#### 一般规范

以下章节列举了一些可应用在 HTML, JavaScript 和 CSS/SCSS 上的通用规则。

1.文件/资源命名

在 web 项目中，所有的文件名应该都遵循同一命名约定。以可读性而言，减号（-）是用来分隔文件名的不二之选。同时它也是常见的 URL 分隔符（i.e. //example.com/blog/my-blog-entry or //s.example.com/images/big-black-background.jpg），所以理所当然的，减号应该也是用来分隔资源名称的好选择。

请确保文件命名总是以字母开头而不是数字。而以特殊字符开头命名的文件，一般都有特殊的含义与用处（比如 compass[1] 中的下划线就是用来标记跳过直接编译的文件用的）。

资源的字母名称必须全为小写，这是因为在某些对大小写字母敏感的操作系统中，当文件通过工具压缩混淆后，或者人为修改过后，大小写不同而导致引用文件不同的错误，很难被发现。

还有一些情况下，需要对文件增加前后缀或特定的扩展名（比如 .min.js, .min.css），抑或一串前缀（比如 3fa89b.main.min.css）。这种情况下，建议使用点分隔符来区分这些在文件名中带有清晰意义的元数据。

#不推荐

html 代码:

MyScript.js

myCamelCaseName.css

i\_love\_underscores.html

1001-scripts.js

my-file-min.css

#推荐

html 代码:

my-script.js

my-camel-case-name.css

i-love-underscores.html

thousand-and-one-scripts.js

my-file.min.css

2.协议

不要指定引入资源所带的具体协议。

当引入图片或其他媒体文件，还有样式和脚本时，URLs 所指向的具体路径，不要指定协议部分（http:, https:），除非这两者协议都不可用。

不指定协议使得 URL 从绝对的获取路径转变为相对的，在请求资源协议无法确定时非常好用，而且还能为文件大小节省几个字节。

#不推荐

html 代码:

<script src="http://cdn.com/foundation.min.js"></script>

#推荐

html 代码:

<script src="//cdn.com/foundation.min.js"></script>

#不推荐

css 代码:

.example {

background: url(http://static.example.com/images/bg.jpg);

}

#推荐

css 代码:

.example {

background: url(//static.example.com/images/bg.jpg);

}

3.文本缩进

一次缩进两个空格。

html 代码:

<ul>

<li>Fantastic</li>

<li>Great</li>

<li>

<a href="#">Test</a>

</li>

</ul>

css 代码:

@media screen and (min-width: 1100px) {

body {

font-size: 100%;

}

}

js 代码:

(function(){

var x = 10;

function y(a, b) {

return {

result: (a + b) \* x

}

}

}());

4.注释

注释是你自己与你的小伙伴们了解代码写法和目的的唯一途径。特别是在写一些看似琐碎的无关紧要的代码时，由于记忆点不深刻，注释就变得尤为重要了。

编写自解释代码只是一个传说，没有任何代码是可以完全自解释的。而代码注释，则是永远也不嫌多。

当你写注释时一定要注意：不要写你的代码都干了些什么，而要写你的代码为什么要这么写，背后的考量是什么。当然也可以加入所思考问题或是解决方案的链接地址。

#不推荐

js 代码:

var offset = 0;

if(includeLabels) {

// Add offset of 20

offset = 20;

}

#推荐

js 代码:

var offset = 0;

if(includeLabels) {

// If the labels are included we need to have a minimum offset of 20 pixels

// We need to set it explicitly because of the following bug: http://somebrowservendor.com/issue-tracker/ISSUE-1

offset = 20;

}

一些注释工具可以帮助你写出更好的注释。JSDoc 或 YUIDoc 就是用来写 JavaScript 注释用的。你甚至可以使用工具来为这些注释生成文档，这也是激励开发者们写注释的一个好方法，因为一旦有了这样方便的生成文档的工具，他们通常会开始花更多时间在注释细节上。

5.代码检查

对于比较宽松自由的编程语言来说，严格遵循编码规范和格式化风格指南就显得极为重要。遵循规范固然很好，但是有自动化流程来确保其执行情况，岂不更佳。Trust is good, control is better.

对于 JavaScript，建议使用 JSLint 或 JSHint。

[1]: Compass 是一个基于 Sass 开源的 CSS 框架，而 Sass 是一个非常流行的 CSS 预编译器。