

CSS规范 - 分类方法

CSS文件的分类和引用顺序

通常，一个项目我们只引用一个CSS，但是对于较大的项目，我们需要把CSS文件进行分类。

我们按照CSS的性质和用途，将CSS文件分成“公共型样式”、“特殊型样式”、“皮肤型样式”，并以此顺序引用（按需求决定是否添加版本号）。

1. 公共型样式：包括了以下几个部分：“标签的重置和设置默认值”、“统一调用背景图和清除浮动或其他需统一处理的长样式”、“网站通用布局”、“通用模块和其扩展”、“元件和其扩展”、“功能类样式”、“皮肤类样式”。
2. 特殊型样式：当某个栏目或页面的样式与网站整体差异较大或者维护率较高时，可以独立引用一个样式：“特殊的布局、模块和元件及扩展”、“特殊的功能、颜色和背景”，也可以是某个大型控件或模块的独立样式。
3. 皮肤型样式：如果产品需要换肤功能，那么我们需要将颜色、背景等抽离出来放在这里。

```
<link href="assets/css/global.css" rel="stylesheet" type="text/css"/>
<link href="assets/css/index.css" rel="stylesheet" type="text/css"/>
<link href="assets/css/skin.css" rel="stylesheet" type="text/css"/>
```

CSS内部的分类及其顺序

1. 重置 (reset) 和默认 (base) (tags)：消除默认样式和浏览器差异，并设置部分标签的初始样式，以减少后面的重复劳动！你可以根据你的网站需求设置！
2. 统一处理：建议在这个位置统一调用背景图（这里指多个布局或模块或元件共用的图）和清除浮动（这里指通用性较高的布局、模块、元件内的清除）等统一设置处理的样式！
3. 布局 (grid) (.g-)：将页面分割为几个大块，通常有头部、主体、主栏、侧栏、尾部等！
4. 模块 (module) (.m-)：通常是一个语义化的可以重复使用的较大的整体！比如导航、登录、注册、各种列表、评论、搜索等！
5. 元件 (unit) (.u-)：通常是一个不可再分的较为小巧的个体，通常被重复用于各种模块中！比如按钮、输入框、loading、图标等！

6. 功能 (function) (.f-)：为方便一些常用样式的使用，我们将这些使用率较高的样式剥离出来，按需使用，通常这些选择器具有固定样式表现，比如清除浮动等！不可滥用！
7. 皮肤 (skin) (.s-)：如果你需要把皮肤型的样式抽离出来，通常为文字色、背景色（图）、边框色等，非换肤型网站通常只提取文字色！非换肤型网站不可滥用此类！
8. 状态 (.z-)：为状态类样式加入前缀，统一标识，方便识别，她只能组合使用或作为后代出现（.u-ipt.z-dis{}，.m-list li.z-sel{}），具体详见命名规则的扩展相关项。

功能类和皮肤类样式为表现化的样式，请不要滥用！以上顺序可以按需求适当调整。

以上分类的命名方法详见**命名规则**

```
/* 重置 */
div,p,ul,ol,li{margin:0;padding:0;}
/* 默认 */
strong,em{font-style:normal;font-weight:bold;}
/* 统一调用背景图 */
.m-logo a,.m-nav a,.m-nav em{background:url(images/sprite.png) no-repeat
9999px 9999px;}
/* 统一清除浮动 */
.g-bdc:after,.m-dimg ul:after,.u-
tab:after{display:block;visibility:hidden;clear:both;height:0;overflow:hidden;con

.g-bdc,.m-dimg ul,.u-tab{zoom:1;}
/* 布局 */
.g-sd{float:left;width:300px;}
/* 模块 */
.m-logo{width:200px;height:50px;}
/* 元件 */
.u-btn{height:20px;border:1px solid #333;}
/* 功能 */
.f-tac{text-align:center;}
/* 皮肤 */
.s-fc,a.s-fc:hover{color:#fff;}
```