**[翻译]OOCSS FAQ**

* 原文：<http://wiki.github.com/stubbornella/oocss/faq>（翻译时为Version 28）
* 翻译：[ytzong](http://www.99css.com)

### 在OOCSS中怎么定义“对象”？

对象类似JAVA中的类，保持着OO的特征。

一个CSS对象由4部分组成：

1. 可能是一个或多个DOM节点的HTML
2. 由wrapper节点的class名开始的CSS样式声明
3. 类似于背景图片和显示用的sprites组件以及
4. JavaScript行为，监听或者与对象关联的方法

这可能令人费解，因为每个CSS class不是其自身必要的对象，但可以是一个wrapper class的一个部件。比如：

<div class="mod">

<div class="inner">

<div class="hd">Block Head</div>

<div class="bd">Block Body</div>

<div class="ft">Block Foot</div>

</div>

</div>

对象是一个class为mod的模块。包括4个部件节点（不能独立于模块外，包括2个区块，inner和body，和两个可选择的区块，head和foot）

### OOCSS如何提升性能？

OOCSS具有双倍的性能优势：

1. 高度重用的CSS代码，只需要很少的CSS代码，意味着：
   * 更小的文件，从而更快的传输
   * CSS代码在站点页面中调用的比重增大则有希望被复用或被浏览器缓存
2. 就浏览器而言更少的重绘和布局计算
   * 单个页面，CSS规则复用的越多，渲染引擎花在“computed values”的计算时间越少
   * 手动增加的”extending”类，重写更少的规则，这再一次意味着引擎做很少去应用规则

### 要用ID来对内容写样式吗？

当你在同一页面（或者同一站点同时缓存良好）复用一个对象时，这是性能的“免费赠品”。用ID来写样式在同一页面中只能使用一次。@cgriego (twitter)拿它与singletons比较过，我认为非常精确。可能有些情况下你要用ID定义样式，比如非常特殊的header menus，此时你可以在用ID来沙箱（译注：动名词）特殊元素并确保此处的代码不会影响站点的其它地方。选择ID而非class前要三思，随着站点的发展，真的很难预料其他人会怎么处理依据你的CSS所构建的HTML。如有选择的余地，尽可能的考虑扩展性。

我正在考虑移除模板head, body, foot中的ID。某些人或许有多个主区域。站点的多个header 和 footer更难以猜测，但我敢打赌肯定有设计师会这样想，所以ID很可能会消失（不太顺，看原文：Someone could have multiple main content areas. Multiple site headers and footers are more difficult to imagine, but I bet there is a designer who can dream up something like that, so the IDs are very likely to disappear.）。

另一方面，ID hooks 对链接非常有用(译注：做锚点使用)。放在HTML中，不过别用它们来写样式。

### 设计师可以写OOCSS吗？

是的，设计师出于本能理解对象，比多数人当前书写CSS的方式要形象 — layers of exceptions (想一下，一个老太太吞了一只苍蝇)。事实上，他们爱上OOCSS的原因有两个：

1. 这使他们能快速创建复杂高点击量的站点。他们不需纠结于不理解的结构除非有足够的能力并充足的了解语法
2. 学CSS时，他们不需创建丑陋的 “hello world!” 站点。设计师在非常在意的是他们的工作看起来很漂亮。如果必需做一些丑陋的东东，即便是学习之由，他们很快就会有挫败感并灰常的郁闷。OO-CSS使得他们的工作在每个学习阶段都看起来很漂亮

设计师是**聪明**D。我们要给他们信任。他们会讲一种不同的，非工程师的语言，但是极客的语言经常以一种丑陋的方式来拒绝人。我们能做的更好。

### 我是个前端架构师，如何向团队传授OOCSS？

作为架构师，你应该写结构对象；圆角如何创建，为角或其他特性放置表象元素，并处理浏览器差异。新手为这些模块写皮肤（borders, colors, background images, 等等）。

我用OO-CSS方式创建了大批站点（千级的页面，百万级的访问者）。正确的完成后，很好扩展，这意味着新手将处理的个别元件可预见性很强。代码审阅很容易，因为有可接受的方法明确的规则来扩展对象。这种回馈使新开发者快速生产。

我在FullSIX（一个法国的网络营销机构）管理一个前端开发团队，是我工作过的最有才能的。某些时候我们的成功意味着我们将会有更多难以把控的工作。雇佣前端专家非常困难（没有这种学校！），所以我开始[对一些对写代码有兴趣的设计师（很少或没有经验）推行一个内部实习项目](http://www.slideshare.net/stubbornella/object-oriented-css/57)，一个月就可以作为团队的初级成员工作。

* 第一周：学习语义并根据现有的CSS创建HTML。学习创建网页：不需要更多的CSS，HTML语法，多个class，验证，语义，介绍代码审阅等
* 第二周：创建简单的内容对象（标题，列表等），持续一周。学习CSS语法，怎么扩展对象，颜色，字体百分比，等等
* 第三周：创建区块的皮肤。边框，颜色，背景图片，基本的布局，sprites。让他们同一个回答过n个问题的资深开发者一起工作，使他们少走弯路，他也应是很好的代码审阅者。
* 第四周：他们已经是团队的皮肤制作成员了。

他们的代码在一个客户的网站上，同资深开发者写的一样好，或许更好因为他们还未学到一些坏习惯:)

### 入门：如何使用这些文件？

3个文件，libraries.css (reset 及 fonts 取自 yui), grids.css 及 template.css 已完成，其它的还非常不稳定。

1. 打开template.html并存为新文件
2. 通过扩展相应的对象来改写列的数量及宽度。站点中只需一个模板，即使你有不同列的页面，因为列也是对象。可以把它们当作任意的区域，可能会有0 ~ n个左列。查阅模板文档可了解更多
3. 用栅格来分割内容区域为小的区块。查阅栅格文档了解更多
4. 添加内容。提示：这也应OO

### 如何部署在站点上？

注意CSS文件在不断改进中，我会依据接到的反馈进行改变。

我把CSS文件分为了模块，比如栅格和模板。在使用时移除不必要的注释并减少HTTP请求，否则站点会超级慢。这意味着要合并CSS文件为一个稍大的文件。我通过嵌套的注释来组织CSS。最后，作为上线/部署的一部分，用CSS压缩器来移除注释.

### 可以修改文件，或者用我自己的样式重写吗？

我不会修改grids, template, 或者 libraries。大量测试表明这些已恰到好处。如果要自定义，考虑下面的[扩展基本对象](http://www.99css.com/?p=65#extend)。

### 粉红不是我要的颜色！怎么处理content.css？

获取你会想要修改content.css。去吧，改颜色，字体大小，大小写。只需注意这个文件在快速发展，同时我还没有任何文档来说明如何正确的处理。我会这么做，我保证。

### 我需要不只6种标题（h1~h6），如何增加？

如果需要不只6中标题样式，通过添加一个新class来扩展标题对象。

.category{font-size:108%; font-weight:normal; font-style: normal; text-transform:uppercase; color: #333;}

**不要**这样做：

#mySaleModule h2, #mySaleModule .h2{font-size:108%; font-weight:normal; font-style: normal; text-transform:uppercase; color: #333;}

## 如何扩展对象？

如果要扩展对象，比如一个160px的左列，而非默认值，你可以再列上增加额外的class。

<div class="leftCol gMail"> ... </div>

如果默认值和扩展的列宽或者页面不适合你的站点，你可以扩展列来实现自定义的宽度。

#### 列

myColumn 扩展列对象来实现自定义列宽。

.myColumn{width:400px;}

HTML

<div class="leftCol myColumn"> ... </div>

不要通过重写我的class来实现，而应扩展此框架提供的对象。我提供了列，标题及其他对象。你可以通过增加另外的class（只指定与我的基本对象的不同点）来扩展这些对象。相对而言此处混合比较好。

**不要**这样做（因为更新我的框架时会有些麻烦）：

.leftCol{… 此处自定义CSS …}

### 没有用到的样式。我的站点没有160px的gmail式的列，可以移除吗？

当然。移除对象或扩展对象非常合理。只需注意在站点发展时，很难预料到其他人用你的CSS创建的什么样的HTML。过早优化很危险。

### 为什么有一个单独的模板？

在OOCSS中，一个重要的目标是所有的页面创建自一个模板。这简化了CMS开发，因为有一个单独的起始点所有页面可以制作为其他的页面。CMS的用户不会落入已创建的页面不能变种为不同的页面类型的陷阱。OO模板的另一个目标是每一个section（列，header等）控制自己。实际上，这意味着如果要在模板上增加一个左列，只需在HTML中增加此列。你从来不想这样写CSS吧，为了使子元素满足表现而DOM树需要很多改变。对CMS开发来说DOM循环相当的昂贵。

### 这有语义吗？我要终止像 .formYellow 或 tinyBlueH2 的class命名吗？

OOCSS可以写得有语义也可无语义，取决于你创建模块时用 errorMod 而非 bigRedModule。我选择class名优一些原则（排名不分先后）：

* 短 – 每一字节计算起来，所以尽可能保持class短
* 清晰 – 可预料的行为/样式要显而易见
* 语义化 – 对象是什么高过看起来像什么。如何用在站点中？
* 通用 – 名称应该适用于多数站点。过于特殊的名称减少了适用场合或导致语义化的class以非语义化的方式使用
* 屏幕 – 移动阅读器或打印样式或许有不同的视图，会重写默认的屏幕视图，所以当有冲突时class为屏幕而特殊定义（Different views might be provided by mobile or print stylesheets, however they override the  
  default screen view, so the classes chosen are screen specific when there was a conflict）。这简化了开发。

### 其它的框架如何？这只能同YUI一起使用吗？

在大量框架的生态系统中，YUI是专业性及可扩展性的一例。我同他们进行了对比，因为我不断受到他们代码和文档的影响。OOCSS不是一个真正意义上的框架，尽管我创建了很多例子，而是一种书写可扩展，健壮，可维护性强的CSS的一种方式。也许最好的比喻是一个新的语言。最终，它是未知的JavaScript库（it is JavaScript library agnostic），我希望贡献代码回YUI及其他框架。

### CSS框架太超过！我需要从头开始编写所有代码吗?

每需要一个随机数字都要写一个数字class吗？

CSS is hard,[not because it is broken](http://www.stubbornella.org/content/2009/02/12/css-doesn’t-suck-you’re-just-doing-it-wrong/) , but because it is a legitimate technology requiring expertise to architect correctly. 为每个站点重复发明轮子非常的愚蠢。

### 源文件中右列在主列之前，会影响可访问性，SEO吗？

我早期工作过的站点有更标准的模板结构，body上有一个极好的class，依靠这个模板显示或隐藏左右列。自定义CMS的用户要创建一个3列布局的页面时非常沮丧，发现需要两列，找到它们不得不一个个从头开始，因为有多个模板/起始点。你或许注意到主列是个流体列，在左右两列渲染后自适应剩余的空间。

我更喜欢标签排序胜过视觉排序（因为如果标签顺序改变会变得非常怪异），但是我也只想要一个模板。在web开发中经常遇到的是，两个重要的目标发生了冲突，我做判断如何解决。这种情况下标签排序满足视觉顺序除了右列。在当前的代码中，只需在HTML增加一个左或右列，没有别处昂贵的DOM变化。

屏幕阅读器用户有两个选择：

1. 跳过链接
2. 导航链接经常为一个链接列表或嵌套的链接列表形式。这非常有趣，因为这允许屏幕阅读者通过屏幕阅读器的特殊操作设置跳过整个列表。

我的意见是对于跳过菜单这两种方式足够了。

每个人似乎都有一个回复观点：SEO，它们都不同，甚至相反。:)基于这个思潮，我倾向于：尤其对含有一定合理数量链接的导航菜单，不用太过在意。曾几何时，我看到过导航链接被索引在搜索结果的内容部分，不过是在几年前了。搜索爬虫非常智能，我已经准备好想当然的认为如果我以语义化，干净的方式标记内容，并非填充一坨垃圾链接，爬虫能区分的出来。

把导航菜单标记为列表，允许屏幕阅读器用户跳过，并提供“跳至内容”链接，对可访问性提供了双倍的备用措施。

### 为什么用了\_ hack？我能把这些代码放在单独的文件中吗？或者添加IE专有的class？

很显然，首先考虑的是尽可能少和长远。

1. 添加一个单独的样式表奖增加一个额外的HTTP请求，增加整体文件的大小，这早已是浏览器性能的对立点
2. 我喜欢把一个对象的代码放在一个地方。我认为这有助于减少hack的数量，尤其是当项目随时间而发展
3. IE6-的开发工具非常原始，这使得hack和普通代码放在一起更加重要。我想能尽快找到一个可以使用属性的IE bug。同样，我认为这有助于减少hack
4. 规则表明浏览器理解不了的属性会被忽略。对IE6-使用\_早已众所周知，可以合理的预料好的浏览器将会忽略这个属性
5. 使用条件注释意味着每个HTML页面必须包含一个IE专用样式。某一天（我不能等了！）当IE6的市场份额下降至像IE5那样低时，我将去除这些代码，但最后我要做的是触及百级或千级的HTML页面。我宁愿只有依赖CSS hack的CSS，而不是把它放在HTML中。不幸的是，恕我直言，IE6兼容性的末日比我们期望的要更加长远，因为IE中的quirksmode会回落到IE5.5的模式

我认为我的选择有助于减少hack的总体数量，提升性能，并只有忽略不计的风险。话说回来，如果觉得代码中的\_令你作呕，你完全可以移至单独的文件。

### 为了添加表象效果比如边框而使用空 <b> 标签容器对象，会给屏幕阅读器用户带来问题吗？

不会，幸运的是屏幕阅读器会忽略空的b标签。使用这个表象标签（b是加粗）来实现表象具有优势。这个标签可以通过服务器端脚本包含，以至于有一天所有的CSS圆角和阴影都支持了，可以关闭脚本并拥有漂亮，干净，语义化的HTML。

### OOCSS声称避免位置依赖的样式。是否意味着我不能使用类似 .myModule .title 的派生选择器？

不使用派生选择器没什么阻碍，只是把它们放在更高的DOM树而已。避免在body或页面中最外层的div放置wide-net class或id，然后对对象产生的变量写大量的样式。这不能复用，同时会使页面渲染变慢。相反，通过直接在对象上添加一个class来增加一个多余的“local”变量（并对其子元素写派生样式）。

一个好的经验是一个容器主体（body of a container）内的任意元素显然是一个单独的对象。

这无可厚非，因为UL显然是一个单独的对象：

#sidebar ul { ... }

因此，carousel显然不是body对象的一部分：

body.browseProducts .carousel

这是适当的层叠应用，因为子节点真正的是较大的父对象的一部分。b（角）和inner显然隶属于moudle，它们不能独立存在：

.myModule { ... }

.myModule b { ... }

.myModule .inner { ... }

### 如何贡献？

最好的方法是涉足其中，开始使用代码（libraries, grids, fonts）并[提交bug 报告及功能需求](http://stubbornella.lighthouseapp.com/projects/26663-object-oriented-css/tickets)。 我建了一个[OOCSS google group](http://groups.google.com/group/object-oriented-css) 来进行超过140个twitter字符的讨论。

### 译注：

OOCSS的官方站点为 [http://oocss.org](http://oocss.org/)，有一些例子及下载等。