css, js, images文件均归档至<系统开发规范>约定的目录中；
2. html文件命名：英文命名，后缀.htm。同时将对对应界面稿放于同目录中，若界面稿命名为中文，请重命名与html文件同名，以方便后端添加功能时查找对应页面；
3. css文件命名：英文命名，后缀.css。共用base.css，首页index.css，其他页面依实际模块需求命名。；
4. Js文件命名：英文命名，后缀.js。共用common.js，其他依实际模块需求命名。

### html书写规范

1. 文档类型声明及编码：统一为html5声明类型；编码统一为utf-8，书写时利用IDE实现层次分明的缩进；
2. 非特殊情况下样式文件必须外链至<header></header>之间；非特殊情况下JavaScript文件必须外链至页面底部；
3. 引入样式文件或JavaScript文件时，须略去默认类型声明，写法如下：

```
<link rel="stylesheet" href="..." />
<style>...</style>
<script src="..."></script>
```

4. 引入JS库文件，文件名须包含库名称及版本号及是否为压缩版，比如jquery-1.4.1.min.js；引入插件，文件名格式为库名称+插件名称，比如jQuery.cookie.js；
5. 所有编码均遵循xhtml标准，标签 & 属性 & 属性命名 必须由小写字母及下划线数字组成，且所有标签必须闭合，包括 br (<br />)，hr(<hr />)等；属性值必须用双引号包括；
6. 充分利用无兼容性问题的html自身标签，比如span, em, strong, optgroup, label, 等等；需要为html元素添加自定义属性时，首先要考虑下有没有默认的已有的合适标签去设置，如果没有，可以使用须以"data-"为前缀来添加自定义属性，避免使用"data:"等其他命名方式；

7. 语义化html，如 标题根据重要性用h\*(同一页面只能有一个h1)，段落标记用p，列表用ul，内联元素中不可嵌套块级元素；

8. 尽可能减少div嵌套，如<div class="box"><div class="welcome">欢迎访问XXX，您的用户名是<div class="name">用户名</div></div></div>完全可以用以下代码替代：<div class="box"><p>欢迎 访问XXX，您的用户名是<span>用户名</span></p></div>；

9. 书写链接地址时，必须避免重定向，例如：href="http://itaolun.com/"，即须在URL地址后面加上 "/" ；

10. 在页面中尽量避免使用style属性，即style="..."；

11. 必须为含有描述性表单元素(input, textarea)添加label，如

```
<p>姓 名: <input type="text" id="name" name="name" /></p>
```

须写成：

```
<p><label for="name">姓 名: </label><input type="text" id="name" /></p>
```

12. 能以背景形式呈现的图片，尽量写入css样式中；

13. 重要图片必须加上alt属性；给重要的元素和截断的元素加上title；

14. 给区块代码及重要功能(比如循环)加上注释，方便后台添加功能；

15. 特殊符号使用：尽可能使用代码替代：比如 <(<) & >(>) & 空格( ) & »(») 等等；

16. 书写页面过程中，请考虑向后扩展性；

17. class & id 参见 css书写规范.

## css书写规范

1. 编码统一为utf-8；

2. 协作开发及分工：根据各个模块，同时根据页面相似程序，事先写好大体框架文件，分配给前端人员实现内部结构&表现&行为；共用css文件为base.css，协作开发过程中，每个页面请务必都要引入，此文件包含reset及头部底部样式，此文件不可随意修改；

3. class与id的使用：

样式都用class而不用id，id专门留给JS用，这样才符合表现与行为分离的原则。命名方式使用下划线规则，如main\_nav。

4. 为JavaScript预留钩子的命名，请以 js\_ 起始，比如：js\_hide, js\_show；

5. class与id命名：

大的框架命名比如header/footer/wrapper/left/right之类的统一命名。其他样式名称由小写英文 & 数字 & \_ 来组合命名，如i\_comment, fontred, width200；避免使用中文拼音，尽量使用简易的单词组合；总之，命名要语义化，简明化。

6. 要在2中已建好框架的页面代码<div class="main\_nav"></div>中加入新的div元素，

按a命名法则：<div class=" main\_nav "><div class="firstnav">...</div></div>，

样式写法： #mainnav .firstnav{.....}

按b命名法则：<div class =" main\_nav "><div class="main\_firstnav">...</div></div>，

样式写法： .main\_firstnav{.....}

7. css属性书写顺序，建议遵循：布局定位属性-->自身属性-->文本属性-->其他属性。  
此条可根据自身习惯书写，但尽量保证同类属性写在一起。

属性列举：

布局定位属性主要包括：display & list-

style & position（相应的 top,right,bottom,left） & float & clear & visibility & overflow;

自身属性主包：width & height & margin & padding & border & background;

文本属性主要包括：color & font & text-decoration & text-align & vertical-align & white-space & 其他 & content;

这里列出的这些属性只是最常用到的，并不代表全部；

8. 书写代码前，考虑并提高样式重复使用率；

9. 充分利用html自身属性及样式继承原理减少代码量，比如：

```
<ul class="list"><li>这儿是标题列表<span>2010-09- 15</span></li></ul>
```

定义

```
ul.list li{position:relative} ul.list li span{position:absolute; right:0}
```

即可实现日期居右显示

10. 样式表中中文字体名，请务必转码成unicode码，以避免编码错误时乱码；

11. 背景图片和ico请尽可能使用sprite技术，减小http请求，考虑到多人协作开发，sprite按模块制作；

12. 使用table标签时(尽量避免使用table标签)，请不要用width/height/cellspacing/cellpadding等table属性直接定义表现，应尽可能的利用table自身私有属性分离结构与表现，

如thead, tr, th, td, tbody, tfoot, colgroup, scope;

（cellspaing及cellpadding的css控制方法：

```
table{border:0;margin:0;border-
```

```
collapse:collapse;} table th, table td{padding:0;} ,
```

base.css文件中由i初始化表格样式)

13. 杜绝使用<meta http-equiv="X-UA-Compatible" content="IE=7" /> 兼容 ie8;

14. 用png图片做图片时，要求图片格式为png-8格式，若png-8实在影响图片质量或其中有半透明效果，请为ie6单独定义背景
15. 避免兼容性属性的使用，比如text-shadow || css3的相关属性；
16. 减少使用影响性能的属性，比如position:absolute || float ；
17. 必须为大区块样式添加注释，小区块适量注释；
18. 代码缩进与格式：建议单行书写，可根据自身习惯，后期优化由i统一处理；

## JavaScript书写规范

1. 文件编码统一为utf-8，书写过程过，每行代码结束必须有分号；  
原则上所有功能均根据XXX项目需求原生开发，以避免网上down下来的代码造成的代码污染(沉冗代码 || 与现有代码冲突 || ...)；
2. 库引入：原则上仅引入jQuery库，若需引入第三方库，须与团队其他人员讨论决定；
3. 变量命名：驼峰式命名。原生JavaScript变量要求是纯英文字母，首字母须小写，如iTaoLun；  
jQuery变量要求首字符为'\_'，其他与原生JavaScript 规则相同，如：\_iTaoLun；  
另，要求变量集中声明，避免全局变量。
4. 类命名：首字母大写，驼峰式命名。如 ITaoLun；
5. 函数命名：首字母小写驼峰式命名。如iTaoLun()；
6. 命名语义化，尽可能利用英文单词或其缩写；
7. 尽量避免使用存在兼容性及消耗资源的方法或属性，比如eval() & innerText；
8. 后期优化中，JavaScript非注释类中文字符须转换成unicode编码使用，以避免编码错误时乱码显示；
9. 代码结构明了，加适量注释。提高函数重用率；
10. 注重与html分离，减小reflow，注重性能。

## 图片规范

1. 所有页面元素类图片均放入img文件夹，测试用图片放于images文件夹；
2. 图片格式仅限于gif || png || jpg；
3. 命名全部用小写英文字母 || 数字 || \_ 的组合，其中不得包含汉字 || 空格 || 特殊字符；尽量用易懂的词汇，便于团队其他成员理解；另，命名分头尾两部分，用下划线隔开，比如ad\_left01.gif || btn\_submit.gif；

4. 在保证视觉效果的情况下选择最小的图片格式与图片质量，以减少加载时间；
5. 尽量避免使用半透明的png图片(若使用，请参考css规范相关说明)；
6. 运用css sprite技术集中小的背景图或图标，减小页面http请求，但注意，请务必在对应的sprite psd源图中划参考线，并保存至img目录下。

## 注释规范

1. html注释：注释格式 `<!--这儿是注释-->`，'--'只能在注释的始末位置，不可置入注释文字区域；
2. css注释：注释格式 `/*这儿是注释*/`；
3. JavaScript注释，单行注释使用 `//这儿是单行注释`，多行注释使用 `/* 这儿有多行注释 */`；

## 开发及测试

1. 不可利用IDE的视图模式'画'代码；
2. 不可利用IDE生成相关功能代码，比如Dw内置的一些功能js；
3. 编码必须格式化，比如缩进；

建议测试顺序：Firefox-->IE7-->IE8-->IE6-->Opera-->Chrome-->Safari，  
建议安装firebug及IE Tab Plus插件。

## 其他规范

1. 开发过程中严格按分工完成页面，以提高css复用率，避免重复开发；
2. 减小沉冗代码，书写所有人都可以看的懂的代码。简洁易懂是一种美德。为用户着想，为服务器着想。

## （四）附件二：常用的模块命名和css文件命名

### （一）页面结构

页面外围控制整体布局宽度：wrapper

容器：container



侧栏：sidebar

栏目：column

左右中：left right center

内容：content/container

页眉：header

页脚：footer

## （二）导航

主导航：main\_nav

子导航：sub\_nav

顶导航：top\_nav

边导航：sidebar

左边导航：left\_side\_bar

右边导航：right\_side\_bar

菜单容量：menu\_contaier

子菜单：sub\_menu

边导航图标：side\_bar\_icon

注释：note

面包屑：bread\_crumb

摘要：summary

标题：title

## （三）功能

商标：label

广告： banner

搜索： search

登录条： login\_bar

登录： login

注册： regsiter

提示信息： msg

小技巧： tips

功能区： shop（如购物车，收银台）

当前的： current

加入： joinus

状态： status

按钮： btn

热点： hot

新闻： news

下载： download

滚动： scroll

标签页： tab

文章列表： list

指南： guild

服务： service

状态： status

投票： vote

合作伙伴： partner

友情链接：link

版权：copyright

#### （四）class 的命名

(1)颜色:使用颜色的名称或者16进制代码,如

```
.red { color: red; }  
.f60 { color: #f60; }  
.ff8600 { color: #ff8600; }
```

(2) 字体大小,直接使用"font+字体大小"作为名称,如

```
.font12px { font-size: 12px; }  
.font9pt { font-size: 9pt; }
```

(3)对齐样式,使用对齐目标的英文名称,如

```
.left { float:left; }  
.bottom { float:bottom; }
```

(4)标题栏样式,使用"类别+功能"的方式命名,如

```
.bar_news { }  
.bar_product { }
```

注意事项:

1. 尽量用英文;
2. 全部小写
3. Class命名使用下划线规则; 如.bar\_news { }
4. 尽量不缩写, 除非一看就明白的单词.

样式文件命名:

主要的 master.css

模块 module.css

基本共用 base.css

主题 themes.css

专栏 columns.css

打印 print.css

文字 font.css

表单 forms.css

补丁 mend.css

布局，版面 layout.css