**命名规范**

**js 命名规范**

**1. 变量命名规范**

变量名包括全局变量，局部变量，类变量，函数参数等等，他们都属于这一类。

**基本规范**

变量命名都以类型前缀+有意义的单词组成，单词首字母都需要大写。例如：sUserName，nCount。

**前缀规范**

每个局部变量都需要有一个类型前缀，按照类型可以分为：

s：表示字符串。例如：sName，sHtml；

n：表示数字。例如：nPage，nTotal；

b：表示逻辑。例如：bChecked，bHasLogin；

a：表示数组。例如：aList，aGroup；

r：表示正则表达式。例如：rDomain，rEmail；

f：表示函数。例如：fGetHtml，fInit；

g：表示全局变量。例如gUserName，gLoginTime。

o：表示以上未涉及到的其他对象，例如：oButton，oDate；

例外情况：

1：作用域不大临时变量可以简写，比如：str，num，bol，obj，fun，arr。

2：循环变量可以简写，比如：i，j，k等。

**常量规范**

常量，全部字母都大写。例如：COPYRIGHT，PI。

**2. 函数命名规范**

统一使用动词或者动词[＋名词]形式，例如：fGetVersion()，fSubmitForm()，fInit()；涉及返回逻辑值的函数可以使用is，has等表示逻辑的词语代替动词。

如果有内部函数，使用\_\_f+动词[+名词]形式，内部函数必需在函数最后定义。

**对象方法实现**

对象方法命名使用 f+对象类名+动词[+名词]形式；例如 fAddressGetEmail

**事件响应函数**

某事件响应函数命名方式为触发事件对象名＋事件名或者模块名＋触发事件对象名＋事件名，例如：fDivClick()，fAddressSubmitButtonClick()

参考<http://blog.jobbole.com/19197/>

**Css编码规范**

* 为选择器分组时，将单独的选择器单独放在一行。
* 为了代码的易读性，在每个声明块的左花括号前添加一个空格。
* 声明块的右花括号应当单独成行。
* 每条声明语句的 : 后应该插入一个空格。
* 为了获得更准确的错误报告，每条声明都应该独占一行。
* 所有声明语句都应当以分号结尾。最后一条声明语句后面的分号是可选的，但是，如果省略这个分号，你的代码可能更易出错。
* 对于以逗号分隔的属性值，每个逗号后面都应该插入一个空格（例如，box-shadow）。
* 不要在 rgb()、rgba()、hsl()、hsla() 或 rect() 值的内部的逗号后面插入空格。这样利于从多个属性值（既加逗号也加空格）中区分多个颜色值（只加逗号，不加空格）。
* 对于属性值或颜色参数，省略小于 1 的小数前面的 0 （例如，.5 代替 0.5；-.5px 代替 -0.5px）。
* 十六进制值应该全部小写，例如，#fff。在扫描文档时，小写字符易于分辨，因为他们的形式更易于区分。
* 尽量使用简写形式的十六进制值，例如，用 #fff 代替 #ffffff。
* 为选择器中的属性添加双引号，例如，input[type="text"]。[只有在某些情况下是可选的](http://mathiasbynens.be/notes/unquoted-attribute-values" \l "css)，但是，为了代码的一致性，建议都加上双引号。
* 避免为 0 值指定单位，例如，用 margin: 0; 代替 margin: 0px;。
* /\* Bad CSS \*/
* .selector, .selector-secondary, .selector[type=text] {
* padding:15px;
* margin:0px 0px 15px;
* background-color:rgba(0, 0, 0, 0.5);
* box-shadow:0px 1px 2px #CCC,inset 0 1px 0 #FFFFFF
* }
* /\* Good CSS \*/
* .selector,
* .selector-secondary,
* .selector[type="text"] {
* padding: 15px;
* margin-bottom: 15px;
* background-color: rgba(0,0,0,.5);
* box-shadow: 0 1px 2px #ccc, inset 0 1px 0 #fff;
* }

### 声明顺序

相关的属性声明应当归为一组，并按照下面的顺序排列：

1. Positioning
2. Box model
3. Typographic
4. Visual

由于定位（positioning）可以从正常的文档流中移除元素，并且还能覆盖盒模型（box model）相关的样式，因此排在首位。盒模型排在第二位，因为它决定了组件的尺寸和位置。

其他属性只是影响组件的内部（inside）或者是不影响前两组属性，因此排在后面。

完整的属性列表及其排列顺序请参考 [Recess](http://twitter.github.com/recess)。

.declaration-order {

/\* Positioning \*/

position: absolute;

top: 0;

right: 0;

bottom: 0;

left: 0;

z-index: 100;

/\* Box-model \*/

display: block;

float: right;

width: 100px;

height: 100px;

/\* Typography \*/

font: normal 13px "Helvetica Neue", sans-serif;

line-height: 1.5;

color: #333;

text-align: center;

/\* Visual \*/

background-color: #f5f5f5;

border: 1px solid #e5e5e5;

border-radius: 3px;

/\* Misc \*/

opacity: 1;

}

### 不要使用 @import

与 <link> 标签相比，@import 指令要慢很多，不光增加了额外的请求次数，还会导致不可预料的问题。替代办法有以下几种：

* 使用多个 <link> 元素
* 通过 Sass 或 Less 类似的 CSS 预处理器将多个 CSS 文件编译为一个文件
* 通过 Rails、Jekyll 或其他系统中提供过 CSS 文件合并功能

请参考 [Steve Souders 的文章](http://www.stevesouders.com/blog/2009/04/09/dont-use-import/)了解更多知识。

### 媒体查询（Media query）的位置

将媒体查询放在尽可能相关规则的附近。不要将他们打包放在一个单一样式文件中或者放在文档底部。如果你把他们分开了，将来只会被大家遗忘。下面给出一个典型的实例。

.element { ... }

.element-avatar { ... }

.element-selected { ... }

@media (min-width: 480px) {

.element { ...}

.element-avatar { ... }

.element-selected { ... }

}

### 带前缀的属性

当使用特定厂商的带有前缀的属性时，通过缩进的方式，让每个属性的值在垂直方向对齐，这样便于多行编辑。

/\* Prefixed properties \*/

.selector {

-webkit-box-shadow: 0 1px 2px rgba(0,0,0,.15);

box-shadow: 0 1px 2px rgba(0,0,0,.15);

}

### 单行规则声明

对于**只包含一条声明**的样式，为了易读性和便于快速编辑，建议将语句放在同一行。对于带有多条声明的样式，还是应当将声明分为多行。

这样做的关键因素是为了错误检测 -- 例如，CSS 校验器指出在 183 行有语法错误。如果是单行单条声明，你就不会忽略这个错误；如果是单行多条声明的话，你就要仔细分析避免漏掉错误了。

/\* Single declarations on one line \*/

.span1 { width: 60px; }

.span2 { width: 140px; }

.span3 { width: 220px; }

### 简写形式的属性声明

在需要显示地设置所有值的情况下，应当尽量限制使用简写形式的属性声明。常见的滥用简写属性声明的情况如下：

* padding
* margin
* font
* background
* border
* border-radius

大部分情况下，我们不需要为简写形式的属性声明指定所有值。例如，HTML 的 heading 元素只需要设置上、下边距（margin）的值，因此，在必要的时候，只需覆盖这两个值就可以。过度使用简写形式的属性声明会导致代码混乱，并且会对属性值带来不必要的覆盖从而引起意外的副作用。

MDN（Mozilla Developer Network）上一片非常好的关于[shorthand properties](https://developer.mozilla.org/en-US/docs/Web/CSS/Shorthand_properties) 的文章，对于不太熟悉简写属性声明及其行为的用户很有用。

/\* Bad example \*/

.element {

margin: 0 0 10px;

background: red;

background: url("image.jpg");

border-radius: 3px 3px 0 0;

}

/\* Good example \*/

.element {

margin-bottom: 10px;

background-color: red;

background-image: url("image.jpg");

border-top-left-radius: 3px;

border-top-right-radius: 3px;

}

**class 命名**

* class 名称中只能出现小写字符和破折号（dashe）（不是下划线，也不是驼峰命名法）。破折号应当用于相关 class 的命名（类似于命名空间）（例如，.btn 和 .btn-danger）。
* 避免过度任意的简写。.btn 代表 *button*，但是 .s 不能表达任何意思。
* class 名称应当尽可能短，并且意义明确。
* 使用有意义的名称。使用有组织的或目的明确的名称，不要使用表现形式（presentational）的名称。
* 基于最近的父 class 或基本（base） class 作为新 class 的前缀。
* 使用 .js-\* class 来标识行为（与样式相对），并且不要将这些 class 包含到 CSS 文件中。

在为 Sass 和 Less 变量命名是也可以参考上面列出的各项规范。

**选择器**

* 对于通用元素使用 class ，这样利于渲染性能的优化。
* 对于经常出现的组件，避免使用属性选择器（例如，[class^="..."]）。浏览器的性能会受到这些因素的影响。
* 选择器要尽可能短，并且尽量限制组成选择器的元素个数，建议不要超过 3 。
* **只有**在必要的时候才将 class 限制在最近的父元素内（也就是后代选择器）（例如，不使用带前缀的 class 时 -- 前缀类似于命名空间）。
* /\* Bad example \*/
* span { ... }
* .page-container #stream .stream-item .tweet .tweet-header .username { ... }
* .avatar { ... }
* /\* Good example \*/
* .avatar { ... }
* .tweet-header .username { ... }
* .tweet .avatar { ... }

### 编辑器配置

将你的编辑器按照下面的配置进行设置，以避免常见的代码不一致和差异：

* 用两个空格代替制表符（soft-tab 即用空格代表 tab 符）。
* 保存文件时，删除尾部的空白符。
* 设置文件编码为 UTF-8。
* 在文件结尾添加一个空白行。

参照文档并将这些配置信息添加到项目的 .editorconfig 文件中。例如：[Bootstrap 中的 .editorconfig 实例](https://github.com/twbs/bootstrap/blob/master/.editorconfig)。更多信息请参考 [about EditorConfig](http://editorconfig.org/)。

### 1. css样式文件名：

1）

主要的 master.css   
布局，版面 layout.css   
专栏 columns.css   
文字 font.css   
打印样式 print.css   
主题 themes.[css](http://cpro.baidu.com/cpro/ui/uijs.php?adclass=0&app_id=0&c=news&cf=1001&ch=0&di=128&fv=18&is_app=0&jk=ac5a3e290aff7d7f&k=css&k0=css&kdi0=0&luki=10&mcpm=0&n=10&p=baidu&q=smileking_cpr&rb=0&rs=1&seller_id=1&sid=7f7dff0a293e5aac&ssp2=1&stid=9&t=tpclicked3_hc&td=1682280&tu=u1682280&u=http%3A%2F%2Fwww%2Eth7%2Ecn%2Fweb%2Fhtml%2Dcss%2F201507%2F112357%2Eshtml&urlid=0)

2）

reset.css // 对[浏览器](http://cpro.baidu.com/cpro/ui/uijs.php?adclass=0&app_id=0&c=news&cf=1001&ch=0&di=128&fv=18&is_app=0&jk=ac5a3e290aff7d7f&k=%E4%AF%C0%C0%C6%F7&k0=%E4%AF%C0%C0%C6%F7&kdi0=0&luki=3&mcpm=0&n=10&p=baidu&q=smileking_cpr&rb=0&rs=1&seller_id=1&sid=7f7dff0a293e5aac&ssp2=1&stid=9&t=tpclicked3_hc&td=1682280&tu=u1682280&u=http%3A%2F%2Fwww%2Eth7%2Ecn%2Fweb%2Fhtml%2Dcss%2F201507%2F112357%2Eshtml&urlid=0" \t "_blank)的默认样式进行重设   
layout.css // 管理页面的布局   
typeset.css // 图文的编排  
color.css // 统一管理颜色的搭配   
print.css // 打印效果样式   
ie.css // 把对ie的hack单独分开

3）

reset.css   
header.css // 头部的所有样式   
container.css // 除头部/底部外的中间区域样式   
footer.css // 底部样式   
print.css   
ie.css

4)

reset.css /\*页面样式充值\*/  
header.css /\*全站头部样式\*/  
footer.css /\*全站尾部样式\*/  
public.css /\*全站公共模块样式\*/  
index.css /\*首页特有样式\*/  
container.css /\*二级及以下主体样式\*/  
print.css /\*打印样式\*/  
ie.[css](http://cpro.baidu.com/cpro/ui/uijs.php?adclass=0&app_id=0&c=news&cf=1001&ch=0&di=128&fv=18&is_app=0&jk=ac5a3e290aff7d7f&k=css&k0=css&kdi0=0&luki=10&mcpm=0&n=10&p=baidu&q=smileking_cpr&rb=0&rs=1&seller_id=1&sid=7f7dff0a293e5aac&ssp2=1&stid=9&t=tpclicked3_hc&td=1682280&tu=u1682280&u=http%3A%2F%2Fwww%2Eth7%2Ecn%2Fweb%2Fhtml%2Dcss%2F201507%2F112357%2Eshtml&urlid=0) /\*IE hack\*/

### ****2. 常用的css选择器命名规则****

### ****1）板块命名法****

头：header

内容：content/container

尾：footer

导航：nav

侧栏：sidebar

栏目：column

页面主体：main

### ****2）位置命名法****

左右中：left right center

### ****3）功能命名法****

登录条：loginbar

标志：[logo](http://cpro.baidu.com/cpro/ui/uijs.php?adclass=0&app_id=0&c=news&cf=1001&ch=0&di=128&fv=18&is_app=0&jk=ac5a3e290aff7d7f&k=logo&k0=logo&kdi0=0&luki=6&mcpm=0&n=10&p=baidu&q=smileking_cpr&rb=0&rs=1&seller_id=1&sid=7f7dff0a293e5aac&ssp2=1&stid=9&t=tpclicked3_hc&td=1682280&tu=u1682280&u=http%3A%2F%2Fwww%2Eth7%2Ecn%2Fweb%2Fhtml%2Dcss%2F201507%2F112357%2Eshtml&urlid=0)

广告：banner

热点：hot

新闻：news

下载：download

子[导航](http://cpro.baidu.com/cpro/ui/uijs.php?adclass=0&app_id=0&c=news&cf=1001&ch=0&di=128&fv=18&is_app=0&jk=ac5a3e290aff7d7f&k=%B5%BC%BA%BD&k0=%B5%BC%BA%BD&kdi0=0&luki=7&mcpm=0&n=10&p=baidu&q=smileking_cpr&rb=0&rs=1&seller_id=1&sid=7f7dff0a293e5aac&ssp2=1&stid=9&t=tpclicked3_hc&td=1682280&tu=u1682280&u=http%3A%2F%2Fwww%2Eth7%2Ecn%2Fweb%2Fhtml%2Dcss%2F201507%2F112357%2Eshtml&urlid=0" \t "_blank)：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：guide

服务：service

注册：register

状态：status

投票：vote

合作伙伴：partner

### ****2.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

### ****（2）页面位置****

左右中：left right center

### ****（3）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](http://cpro.baidu.com/cpro/ui/uijs.php?adclass=0&app_id=0&c=news&cf=1001&ch=0&di=128&fv=18&is_app=0&jk=ac5a3e290aff7d7f&k=%B5%BC%BA%BD&k0=%B5%BC%BA%BD&kdi0=0&luki=7&mcpm=0&n=10&p=baidu&q=smileking_cpr&rb=0&rs=1&seller_id=1&sid=7f7dff0a293e5aac&ssp2=1&stid=9&t=tpclicked3_hc&td=1682280&tu=u1682280&u=http%3A%2F%2Fwww%2Eth7%2Ecn%2Fweb%2Fhtml%2Dcss%2F201507%2F112357%2Eshtml&urlid=0" \t "_blank)：rightsidebar

菜单：menu

子菜单：submenu

标题: title

摘要: summary

### ****（4）功能****

标志：[logo](http://cpro.baidu.com/cpro/ui/uijs.php?adclass=0&app_id=0&c=news&cf=1001&ch=0&di=128&fv=18&is_app=0&jk=ac5a3e290aff7d7f&k=logo&k0=logo&kdi0=0&luki=6&mcpm=0&n=10&p=baidu&q=smileking_cpr&rb=0&rs=1&seller_id=1&sid=7f7dff0a293e5aac&ssp2=1&stid=9&t=tpclicked3_hc&td=1682280&tu=u1682280&u=http%3A%2F%2Fwww%2Eth7%2Ecn%2Fweb%2Fhtml%2Dcss%2F201507%2F112357%2Eshtml&urlid=0)

广告：banner

登陆：login

登录条：loginbar

注册：register

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：guide

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

## HTML编码规范

* 嵌套元素应当缩进一次（即一个tab）。
* 对于属性的定义，确保全部使用双引号，绝不要使用单引号。
* 不要在自闭合（self-closing）元素的尾部添加斜线 -- [HTML5 规范](http://dev.w3.org/html5/spec-author-view/syntax.html#syntax-start-tag)中明确说明这是可选的。
* 不要省略可选的结束标签（closing tag）（例如，</li> 或 </body>）。
* **HTML5 doctype。**为每个 HTML 页面的第一行添加标准模式（standard mode）的声明，这样能够确保在每个浏览器中拥有一致的展现。<!DOCTYPE html>
* **语言属性。**强烈建议为 html 根元素指定 lang 属性，从而为文档设置正确的语言。这将有助于语音合成工具确定其所应该采用的发音，有助于翻译工具确定其翻译时所应遵守的规则等等。<html lang=”zh-CN”>
* **IE兼容模式。**IE 支持通过特定的 <meta> 标签来确定绘制当前页面所应该采用的 IE 版本。除非有强烈的特殊需求，否则最好是设置为 **edge mode**，从而通知 IE 采用其所支持的最新的模式。例

<meta http-equiv="X-UA-Compatible" content="IE=Edge">

* **字符编码。**通过明确声明字符编码，能够确保浏览器快速并容易的判断页面内容的渲染方式。这样做的好处是，可以避免在 HTML 中使用字符实体标记（character entity），从而全部与文档编码一致（一般采用 UTF-8 编码）。

<meta charset="UTF-8">

* **引入css和javacript文件。**根据 HTML5 规范，在引入 CSS 和 JavaScript 文件时一般不需要指定 type 属性，因为text/css 和 text/javascript 分别是它们的默认值。
* **适用为王。**尽量遵循 HTML 标准和语义，但是不要以牺牲实用性为代价。任何时候都要尽量使用最少的标签并保持最小的复杂度。
* **属性顺序。**class 用于标识高度可复用组件，因此应该排在首位。id 用于标识具体组件，应当谨慎使用（例如，页面内的书签），因此排在第二位。
* class
* id, name
* data-\*
* src, for, type, href
* title, alt
* aria-\*, role
* <a class="..." id="..." data-modal="toggle" href="#">

### 编辑器配置

将你的编辑器按照下面的配置进行设置，以避免常见的代码不一致和差异：

* 用两个空格代替制表符（soft-tab 即用空格代表 tab 符）。
* 保存文件时，删除尾部的空白符。
* 设置文件编码为 UTF-8。
* 在文件结尾添加一个空白行。

参照文档并将这些配置信息添加到项目的 .editorconfig 文件中。例如：[Bootstrap 中的 .editorconfig 实例](https://github.com/twbs/bootstrap/blob/master/.editorconfig)。更多信息请参考 [about EditorConfig](http://editorconfig.org/)。

参考：<http://codeguide.bootcss.com/>