

学习心得

CSS代码规范

文件名规范：

小写字母加中横线的方式。

属性书写顺序：

重要的属性写在前面，方便调试。

不要使用样式特点命名：

当样式要修改时，该类名就不具有可读性了。类的命名应当使用它所表示的逻辑意义。

不要使用hack，

//或者_开头的CSS属性会被忽略浏览器或者把.css文件改成.less或者.scss文件，这样就可以用//注释了。

选择器的性能：

不要写超过3个，长的选择器的性能比较差。一般只要写两三个重要的选择器，不用每个容器都写进去，重要的目标元素套上class或者id。最后一个选择器的标签的少用，如果写.container div{}的话，页面上所有的div第一次都匹配中，因为它是从右往左匹配的，合适才使用，最起码不能滥用。

避免选择器误选，

不要写*全局匹配选择器，不管从性能还是影响范围来说都太大了。

不要滥用了first-child、:nth-of-type选择器，使用这种选择器的后果是扩展性不好，只要html改了，就会导致样式不管用了，或者影响到了其它无关元素。不能滥用，合适的时候才使用，而不是仅仅为了少写类名。

减少覆盖。

例如实现

```
.house{  
  margin-top: 20px;  
}  
.house:first-child{  
  margin-top: 0;  
}
```

可以改成

```
.house + .house{  
  margin-top: 20px;  
}
```

只有前面有.house的.house才能命中这个选择器，由于第一个.house前面没有，所以命不中。

使用CSS3的选择器完成一些高级的功能

少用!important

过不了多久又要再写一个优先级更高的覆盖掉，要覆盖还是先通过增加添加选择器权重的方式。

多写注释

1. 文件顶部的注释
2. 模块的注释
3. 简单注释
4. TODO的注释

表示这些代码还有待完善，或者有些缺陷需要以后修复。而这种TODO的注释一般编辑器会把TODO高亮。

排版规范：

1. 不管是JS/CSS，缩进都调成4个空格。
2. 多个选择器共用一个样式集，每个选择器要各占一行
3. 每个属性名字冒号后面要带个空格，~、>、+选择器的前后也要带一个空格。

属性值规范

1. 如果值是0，通常都不用带单位。时间有关的时间单位都要带上秒s
2. 色值用十六进制，少用rgb。因为使用rgb是一个函数，它还要计算一下转换。如果是带有透明度的才用rgba。如果色值的六个数字一样，那么写3位就好。
3. 注意border下none和0的区别。
4. 如下两个意思一样：

```
border: 0;
border-width: 0;
```

而下面这两个一样：

```
border: none;
border-style: none;
```

font-family的设置

使用系统字体的对应的font-family名称,把要用到该字体的标签选择器写在一起。到时候整体替换。

不要设置太大的z-index

页面的业务逻辑的z-index应该保持在个位数就好了。

合并属性

为了提高性能，属性要合并。虽然Chrome每个属性都是单独的，就算你合在一起，它也会帮你拆出来，还是推荐写成合的，因为它可以让代码看起来更简洁。

注意float/absolute/fixed定位会强制设置成block

清除浮动

清除浮动有多种方法，一般用clearfix大法，虽然这个方法有缺陷，但是它比较简单且能够适用绝大多数的场景

```
.clearfix:after{
  content: "";
  display: table;
  clear: both;
}
```

引号的使用

1. font-family

一般来说font-family不需要添加引号，即使字体名称带有空格也没关系，但是有一种情况是一定要加上引号的，就是字体名称刚好是关键词，如下字体都需要加关键词：

font-family: "inherit", "serif", "sans-serif", "monospace", "fantasy", and "cursive"

2. background的url

是有一种一定要加引号，那就是url里面带有特殊字符没有转义。图片的名字带有空格，后端给的url没有对特殊字符做处理，就会有问题，所以当url是可变的时候，最好还是带上引号

3. 单引号还是双引号

这两个都是合法的，只是统一一下比较好。比较普遍的推荐是html使用双引号，css使用单引号。

CSS动画规范

1. 不要使用all属性做动画

在有一些浏览器上面可能会有一些问题，哪个属性做动画就写哪个，如果有多个就用隔开。

2. 使用transform替代position做动画

能用transform做动画的，就不会使用left/top/margin等，因为transform不会造成重绘，性能要比position那些高很多。

3. 偏向于使用CSS动画替代JS动画

不要断词

不能使用word-break: break-all把一个单词拆成两行

不要设置图标字体font-family

不要手动一个个去设置font-family了

设置常见样式reset

图片压缩

不管是UI直接给的图片还是自己从UI图里切出来的图片，都需要把图片压缩。使用tinypng，它可以在保持图片质量减少较低的情况下，把图片压得很厉害。色彩比较丰富的图片要使用jpg格式，不能使用png格式，png会大得多，如果是logo那种矢量图片，直接使用svg格式即可。一般来说要把图片控制在300k以内。

正确使用background和img

如果是头图等直接展示的图片还是要img标签，，如果是做为背景图就使用background，因为使用img可以写个alt属性增强SEO，而背景图那种本身不需要SEO。

响应式的规范

响应式开发最不好不要杂合使用rem，文字字号要么全部使用rem，要么不要用，也不要使用transform: scale去缩小，因为被缩小的字号看起来会奇怪。响应式的原则一般是保持中间或者两边间距不变，然后缩小主体内容的宽度。

适当使用:before/:after

:before和:after可以用来画页面的一些视觉上的辅助性元素，如三角形、短的分隔线、短竖线等，可以减少页面上没有用的标签。

少用absolute定位

absolute定位的元素渲染性能会比较高，为absolute定位的可扩展性很差，你把每个元素的位置都定死了就变不了了，可以多用float，虽然float的性能相对较差，但是不管是实用性还是兼容性都是挺好的。

少用inline-block布局

图片的居中和宽高设定

分图片是宽比长大。所以需要居中裁剪展示

移动端提高可点区域范围

移动端的的一些图标如X，可能会设计得比较小，所以点起来会不太好点，因此要提高可点区域范围，可通过增加padding

不要设置input的line-height

设置了line-height为一个很高的值，这样会导致Safari浏览器的输入光标|变得巨大，所以如果你要居中的话，使用padding。

移动端弹框要禁止body滑动

Safari切换输入框的时候会页面会弹闪得很厉害，因为你在切的时候它会先把键盘收起来，然后再弹出来。，但如果把body禁止滑动了就不会有这个问题。两个解决办法，第一种是把body fixed住，第二种设置body overflow: hidden

行内元素可以直接设置margin-left/margin-right

拖拽的实现

今日将昨天的一个div的拖拽，变成将一个小div拖至大的div中。当小的div完全进入大div之后，就不再允许超出大div的边框。

难点是，对于逻辑上的处理。

1. 要注意offsetLeft、pageX等的熟练使用。
2. 明白不同时刻要判断的距离是什么。
3. 注意标志位的使用，注意判断小div左边框是否进入大div，此处必须要判断是否是第一次进入，如果是，小div的边宽在大div的外面也不做处理，当小div进入到大div中后，在超出大div的范围才做

处理。

4. 注意四个边框都要判断

明日计划

1. 找了github上三个基础的项目，下载，借鉴别人的项目，对项目代码进行规范化的书写。主要看笔记应用、番茄钟、产品登陆页面这三个项目。
2. 学习js代码规范
3. 学习浏览器渲染机制
4. 学习js高级编程
5. 有时间的话写一题leetcode