

# Web 前端开发规范手册

## 目录

修订历史记录.....	错误！未定义书签。
一、规范目的.....	2
1.1 概述.....	2
二、基本准则.....	2
三、文件规范.....	2
2.1 文件命名规则.....	2
2.1.1 HTML 的命名原则.....	3
2.1.2 图片的命名原则.....	3
2.1.3. javascript 的命名原则.....	3
2.1.4 动态语言文件命名原则.....	3
2.2 文件存放位置规范.....	4
2.3 CSS 书写规范.....	4
2.3.1 基本原则.....	4
2.3.2 注意细则.....	5
2.3.3 命名规则.....	6
2.4 html 书写规范.....	9
2.4.1 head 区代码规范.....	9
2.4.2 body 区代码规范.....	10
2.5 JavaScript 书写规范.....	10

2.6 图片规范.....	10
2.7 注释规范.....	11
2.7.1 html 注释.....	11
2.7.2 css 注释.....	11
2.7.3 JavaScript 注释.....	11
四、执行模式.....	11

## 一、规范目的

### 1.1 概述

- 提高团队协作效率
- 便于前端开发以及后期优化维护
- 方便新进的成员快速上手
- 输出高质量的代码

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新此文档。

## 二、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。
  - 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。
- 开发时需要遵循如上基本准则 特殊情况可以有所宽限 ,如一些老项目的页面改造。

## 三、文件规范

### 2.1 文件命名规则

[使用场景：在新建网页、图片、脚本、CSS 文件时，根据此规则给文件命名并放入指定位置]

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“按名称排列”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。

### 2.1.1 HTML 的命名原则

索引文件统一使用 **index.htm** **index.html** **index.asp** 文件名。

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：

子页	命名
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**。

### 2.1.2 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**

### 2.1.3. javascript 的命名原则

一般使用脚本功能的英文小写缩写命名。

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

### 2.1.4 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

文件夹	说明
flash	存放 flash 文件
images	存放图片文件
inc	存放 include 文件
library	存放 DW 库文件
media	存放多媒体文件
script	存放 javascript 脚本
css	存放 css 文件

## 2.3 CSS 书写规范

[使用场景：书写 CSS 样式时，根据此规则对 id 和 class 进行命名，并遵循书写规则，避免出现常犯错误，提高效率。]

### 2.3.1 基本原则

CSS 样式可细分为 3 类：重新定义 HTML 样式、链接状态样式、自定义样式。

1. 重新定义 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 }**

2. 链接状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**(中间无空格)、**.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。

3. 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**”

“**.no12**”类型一般出现在 base.css 文件中，只可选用，不可修改，避免名称含义和实际内容不符。自定义的样式名最好用意义命名。

比如：重要的新闻高亮显示（像红色），有两种写法：

```
red{color:red} ;
```

```
.important_news{color:red} ;
```

很显然第二种传达的意义更加明确，所以尽量不要用意义不明确的作为自定义命名。

4. 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">
```

5. 重新定义的最先，伪类其次，自定义样式最后，便于自己和他人阅读。

6. 一般不允许将样式的定义写在标签中（如：**style="color:red;"**）；
7. 所有装饰性的图片（如背景图），要定义在 CSS 中；

### 2.3.2 注意细则

1. class与id命名：大的框架命名比如header/footer/wrapper/left/right之类的由设计框架之人统一命名。其他样式名称由**小写英文&数字&\_**来组合命名，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。

2. 规避 class 与 id 重名，不建议使用 id 选择器。

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，可加代号加下划线起始，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{.....}`

3. 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。

文件夹	说明
布局定位属性	Margin\padding\float\clear\position ( 相应的 top,right,bottom,left ) \display\visibility\overflow 等
自身属性	Width\height\background\ border
文本属性	font\color\text-align\text-decoration\text-indent\ white-space\othertext\content 等
其他属性	list-style(列表样式)\vertical-align\cursor\z-index(层叠顺序)\zoom 等

4. 书写代码前，考虑并提高样式重复使用率；

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

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

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

8. 重要图片加上 alt 属性；重要的元素和截断的元素加上 title；

9. **避免使用!important**，!important 的优先级最高；

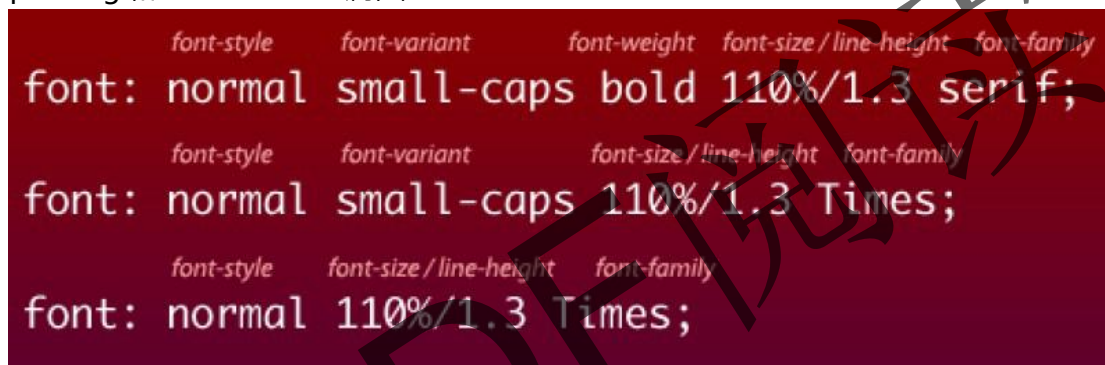
10. 必须为大区块样式添加注释，小区块适量注释；

11. 代码缩进与格式：建议单行书写，可根据自身习惯，后期优化统一处理；

12. Margin、padding、border、font 等样式尽可能采用合并写法；



上面的简写图解说明的是依据不同的属性值可以进行不同的缩写，这种方法同样适用于 `padding` 和 `Border-width` 属性;



Font 的简写对于缩小代码量很有帮劣，如果没有定义某个属性值的话，将继承父级元素的定义或使用默认值。

### 2.3.3 命名规则

页面结构	
容器: container/wrap	整体宽度: wrapper
页头: header	内容: content
页面主体: main	页尾: footer
侧栏: sidebar	栏目: column
中间内容: center	
导航	
导航: nav	主导航: mainbav/globalnav
子导航: subnav	顶导航: topnav
边导航: sidebar	左导航: leftsidebar
右导航: rightsidebar	边导航图标: sidebarlcon
菜单: menu	子菜单: submenu

标题: title	
<b>功能</b>	
标志: logo	登陆: login
登录条: loginbar	注册: regsite
产品: products	产品价格: products_prices
产品评论: products_review	编辑评论: editor_review
最新产品: news_release	最新产品: news_release
广告/标语: banner	摘要: summary
生产商: publisher	缩略图: screenshot
常见问题: faqs	关键词: keyword
博客: blog	论坛: forum
搜索: search	搜索输入框: search_input
搜索输出: search_output	搜索结果: search_results
加入我们: joinus	状态: status
按钮: btn	滚动: scroll
标签页: tab	文章列表: list
提示信息: msg/message	当前的: current
小技巧: tips	皮肤: skin
充值: pay	活动: activities
推广: promotion	公告: announcement
排行: ranking	公司简介: company_profile
公司设备: equipment	公司荣誉: glories
企业文化: culture	企业规模: scale
营销网络: sales_network	组织机构: organization
技术力量: technology	分支机构: branches
企业资质: enterprise_qualification	公司实力: strength_strength
经营理念: operation_principle	经理致辞: manager_oration
发展历程: development_history	工程案例: engineering_projects
分类浏览: browse_by_category	应用领域: application_fields
人力资源: human_resource_hr	领导致辞: leader_oration
客户留言: customer_message	客户服务: customer_service
您的要求: your_requirements	销售信息: sales_information
招商: enterprise_establishing	教育培训: education_training
在线交流: online_communication	质量认证: quality_certification
合作加盟: joinIn_cooperation	产品描述: products_description
业务范围: business_scope	产品销售: sales_sales
联系我们: contact_us	信息发布: information



返回首页：homepage	产品订购：order
电子商务：e_business	版权所有：copy_right
友情连结：hot_link	行业新闻：trade_news
行业动态：trends	邮编：postal_code_zipcode
新闻动态：news_trends	公司名称：company_name
销售热线：sales_hotline	联系人：contact_person
建设中：in_construction	证书：certificate
地址：add	电话：tel
传真：fax	产品名称：product_name
产品说明：description	价格：price
品牌：brand	规格：specification
尺寸：size	生产厂家：manufacuturer
型号：model	产品标号：item_no
技术指标：technique_data	产品描述：description
产地：production_place	用途：application
论坛：forum	在线订购：on_line_order
招标：bidInviting	综述：general
业绩：achievements	大事：great_event
动态：trends	服务：service
投资：investment	行业：industry
规划：programming	环境：environment
发送：delivery	提交：submit
重写：reset	社区：community
业务：business	在线调查：online_inquiry
下载中心：download	意见反馈：feedback
常见问题：faq	中心概况：general_profile
游乐园：amusement_park	专题报道：special_report
图标：icon	注释：note
指南：guild	服务：service
热点：hot	新闻：news
下载：download	投票：vote
商标：label/branding	当前位置：breadcrumb/loc
购物车：shop	标签：tag
信誉：siteinfo_credits	网站信息：siteinfo
法律声明：siteinfo_legal	合作伙伴：partner
友情链接：friendlink	版权：copyright



## 2.4 html 书写规范

[使用场景：书写 html 页面时，根据此规则对 html 内容进行补充和完善。]

### 2.4.1 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

#### 1. 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">
```

#### 2. 搜索关键字（网站类型）

```
< meta name="keywords" content="xxxx,xxxx,xxx,xxxxx,xxx,">
```

#### 3. 网页的 css 规范 <link href=" ../css/style.css" rel="stylesheet" type="text/css">

#### 4. 网页标题 <title>xxxxxxxxxxxxxxxxxxxx</title>

➤ 可以选择加入的标签

#### 1. 公司版权注释

```
<!-- The site is designed by EHM, Inc 07/2005 --->
```

#### 2. 网页制作者信息

```
<meta name="author" content="webmaster@maketown.com">
```

#### 3. 网站简介

```
<meta name="DESCRIPTION" content="xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx">
```

#### 4. 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">
```

#### 5. 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">
```

#### 6. 用来防止别人在框架里调用你的页面。

```
<meta http-equiv="Window-target" content="_top">
```

#### 7. 自动跳转。

```
<meta http-equiv="Refresh" content="5;url=http://www.yahoo.com">
```

5 指时间停留 5 秒

#### 8. 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。

#### 9. 收藏夹图标 <link rel="Shortcut Icon" href="favicon.ico">

#### 10. 所有的 javascript 的调用尽量采取外部调用。

```
<script language="JavaScript" src="script/xxxxx.js"></script>
```

## 2.4.2 body 区代码规范

### 1. 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">（此处为示意，样式需写在 CSS 文件中）

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。

```
<a href="javascript:void(null)">链接</a>
```

## 2.5 JavaScript 书写规范

*[使用场景：书写 js 脚本时，根据此规则对变类、函数、变量命名，并遵循书写规则，避免出现常犯错误，提高脚本效率。]*

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，以及组件组开发的相关库。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 私有变量要求首字符为 '\_'，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，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

*[使用场景：引用添加图片文件时，根据此规则对图片格式进行选择以及命名存放。]*

1. 所有页面元素类图片均放入 images 文件夹，测试用图片放于 images/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 images 目录下.

## 2.7 注释规范

[使用场景: 书写代码时, 根据此规则对代码进行适当注释说明, 方便自己或他人阅读和理解。]

### 2.7.1 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。

### 2.7.2 css 注释

1. 行间注释: 直接写于属性值后面。

例: `.search{ border:1pxsolid#fff;/*定义搜索输入框边框*/  
background:url(..images/icon.gif)no-repeat#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。

例:  
`/*=====搜索条=====*/  
.search{ border:1pxsolid#fff;  
background:url(..images/icon.gif)no-repeat#333;}  
/*=====搜索条结束=====*/`

### 2.7.3 JavaScript 注释

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 */;`

## 四、执行模式

- 规范文档存入部门 SVN 文件库, 方便下载查阅。
- 根据规范内容, 制作网页格式帮助手册。
- 根据规则建立和不断完善框架模板库, 给不同类型的网站提供示例参考。
- 执行规范过程中根据实际情况不断补充完善规则。
- 开发负责人需要检查提交的代码是否符合规范, 并记录检查结果。

以上就是本文的全部内容, 希望对大家的学习有所帮助, 有任何疑问可以给小编我留言, 谢谢! 当然了, 想多多了解实战技术可以加小编微信 (li18325593873) 一起研讨技术行业的博大精深。小编也整理了一些全套高级资料 (如 web, JavaScript, Html5, CSS3, JS, jQuery, Nodejs, Vuejs, Angular 等 后台相关: PHP, Mysql, Redis, MongoDB, 等一系列的相关前端知识点, 欢迎小伙伴一起来研讨技术。), 当然啦, 这些资料整理出来的目的不都是为了分享出去嘛, 可以直接加微信领取! (备注领资料)