

经典水果&水果拼盘——css的快乐水

§1、概述

css，学名层级样式表，顾名思义，样式层层叠加的表。严格来说，css不属于编程的范畴，只是层级样式的排列组合。所以，我们能操纵的，就是它的排列组合方式。

§2、组合方式：经典水果vs水果拼盘

§2.1、水果拼盘

我们常用的方式，是将样式组合好在一个类名中，dom节点引用单一类名，就像把水果切成拼盘，直接食用。这种水果拼盘的组合方式，优点自不必说，那么问题在哪里呢？

- 1、命名头疼，俩果盘水果种类差不多，更头疼！
- 2、保质期短，新鲜拼出来的，还能卖个好价，还能将名字和果盘和买家对上号，时间一长三脸懵逼，无人问津，弃之可惜！
- 3、所以，最恐怖的是维护，水果拼盘不知道被卖给多少家，若是已经卖给了许多家，那改了拼盘里的水果，其他买家不买账，岂不赔大发了？

§2.2、经典水果 —— 一年四季都畅销的水果

若完全经典水果，即完全引用经典样式（原子样式或分子样式），在dom节点处多类名组合，即不单独切水果了，而是在吃的时候直接处理，这样做，有什么好处，又有什么弊端呢？

- 1、好处：
 - 1) 保质期为永久。
 - 2) 苹果就是苹果，西瓜就是西瓜，很底层，用途很单一，命名自然也很简单，再不用为起什么名字愁白了头，效率起码提高50%。
 - 3) 解耦合！试想一下，一个个性化div，只有一个地方用到，通过经典样式类在dom节点就地组合去实现（class="xxx xxx xxx"），将来若删除这个div节点，或者改变另一种风格了，那么只需要改动该div自身即可，丝毫不影响别的节点，更不需要删除css文件中冗余的水果拼盘代码，做到解耦合。
- 2、弊端：

试想一下，若这个div，写了一长串类名，然后用于很多地方，突遇改版，设计把这个div的风格改了，那我们是不是每个div都要改一遍，累不累？

§3、取长补短

由此已看出两者是欢喜冤家，天生的一对，是以两者阴阳相融，岂不妙哉！两者该怎么取长补短，做到既能解耦合，又方便维护呢，大家应该都有想法了。

- 1、先回顾一下可以单独使用的场景
 - 1) 水果拼盘：用于重复使用的控件样式。
 - 2) 经典水果：用于个性化定制的风格。
- 2、更多的场景需要两者结合，如：
 - 1) 按钮等控件：位置采用经典水果（margin），核心采用水果拼盘。
 - 2) 作为容器的节点：布局用经典水果（flex、padding），边框阴影等共性的样式用水果拼盘（ps：也可以说

是该项目自定义的经典样式）。

§4、具体实践方案

可参考项目：<http://192.168.0.116/petra/webapp-asclepius>

一个项目的样式目录，取名**sass**，只需包含四个文件：**global.scss**、**variable.scss**、**main.scss**、以及**sass-index.scss**。目录如下：

```
├-sass
  ├──global.scss
  ├──main.scss
  ├──sass-index.scss
  └-variable.scss
```

§4.1、global.scss。

1、概述：

全局样式，任意项目皆可引用。采用中划线（个别驼峰）直接根据属性和值命名，如.margin10X（上下外间距10px，左右未知）。该文件涉及盒模型padding、margin，布局flex，定位position等等一些基础的，使用率最高的原子、分子样式，具体有哪些，目前没有文档，因为命名简单，基本看或用上几遍就熟能生巧了。此文件拓展性良好，大家经常使用的经典样式，只要保持队形，就能加入进来，后期经典样式多了还可以写个文档，以及如何增加灵活性，这个需要大家一起去努力完善。（ps：这个文件由管理员维护，不要直接去改动，而是告知管理员，通过评审之后更新上去。之后可以通过将文件放在cdn资源管理器中去规避这个问题）

2、**global.scss**命名的几个原则：

- 1）中划线
- 2）属性值辨识度高的，直接采用属性值命名（如.bold），根据值无法区分的加上属性名(如.float-right)
- 3）数字紧跟属性，单位有两个：配置单位、百分比，根据属性自行区分，很简单（如.border1、.width100）

§4.2、variable.scss

存放该项目的变量，注意，是供该项目用的，色值等变量都可以放在这里。理想状态是：如果有改皮肤的需求，那么只需要改动这个文件即可。

§4.3、main.scss

存放该项目跨页面使用的水果拼盘样式。即一些基础的控件核心，比如按钮、图标这样一些分布在各处的统一风格的样式。

那么同一个页面的水果拼盘呢，当然可以在单页面中，用"<style lang='scss' scoped>"就好了。注意，加scoped哦。

ps：看上去是组件的模块（如搜索框），或重复的模块（如相同的列表），第一时间写成组件单页面，便于后面的开发和维护。

§4.4、sass-index.scss

用于import其他三个scss文件。

§5、赘述一下，这样统一样式编码风格之后，有什么好处。

1、跨项目无缝嫁接！

想象一下，一个组件页面，在A项目里不断打磨完善好了，当B项目要用的时候，需要怎么做，一般情况下，除了copy该组件页面，多多少少还牵扯一些全局样式等vue页面之外的种种琐碎的东西，可是用了以上方案，就只需要一键copy组件页面，再无后顾之忧，爽不爽！

2、耦合度低，大大降低维护成本！毕竟一个项目80%在于维护。

3、解放为css命名愁白了头的烦恼，从此可以快乐无忧地，飞一般的写页面！

4、可读性强，团队开发，零障碍！

还有很多，在开发中慢慢去体会吧~

版权归卓建科技掌上医院-自运营平台前端团队所有，翻版必究！