# ****HTML编码规范****

## （一）命名规则：

 头：header

 内容：content/container

 尾：footer

 导航：nav

 侧栏：sidebar

 栏目：column

 页面外围控制整体布局宽度：wrapper

 左右中：left right center

 登录条：loginbar

 标志：logo

 广告：banner

 页面主体：main

 热点：hot

 新闻：news

 下载：download

 子导航：subnav

 菜单：menu

 子菜单：submenu

 搜索：search

 友情链接：friendlink

 页脚：footer

 版权：copyright

 滚动：scroll

 内容：content

 标签页：tab

 文章列表：list

 提示信息：msg

 小技巧：tips

 栏目标题：title

 加入：joinus

 指南：guild

 服务：service

 注册：regsiter

 状态：status

 投票：vote

 合作伙伴：partner

## (二)注释的写法:

 /\* Footer \*/

 内容区

 /\* End Footer \*/

## (三)id的命名:

##  (1)页面结构

 容器: container

 页头：header

 内容：content/container

 页面主体：main

 页尾：footer

 导航：nav

 侧栏：sidebar

 栏目：column

 页面外围控制整体布局宽度：wrapper

 左右中：left right center

##  (2)导航

 导航：nav

 主导航：mainbav

 子导航：subnav

 顶导航：topnav

 边导航：sidebar

 左导航：leftsidebar

 右导航：rightsidebar

 菜单：menu

 子菜单：submenu

 标题: title

 摘要: summary

##  (3) 功能

 标志：logo

 广告：banner

 登陆：login

 登录条：loginbar

 注册：regsiter

 搜索：search

 功能区：shop

 标题：title

 加入：joinus

 状态：status

 按钮：btn

 滚动：scroll

 标签页：tab

 文章列表：list

 提示信息：msg

 当前的: current

 小技巧：tips

 图标: icon

 注释：note

 指南：guild

 服务：service

 热点：hot

 新闻：news

 下载：download

 投票：vote

 合作伙伴：partner

 友情链接：link

 版权：copyright

# WEB前端命名规范

## 一、规范目的

### 1.1 概述

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, 经讨论决定后可以更改此文档.

## 二、文件规范

### 2.1 文件命名规则

文件名称统一用小写的英文字母、数字和下划线的组合，其中不得包含汉字、空格和特殊字符；命名原则的指导思想一是使得你自己和工作组的每一个成员能够方便的理解每一个文件的意义，二是当我们在文件夹中使用“按名称排例”的命令时，同一种大类的文件能够排列在一起，以便我们查找、修改、替换、计算负载量等等操作。

#### a. HTML的命名原则

引文件统一使用index.htm index.html index.asp文件名（小写）

各子页命名的原则首先应该以栏目名的英语翻译取单一单词为名称。例如：

关于我们 \ aboutus

信息反馈 \ feedback

产 品 \ product

如果栏目名称多而复杂并不好以英文单词命名，则统一使用该栏目名称拼音或拼音的首字母表示；

每一个目录中应该包含一个缺省的html 文件，文件名统一用index.htm index.html index.asp；

#### b. 图片的命名原则

图片的名称分为头尾两部分，用下划线隔开，头部分表示此图片的大类性质

例如：广告、标志、菜单、按钮等等。

放置在页面顶部的广告、装饰图案等长方形的图片取名： banner

标志性的图片取名为： logo

在页面上位置不固定并且带有链接的小图片我们取名为 button

在页面上某一个位置连续出现，性质相同的链接栏目的图片我们取名： menu

装饰用的照片我们取名： pic

不带链接表示标题的图片我们取名： title

范例：banner\_sohu.gif banner\_sina.gif menu\_aboutus.gif menu\_job.gif title\_news.gif logo\_police.gif logo\_national.gif pic\_people.jpg

鼠标感应效果图片命名规范为”图片名+\_+on/off”。

例如：menu1\_on.gif menu1\_off.gif

#### c. javascript的命名原则

例如：广告条的javascript文件名为 ad.js 弹出窗口的javascript文件名为 pop.js

#### d. 动态语言文件命名原则

以性质\_描述，描述可以有多个单词，用“\_”隔开，性质一般是该页面得概要。

范例：register\_form.asp register\_post.asp topic\_lock.asp

### 2.2 文件存放位置规范

\_Root

cn 存放中文HTML文件

en 存放英文HTML文件

flash 存放Flash文件

images 存放图片文件

imagestudio 存放PSD源文件

flashstudio 存放flash源文件

inc 存放include文件

library 存放DW库文件

media 存放多媒体文件

project 存放工程项目资料

temp 存放客户原始资料

js 存放JavaScript脚本

css 存放CSS文件

2.3 CSS 书写规范

#### 基本原则：

CSS样式可细分为3类：自定义样式、重新定义HTML样式、链接状态样式。

1. 样式为设计师自定义的新 CSS 样式，影响被使用本样式的区域，用于完成网页中局部的样式设定。样式名 “.”+“相应样式效果描述的单词或缩写”例：“ .shadow ”

文字样式样式名“.no”+“字号”+“行距”+“颜色缩写”例：“ .no12 ” 、“ .no12-24 ”

2. 义HTML样式为设计师重新定义已有的HTML标签样式，影响全部的被设定标签样式，用于统一网页中某一标签的样式定义。样式名“HTML标签”例：hr { border: 1px dotted #333333 }

3. 态样式为设计师对链接不同状态设定特殊样式，影响被使用本样式区域中的链接。

该样式写法有2种： a.nav:link nav.a:link 第一种只能修饰<a>标签中；第二种可以修饰所有包含有<a>标签的其他标签。

页面内的样式加载必须用链接方式<link rel=”stylesheet” type=”text/css” href=”style/style.css”>

#### 注意细则：

1. 协作开发及分工: i会根据各个模块, 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base.css由i书写, 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;

2. class与id的使用: id是唯一的并是父级的, class是可以重复的并是子级的, 所以id仅使用在大的模块上, class可用在重复使用率高及子级中; id原则上都是由我分发框架文件时命名的, 为JavaScript预留钩子的除外;

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

4. class与id命名: 大的框架命名比如header/footer/wrapper/left/right之类的在2中由i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.

5. 规避class与id命名(此条重要, 若有不明白请及时与i沟通):

a, 通过从属写法规避, 示例见d;

b, 取父级元素id/class命名部分命名, 示例见d;

c, 重复使用率高的命名, 请以自己代号加下划线起始, 比如i\_clear;

d, a,b两条, 适用于在2中已建好框架的页面, 如, 要在2中已建好框架的页面代码<div id=”mainnav”></div>中加入新的div元素,

按a命名法则: <div id=”mainnav”><div class=”firstnav”>…</div></div>,

样式写法: #mainnav .firstnav{…….}

按b命名法则: <div id=”mainnav”><div class=”main\_firstnav”>…</div></div>,

样式写法: .main\_firstnav{…….}

6. 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: margin、padding、float（包括clear）、position（相应的 top,right,bottom,left）、display、visibility、overflow等；自身属性主要包括: width & height & background & border; 文本属性主要包括：font、color、text-align、text-decoration、text-indent等；其他属性包括: list-style(列表样式)、vertical-vlign、cursor、z-index(层叠顺序) 、zoom等.我所列出的这些属性只是最常用到的, 并不代表全部;

7. 书写代码前, 考虑并提高样式重复使用率;

8. 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></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

9. 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;

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

11. 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文件中我会初始化表格样式)

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

13. 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:

background:none;\_filter:progid:DXImageTransform.Microsoft.AlphaImageLoader(sizingMethod=crop, src=’img/bg.png’);

14. 避免兼容性属性的使用, 比如text-shadow || css3的相关属性;

15. 减少使用影响性能的属性, 比如position:absolute || float ;

16. 必须为大区块样式添加注释, 小区块适量注释;

17. 代码缩进与格式: 建议单行书写, 可根据自身习惯, 后期优化i会统一处理;

#### （一）命名规则：

头：header

内容：content/container

尾：footer

导航：nav

侧栏：sidebar

栏目：column

页面外围控制整体布局宽度：wrapper

左右中：left right center

登录条：loginbar

标志：logo

广告：banner

页面主体：main

热点：hot

新闻：news

下载：download

子导航：subnav

菜单：menu

子菜单：submenu

搜索：search

友情链接：friendlink

页脚：footer

版权：copyright

滚动：scroll

内容：content

标签页：tab

文章列表：list

提示信息：msg

小技巧：tips

栏目标题：title

加入：joinus

指南：guild

服务：service

注册：regsiter

状态：status

投票：vote

合作伙伴：partner

#### (二)注释的写法:

/\* Footer \*/

内容区

/\* End Footer \*/

#### (三)id的命名:

(1)页面结构

容器: container

页头：header

内容：content/container

页面主体：main

页尾：footer

导航：nav

侧栏：sidebar

栏目：column

页面外围控制整体布局宽度：wrapper

左右中：left right center

(2)导航

导航：nav

主导航：mainbav

子导航：subnav

顶导航：topnav

边导航：sidebar

左导航：leftsidebar

右导航：rightsidebar

菜单：menu

子菜单：submenu

标题: title

摘要: summary

(3)功能

标志：logo

广告：banner

登陆：login

登录条：loginbar

注册：regsiter

搜索：search

功能区：shop

标题：title

加入：joinus

状态：status

按钮：btn

滚动：scroll

标签页：tab

文章列表：list

提示信息：msg

当前的: current

小技巧：tips

图标: icon

注释：note

指南：guild

服务：service

热点：hot

新闻：news

下载：download

投票：vote

合作伙伴：partner

友情链接：link

版权：copyright\

基本样式：

/\* CSS Document \*/

body {margin:0; padding:0; font:12px “\5B8B\4F53″,san-serif;background:#fff;}

div,dl,dt,dd,ul,ol,li,h1,h2,h3,h4,h5,h6,pre,form,fieldset,input,textarea,blockquote,p{padding:0; margin:0;}

table,td,tr,th{font-size:12px;}

li{list-style-type:none;}

img{vertical-align:top;border:0;}

ol,ul {list-style:none;}

h1,h2,h3,h4,h5,h6 {font-size:12px; font-weight:normal;}

address,cite,code,em,th {font-weight:normal; font-style:normal;}

.fB{font-weight:bold;}

.f12px{font-size:12px;}

.f14px{font-size:14px;}

.left{float:left;}

.right{float:right;}

a {color:#2b2b2b; text-decoration:none;}

a:visited {text-decoration:none;}

a:hover {color:#ba2636;text-decoration:underline;}

a:active {color:#ba2636;}

重定义的最先，伪类其次，自定义最后，便于自己和他人阅读！

不同浏览器上字号保持一致，字号建议用点数pt和像素px来定义，pt一般使用中文宋体的9pt 和11pt，px一般使用中文宋体12px 和14.7px 这是经过优化的字号，黑体字或者宋体字加粗时，一般选用11pt 和14.7px 的字号比较合适。中英文混排时，我们尽可能的将英文和数字定义为verdana 和arial 两种字体。

2.4 html 书写规范

1. 网页制作细节 —- head区代码规范

head区是指HTML代码的<head>和</head>之间的内容。

必须加入的标签

a) 公司版权注释 <!— The site is designed by EHM,Inc 07/2005 —>

b) 网页显示字符集

简体中文：<META HTTP-EQUIV=”Content-Type” CONTENT=”text/html; charset=gb2312″>

繁体中文：<META HTTP-EQUIV=”Content-Type” CONTENT=”text/html; charset=utf-8″>

英 语：<META HTTP-EQUIV=”Content-Type” CONTENT=”text/html; charset=utf-8″>

c) 网页制作者信息 <META name=”author” content=”webmaster@maketown.com”>

d) 网站简介 <META NAME=”DESCRIPTION” CONTENT=”xxxxxxxxxxxxxxxxxxxxxxxxxx”>

e) 搜索关键字 <META NAME=”keywords” CONTENT=”xxxx,xxxx,xxx,xxxxx,xxxx,”>

f) 网页的css规范 <LINK href=”../css/style.css” rel=”stylesheet” type=”text/css”>

g) 网页标题 <title>xxxxxxxxxxxxxxxxxx</title>

可以选择加入的标签

a) 设定网页的到期时间。一旦网页过期，必须到服务器上重新调阅。

<META HTTP-EQUIV=”expires” CONTENT=”Wed, 26 Feb 1997 08：21：57 GMT”>

b) 禁止浏览器从本地机的缓存中调阅页面内容。

<META HTTP-EQUIV=”Pragma” CONTENT=”no-cache”>

c) 用来防止别人在框架里调用你的页面。

<META HTTP-EQUIV=”Window-target” CONTENT=”\_top”>

d) 自动跳转。

<META HTTP-EQUIV=”Refresh” CONTENT=”5;URL=http：//www.yahoo.com”> 5指时间停留5秒

e) 网页搜索机器人向导。用来告诉搜索机器人哪些页面需要索引，哪些页面不需要索引。

<META NAME=”robots” CONTENT=”none”>

CONTENT的参数有all,none,index,noindex,follow,nofollow。默认是all。

f) 收藏夹图标 <link rel = “Shortcut Icon” href=”favicon.ico”>

g) 所有的javascript的调用尽量采取外部调用.

<SCRIPT LANGUAGE=”JavaScript” SRC=”script/xxxxx.js”></SCRIPT>

h) 附<body>标签：

<body>标签不属于head区，这里强调一下，为了保证浏览器的兼容性，必须设置页面背景<body bgcolor=”#FFFFFF”>

2. 网页制作细节 —- 字体

1. 在设定字体样式时对于文字字号样式和行间距应必须使用CSS样式表。禁止在页面中出现 <font size=?> 标记。

2.在网页中中文应首选使用宋体。英文和数字首选使用verdana 和arial 两种字体。一般使用中文宋体的9pt 和11pt 或12px 和14.7px 这是经过优化的字号，黑体字或者宋体字加粗时，一般选用11pt 和14.7px 的字号比较合适。

3. 为了最大程度的发挥浏览器自动排版的功能，在一段完整的文字中请尽量不要使用<br> 来人工干预分段。

4.不同语种的文字之间应该有一个半角空格，但避头的符号之前和避尾的符号之后除外，汉字之间的标点要用全角标点，英文字母和数字周围的括号应该使用半角括号。

5. 请不要在网页中连续出现多于一个的 也尽量少使用全角空格（英文字符集下，全角空格会变成乱码），空白应该尽量使用 text-indent, padding, margin, hspace, vspace 以及透明的gif 图片来实现。

6. 行距建议用百分比来定义，常用的两个行距的值是line-height:120%/150%.

7. 排版中我们经常会遇到需要进行首行缩进的处理，不要使用 或者全角空格来达到效果，规范的做法是在样式表中定义 p { text-indent: 2em; } 然后给每一段加上 <p> 标记，注意，一般情况下，请不要省略 </p> 结束标记 。

3. 网页制作细节 —- 链接

1. 网站中的链接路径全部采用相对路径，一般链接到某一目录下的缺省文件的链接路径不必写全名，如我们不必这样：<a href=”aboutus/index.htm”> 而应该这样：<a href=”aboutus/”>，所有内页指向首页的链接写成<a href=”/”>

2. 在浏览器里，当我们点击空链接时，它会自动将当前页面重置到首端，从而影响用户正常的阅读内容，我们用代码“javascript:void(null)”代替原来的“#”标记

4. 网页制作细节 —- 表格

1px表格 style=”border-collapse: collapse”

实例如下：

<table border=”1″ cellspacing=”0″ width=”32″ height=”32″ style=”border-collapse: collapse”

bordercolor=”#000000″ cellpadding=”0″>

<tr>

<td></td>

</tr>

</table>

设置亮、暗边框颜色

表格有亮边框（bordercolorlight）和暗边框（bordercolordark）两个属性可以对表格样式设置。

<table border=”1″ width=”500″ bordercolorlight=”#000000″ bordercolordark=”#FFFFFF”>

在写 <table> 互相嵌套时，严格按照的规范，对于单独的一个<table>来说，<table><tr>对齐，<td> 缩进两个半角空格，<td> 中如果还有嵌套的表格，<table>也缩进两个半角空格，如果<td>中没有任何嵌套的表格，</td> 结束标记应该与 <td> 处于同一行，不要换行，

如我们注意在源代码中不应有这样的代码：

<td><img src=”../images/sample.gif”>

</td>

而应该是这样的：

<td><img src=”../images/sample.gif”></td>

这是因为浏览器认为换行相当于一个半角空格，以上不规范的写法相当于无意中增加一个半角空格，如果确实有必要增加一个半角空格，也应该这样写：

<td><img src=”../images/sample.gif”> </td>

一个网页要尽量避免用整个一张大表格，所有的内容都嵌套在这个大表格之内，因为浏览器在解释页面的元素时，是以表格为单位逐一显示，如果一张网页是嵌套在一个大表格之内，那么很可能造成的后果就是，当浏览者敲入网址，他要先面对一片空白很长时间，然后所有的网页内容同时出现。如果必须这样做，请使用 <tbody>标记，以便能够使这个大表格分块显示

5. 网页制作细节 —- 下载速度

首页Flash 网页大小应限定在 200K 以下，尽可能的使用矢量图形和Action来减小动画大小。非首页静态页面含图片大小应限定在 70K 左右，尽可能的使用背景颜色替换大块同色图片。

6. 网页制作细节 —- include

asp标准写法 <!–#include file=”inc/index\_top.asp” –>

jsp 标准写法 <%@ include file=”../inc/index\_top..jsp” %>

7. 网页制作细节 —- Alt和Title

都是提示性语言标签，请注意它们之间的区别。

在我们浏览网页时，当鼠标停留在图片对象或文字链接上时，在鼠标的右下角有时会出现一个提示信息框。对目标进行一定的注释说明。在一些场合，它的作用是很重要的。

alt 用来给图片来提示的。Title用来给链接文字或普通文字提示的。

用法如下：

<p Title=”给链接文字提示”>文字</p>

<a href=”#” Title=”给链接文字提示”>文字</a>

<img src=”图片.gif” alt=”给图片提示”>

8. 网页制作细节 —- 缓存

网页不会被缓存

HTM网页

<META HTTP-EQUIV=”pragma” CONTENT=”no-cache”>

<META HTTP-EQUIV=”Cache-Control” CONTENT=”no-cache, must-revalidate”>

<META HTTP-EQUIV=”expires” CONTENT=”0″>

ASP网页

Response.Expires = -1

Response.ExpiresAbsolute = Now() – 1

Response.cachecontrol = “no-cache”

9. 网页制作细节 —- 浏览器兼容性

创建站点时，应该明白访问者可能使用各种 Web 浏览器。在已知的其他设计限制下，尽可能将站点设计为具有最大的浏览器兼容性。

目前使用的 Web 浏览器有二十多种，大多数已发行了多个版本。即使您只针对使用 Netscape Navigator 和 Microsoft Internet Explorer 的大多数 Web 用户，但您应明确并不是每个人都在使用这两种浏览器的最新版本。

您的站点越复杂（在布局、动画、多媒体内容和交互方面），跨浏览器兼容的可能性就越小。例如，并非所有的浏览器都可以运行JavaScript。不使用特殊字符的纯文本页面或许能够在任何浏览器中正确显示，但比起有效地使用图形、布局和交互的页面，这样的页面在美感上可能要差得多。所以，应尽量在最佳效果设计和最大浏览器兼容性设计之间取得平衡。

所有的HTML 标签的属性都要用单引号或者双引号括起，即我们应该写 <a href=”url”> 而不 是 <a href=url>.

10. 图片处理细节 —- banner

全尺寸banner为468X60px，半尺寸banner为234X60px，小banner为88X31px。

另外120X90，120X60也是小图标的标准尺寸。全尺寸banner不超过14K。

普遍的banner尺寸760X100，750X120，468X60，468X95，728X90，585X140

次级页的pip尺寸360X300，336X280

游标:100X100或120X120

11. 图片处理细节 —- LOGO的国际标准规范

为了便于INTERNET上信息的传播，一个统一的国际标准是需要的。实际上已经有了这样的一整套标准。其中关于网站的LOGO，目前有三种规格：

88\*31 这是互联网上最普遍的LOGO规格。

120\*60 这种规格用于一般大小的LOGO。

120\*90 这种规格用于大型LOGO。

12. 图片处理细节 —- 页面修饰图片处理

图片经过优化以加快下载的速度,有较佳的视觉空间效果，用图要与页面风格、页面内容相符；制作精美，细节处理得当。

2.5 JavaScript书写规范

1. 书写过程中, 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.

2.6 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img/demoimg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.

2.7 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 \*/;

2.8 浏览器兼容性 CSS hack

一、标识区别：

区别IE6,IE7,IE8,FF。

1. IE都能识别\* ; 标准浏览器(如FF)不能识别\*；

2. IE6能识别\*，但不能识别 !important; IE6在样式前面加\_

3. IE7能识别\*，也能识别!important;

4. IE8能识别\ 9 例如：background:red \9;

5. firefox不能识别\*，但能识别!important;

1．IE6和firefox的区别：

background:orange;\*background:blue;

意思就是火狐浏览器的背景颜色是橙色,而IE浏览器的背景色是蓝色.

2. IE6和IE7的区别：

background:green !important;background:blue;

意思指的是:IE7的背景颜色是绿色,IE6的背景颜色是蓝色

3. IE7和FF的区别：

background:orange; \*background:green;

意思指的是:火狐浏览器的背景颜色是橙色,而IE7的背景颜色是绿色

4. FF，IE7，IE6的区别：

background:orange;

\*background:green !important;

\*background:blue;

意思是火狐浏览器的的背景橙色,IE7浏览器的背景颜色是绿色,而IE6浏览器的颜色是蓝色.

二、实践建议

(1). 开发平台的选择

在 Firefox 上编写CSS, 同时兼容其他浏览器的. 这样做肯定会比在 IE 做好再到别的浏览器兼容来得容易, 因为 IE 对老标准支持还是很不错的, 而 IE 的一些特有功能人家却不支持. 所以推荐以 Firefox 结合 Firebug 扩展作为平台。

(2). CSS Hack 的顺序

使用 Firefox 作为平台, 只要代码写得够标准, 其实要 Hack 的地方不会很多的, IE 以外的浏览器几乎都不会有问题, 所以可以暂时忽略,

顺序如下：Firefox -> IE6 -> IE7 -> 其他

(3). Hack 的方法

说到方法有两种, 一种是在不同文件中处理, 另一种则是在同一个文件中处理. 其实作用是相同的, 只是出发点不一样而已.

1. 同一文件中处理.

如: id=”bgcolor”的控件要在 IE6中显示蓝色, IE7中显示绿色, Firefox等其他浏览器中显示红色。

IE6不认 !important,也不认 \*+html.所以 IE6只能是 blue.

IE7认 !important,也认 \*+html,优先度: (\*+html + !important) > !important > +html. IE7可以是 red, blue和 green,但 green的优先度最高.

Firefox和其他浏览器都认 !important. !important优先, Firefox可以是 red和 blue,但 red优先度高.上述的优先符号均是 CSS3标准允许的,其他浏览器也还有其他的 Hack方法,但我迄今还没遇到过 Firefox正常, IE以外的其他浏览器不正常的情况,所以无可分享.只要代码规范,相信这种情况的发生应该是很罕见 (JavaScript除外).

2. 不同文件中处理.

为什么同一文件中可以处理还要写在多个文件里面针对不同的浏览器?这是为了欺骗 W3C的验证工具,其实只需要两个文件,一个是针对所有浏览器的,一个只为 IE服务.将所有符合 W3C的代码写到一个里面去,而一些 IE中必须的,又不能通过 W3C验证的代码 (如: cursor:hand;)放到另一个文件中,再用下面的方法导入.

========================================================================================

Web 目录结构规范：

目录建立的原则：以最少的层次提供最清晰简便的访问结构。

目录的命名以小写英文字母，下划线组成。

根目录一般只存放htm以及其他必须的系统文件

根目录下的images用于存放各页面都要使用的公用图片。

所有JS等脚本存放在根目录下的scripts或js目录

所有CSS文件存放在根目录下style或css目录

多语言版本存放于独立的目录。例如：简体中文存放在cn目录下，繁体中文存放在 tw目录下，英文存放在en目录下等。

所有flash, avi, ram, quicktime 等多媒体文件存放在根目录下的media目录下。

Web UI设计命名规范：

2.1 Web 界面设计最佳尺寸

960网格系统——web界面设计最佳尺寸

目前绝大多数显示器都支持 1024 x 768 及其以上分辨率。为了有效的利用屏幕宽度同时保证栅格的灵活度，可以看出960是非常合适的。这样，在目前主流显示器下，960就成为网页栅格系统中的最佳宽度了。

2.2 网站设计及基本框架结构

Container

“container“ 就是将页面中的所有元素包在一起的部分，这部分还可以命名为: “wrapper“, “wrap“, “page“.

Header

“header” 是网站页面的头部区域，一般来讲，它包含网站的logo和一些其他元素。这部分还可以命名为:“page-header” (或 pageHeader).

Navbar

“navbar“等同于横向的导航栏，是最典型的网页元素。这部分还可以命名为:“nav”, “navigation”, “nav-wrapper”.

Menu

“Menu”区域包含一般的链接和菜单，这部分还可以命名为: “subNav “, “links“，“sidebar-main”.

Main

“Main”是网站的主要区域，如果是博客的话它将包含的日志。这部分还可以命名为: “content“, “main-content” (或“mainContent”)。

Sidebar

“Sidebar” 部分可以包含网站的次要内容，比如最近更新内容列表、关于网站的介绍或广告元素等…这部分还可以命名为: “subNav “, “side-panel“, “secondary-content“.

Footer

“Footer”包含网站的一些附加信息，这部分还可以命名为: “copyright“

2.3命名规则注意几点

尽量考虑为元素命名其本身的作用或”用意”，达到语义化。不要使用表面形式的命名.

如:red/left/big等。

组合命名规则:

[元素类型]-[元素作用/内容]

如:搜索按钮: btn-search、登录表单:form-login、新闻列表:list-news。

涉及到交互行为的元素命名:

凡涉及交互行为的元素通常会有正常、悬停、点击和已浏览等不同样式，命名可参考以下规则:

鼠标悬停:：hover 点击：click 已浏览：visited

如:搜索按钮: btn-search、btn-search-hover、btn-search-visited

2.4常用命名汇总

名称 命名规范

页头 header、top

登录条 loginbar

标志 logo

侧栏 sidebar

广告条 banner

导航: nav

子导航: subNav

菜单: menu

子菜单: subMenu

下拉菜单: dropMenu

工具条: toolbar

表单: form

栏目: column

箭头: arrow

搜索: search

搜索按钮: btn-search

滚动条: scroll

内容: content

标签页: tab

文章列表: list

小技巧: tips

栏目标题: title

链接: links

页脚: footer、end

服务: service

热点: hot

新闻: news

下载: download

注册: Register 、reg

状态: status

按钮: btn

上传: upload

登陆: login

投票: vote

合作伙伴: partner

版权: copyright

网站地图: sitemap

Web UI图片命名规范：

图片名称分为头尾两两部分，用下划线隔开。

头部分表示此图片的大类性质。

例如： 放置在页面顶部的广告、装饰图案等长方形的图片我们取名：banner ；标志性的图片我们取名为：logo ；在页面上位置不固定并且带有链接的小图片我们取名为button ；在页面上做栏目链接的图片我们取名：menu ；不带链接表示标题的图片我们取名：title ；装饰用的照片我们取名：pic ；依照此原则类推。

尾部分用来表示图片的具体含义，用英文字母表示。例如：gif logo\_lntu.gif button\_next.gif menu\_aboutus.gif title\_news.gif pic\_people.jpg

有onmouse效果的图片，两张分别在原有文件名后加”\_on”和”\_off”命名。

多语言的图片应放到多语言目录下

Web基本框架布局规范：

Web基本框架布局规范 常用到的Div和Table 两种

采用CSS+DIV对网站重构日趋被大家重视起来，有以下优势：

1：表现和内容相分离

将设计部分剥离出来放在一个独立样式文件中，HTML文件中只存放文本信息。

2：提高搜索引擎对网页的索引效率

用只包含结构化内容的HTML代替嵌套的标签，搜索引擎将更有效地搜索到你的网页内容，并可能给你一个较高的评价。

3：提高页面浏览速度

对于同一个页面视觉效果，采用CSS+DIV重构的页面容量要比TABLE编码的页面文件容量小得多，前者一般只有后者的1/2大小。

还有因为<table>标签是要等</table>下载好以后才可以显示的，所以整个网页是一下子跳出来的，这样让人感觉起来显示速度非常的慢。而使用css+div,<div> 标签不用等</div>下载好就可以显示里面的内容的,所以让人感觉打开网页的速度非常之快。

4：易于维护和改版

你只要简单的修改几个CSS文件就可以重新设计整个网站的页面。

从以上的描述来看，采用CSS+DIV对网站重构可以大大提升网站用户与搜索引擎的友好度。

CSS+DIV所以成为目前网页布局主流。

Table最大缺点是代码臃肿，你至少需要写下 <table><tr><td>这三个标签之后，才能开始真正的内容，另外，Table 的各种标签中还包含了复杂的属性定义，而 Div 只需 <div>一个标签。

页面渲染性能问题：浏览器需要将整个表格完全读完后才会开始渲染。

不利于搜索引擎优化：搜索引擎喜欢内容与修饰分开。

可访问性差：屏幕朗读软件和盲文浏览器无法很好地理解 Table 中的内容。

不够语义（Semantic）：我们需要语义的 Web。

一般table 只用做于数据处理。

CSS样式书写规范及优化：

5.1．CSS文件命名规范

全局样式：global.css；

框架布局：layout.css；

字体样式：font.css；

链接样式：link.css；

打印样式：print.css；

5.2．常用类/ID命名规范

页　眉：header

内　容：content

容　器：container

页　脚：footer

注　释：note

……（更多参照2.4常用命名汇总）

1、常用类的命名规则：

应尽量以常见英文单词为准，做到通俗易懂，并在适当的地方加以注释。对于二级类/ID命名，则采用组合书写的模式，后一个单词的首字母应大写：诸如“搜索框”则应命名为“searchInput”、“搜索图标”命名这“searchIcon”、“搜索按钮”命名为“searchBtn”……

2、辅助图片用背影图处理：

这里的”辅助图片”是指那些不是作为页面要表达的内容的一部分，而仅仅用于修饰、间隔、提醒的图片。将其做背影图处理，可以在不改动页面的情况下通过CSS样式来进行改动，如：#logo {background:url(images/logo.jpg) #FEFEFE no-repeat right bottom;}

5.3结构与样式分离：

在页面里只写入文档的结构，而将样式写于css文件中，通过外部调用CSS样式表来实现结构与样式的分离。

5.4、文档的结构化书写：

页面CSS文档都应采用结构化的书写方式，逻辑清晰易于阅读。如：

<div id=”mainMenu”>

<ul>

<li><a href=”#” >首页</a></li>

<li><a href=”#” >介绍</a></li>

<li><a href=”#” >服务</a></li>

</ul>

</div>

/\*=====主导航=====\*/

#mainMenu {

width:100%;

height:30px;

background:url(images/mainMenu\_bg.jpg) repeat-x;

}

#mainMenu ul li {

float:left;

line-height:30px;

margin-right:1px;

cursor:pointer;

}

/\*=====主导航结束=====\*/

5.5注释书写规范

1、行间注释：

直接写于属性值后面，如：

.search{

border:1px solid #fff;/\*定义搜索输入框边框\*/

background:url(../images/icon.gif) no-report #333;/\*定义搜索框的背景\*/

}

2、整段注释：

分别在开始及结束地方加入注释，如：

/\*=====搜索条=====\*/

.search {

border:1px solid #fff;

background:url(../images/icon.gif) no-repeat #333;

}

/\*=====搜索条结束=====\*/

5.6 CSS样式属性代码优化缩写

1、不同类有相同属性及属性值的缩写：

对于两个不同的类，但是其中有部分相同甚至是全部相同的属性及属性值时，应对其加以合并缩写，特别是当有多个不同的类而有相同的属性及属性值时，合并缩写可以减少代码量并易于控制。如：

#mainMenu {

background:url(../images/bg.gif);

border:1px solid #333;

width:100%;

height:30px;

overflow:hidden;

}

#subMenu {

background:url(../images/bg.gif);

border:1px solid #333;

width:100%;

height:20px;

overflow:hidden;

}

两个不同类的属性值有重复之处，刚可以缩写为：

#mainMenu,#subMenu {

background:url(../images/bg.gif);

border:1px solid #333;

width:100%;

overflow:hidden;

}

#mainMenu {height:30px;}

#subMenu {height:20px;}

2、同一属性的缩写：

同一属性根据它的属性值也可以进行简写，如：

.search {

background-color:#333;

background-image:url(../images/icon.gif);

background-repeat: no-repeat;

background-position:50% 50%;

}

.search {

background:#333 url(../images/icon.gif) no-repeat 50% 50%;

}

3、内外侧边框的缩写：

在CSS中关于内外侧边框的距离是按照上、右、下、左的顺序来排列的，当这四个属性值不同时也可直接缩写，如：

.btn {

margin-top:10px;

margin-right:8px;

margin-bottom:12px;

margin-left:5px;

padding-top:10px;

padding-right:8px;

padding-bottom:12px;

padding-left:8px;

}

则可缩写为：

.btn {

Margin:10px 8px 12px 5px;

Padding:10px 8px 12px 5px;

}

而如果当上边与下边、左边与右边的边框属性值相同时，则属性值可以直接缩写为两个，如：

.btn {

margin-top:10px;

margin-right:5px;

margin-bottom:10px;

margin-left:5px;

}

缩写为：.btn {margin:10px 5px;}

而当上下左右四个边框的属性值都相同时，则可以直接缩写成一个，如：

.btn {

margin-top:10px;

margin-right:10px;

margin-bottom:10px;

margin-left:10px;

}

缩写为：.btn{margin:10px;}

4、颜色值的缩写：

当RGB三个颜色值数值相同时，可缩写颜色值代码。如：

.menu { color:#ff3333;}

可缩写为:.menu {color:#f33;}

5.7 CSS各浏览器解决不兼容问题

因为不同浏览器对W3C标准的支持不一样，各个浏览器对于页面的解释呈视也不尽相同，比如IE在很多情况下就与FF存在3px的差距，对于这些差异性，就需要利用css 的hack来进行调整，当然在没有必要的情况下，最好不要写hack来进行调整，避免因为hack而导致页面出现问题。

1、 IE6、IE7、Firefox之间的兼容写法：

写法一：

IE都能识别\*;标准浏览器(如FF)不能识别\*；

IE6能识别\*，但不能识别 !important,

IE7能识别\*，也能识别!important;

FF不能识别\*，但能识别!important;

根据上述表达，同一类/ID下的CSS 　hack可写为：

.searchInput {

background-color:#333;/\*三者皆可\*/

\*background-color:#666　!important; /\*仅IE7\*/

\*background-color:#999; /\*仅IE6及IE6以下\*/

}

一般三者的书写顺序为：FF、IE7、IE6.

写法二：

IE6可识别“\_”，而IE7及FF皆不能识别，所以当只针对IE6与IE7及FF之间的区别时，可这样书写：

.searchInput {

background-color:#333;/\*通用\*/

\_background-color:#666;/\*仅IE6可识别\*/

}

写法三：

\*+html 与 \*html 是IE特有的标签, Firefox 暂不支持。

.searchInput {background-color:#333;}

\*html .searchInput {background-color:#666;}/\*仅IE6\*/

\*+html .searchInput {background-color:#555;}/\*仅IE7\*/

屏蔽IE浏览器：

select是选择符，根据情况更换。第二句是MAC上safari浏览器独有的。

\*:lang(zh) select {font:12px !important;} /\*FF的专用\*/

select:empty {font:12px !important;} /\*safari可见\*/

IE6可识别：

这里主要是通过CSS注释分开一个属性与值，注释在冒号前。

select { display /\*IE6不识别\*/:none;}

IE的if条件hack写法：

所有的IE可识别：

<!–[if IE]> Only IE <![end if]–>

只有IE5.0可以识别:

<!–[if IE 5.0]> Only IE 5.0 <![end if]–>

IE5.0包换IE5.5都可以识别:

<!–[if gt IE 5.0]> Only IE 5.0+ <![end if]–>

仅IE6可识别:

<!–[if lt IE 6]> Only IE 6- <![end if]–>

IE6以及IE6以下的IE5.x都可识别:

<!–[if gte IE 6]> Only IE 6/+ <![end if]–>

仅IE7可识别:

<!–[if lte IE 7]> Only IE 7/- <![end if]–>

2、清除浮动：

在Firefox中，当子级都为浮动时，那么父级的高度就无法完全的包住整个子级，那么这时用这个清除浮动的HACK来对父级做一次定义，那么就可以解决这个问题。

select:after {

content:”.”;

display:block;

height:0;

clear:both;

visibility:hidden;

}

其他规范：

js的命名原则以功能的英语单词为名。例如：广告条的js文件名为：js.

所有的CSS的尽量采用外部调用.

所有的javascript脚本尽量采取外部调用.

Web 页面编码一律统一用UTF-8编码：UTF-8是世界性通用代码，也完美的支持中文编码，如果我们做的网站能让国外用户正常的访问，就最好用UTF-8。

字体符号用像素px来定义，px使用中文宋体12px 和14px黑体

字体采用：Arial, Helvetica, sans-serif,宋体。

所有连接使用相对路径/images/，切记不可使用绝对路径。如：../images/

所有文件，目录，图片全部以小写字母命名，禁止用中文命名。

清理无效的链接和图片