CSS书写规范

1、编码声明

如HTML文件一般，css文件也必须声明该文件使用什么编码，并且使用UTF-8。

同时于文件最顶部写上charset指令（@charset “utf-8”;）。这样我们可以放心地在该文件中写上中文注释。

2、语法风格

* 在只有单个属性时，使用单行编写样式，如 .xxx { border: solid 1px black; }。
* 在多个属性时，可以每个属性占一行，如果属性条数不是太多，全部占一行即可。
* 在选择器与左花括号之间加空格。
* 在花括号与属性之间加空格。
* 在属性名称与属性值之间的:后面加空格。
* 每条属性都必须以分号结尾。
* 属性值如果需要用逗号，除非像rgb()这种加了空格会出错的情况，一率在逗号右边加空格。
* 尽量用简洁的选择器限定方式。
* 尽量避免使用hack方式实现浏览器兼容。

3、选择器

时时考虑移植性问题，请使用class来作选择器，避免使用标签名，杜绝使用id。

让你的class变得更加语义化，允许缩写，但保证每个人都看得懂它。

class避免使用下划线\_来分隔单词，而是使用横杠-来分隔，这是因为用于IE6的css hack写法是使用下划来\_的，所以直接避免而使用横杠-，这也可以与js的变量区分开来。

4、css属性前缀

某些css3属性或者规则已经在各个浏览器上实现了，但它尚未成为W3C标准的一部份时，我们需要为它添加前缀。一般来说需要以下几个前缀，以实现圆角边框的border-radius为例：

.div1 {

-webkit-border-radius: 3px;

-o-border-radius: 3px;

-moz-border-radius: 3px;

-ms-border-radius: 3px;

border-radius: 3px;

}

当该属性或者规则已经在流行的各大浏览器普遍实现支持标准写法时，可以去除前缀。

有许多种方法可以简化前缀写法，比如使用sass的mixin、一些代码编码器的功能模块都可以让我们把这个过程自动化，请自行上网查询。

注意：请把标准写法放到最后。

5、css hack

由于不同浏览器渲染效果的差距导致同样的样式效果需要写不同的css代码或者微调属性值来实现，因此不可避免地需要找到方法让它们区分开来。

下方列了足够区分IE6、IE7、IE8、IE9、IE10+、标准浏览器（chrome、safari、firefox以及opera）的hack写法，但请注意：尽量使用规范、非hack的写法来实现兼容。

.text {

color: red; /\* 所有浏览器都支持 \*/

color: black\0; /\* IE8-11，被后续样式影响只有IE8生效 \*/

color: orange\9\0; /\* IE9-10，最终只有IE9生效 \*/

\*color: blue; /\* IE6、IE7，最终只有IE7生效 \*/

\_color: green; /\* IE6 \*/

}

/\* IE10+，由于IE10以上版本已经基本上就是标准浏览器了，极少需要hack，不需要再做区分 \*/

@media screen and (-ms-high-contrast: active), (-ms-high-contrast: none) {

.text { color: yellow; }

}

另外，color:red\9; 可以被所有IE已知版本识别，但非IE浏览器不识别。

请特别注意一下，这些hack写法由于有交叉影响，因此在书写顺序上要特别注意。

6、文件粒度管理

一般来说一个页面不要加载超过2个css文件，禁止在css文件中使用@import指令加载额外的样式文件。如果由于项目需要必须对css文件做更为细致的划分，存放到大量css文件中，那么请使用sass、less等css预处理器来实现css文件的管理。

7、代码注释

为了让你的小伙伴们能够更方便地维护你写出来的代码，请为你的css代码加上足够的注释。

8、临界值处理

请注意，HTML的动态内容部份长度有可能为空，也有可能会长到你设定的宽度高度放不下最终导致容器撑开，破坏了页面的美观。

在css中可以使用一些小技巧来截取溢出部份的文字，并加上省略号，方法是设置width、height、overflow:hidden;text-overflow:ellipsis;

多行的处理方式是：overflow : hidden; text-overflow: ellipsis; display: -webkit-box; -webkit-line-clamp: 2; -webkit-box-orient: vertical;

需要处理的情况不止文字长度溢出，在书写css一定要考虑到这各种状态。

9、兼容性处理

请关注你的项目所需要兼容的浏览器环境，一般来说一个前端应该关注以下浏览器：

* Internet Explorer 6+（Trident内核）
* Chrome（Blink内核）
* Firefox（Gecko内核）
* Safari（webkit内核）

除非确认未来项目肯定不需要兼容旧版本浏览器，否则尽可能兼容旧版浏览器。

如果你不知道某个属性在某个浏览器是否可用，请阅读参考资料：http://caniuse.com