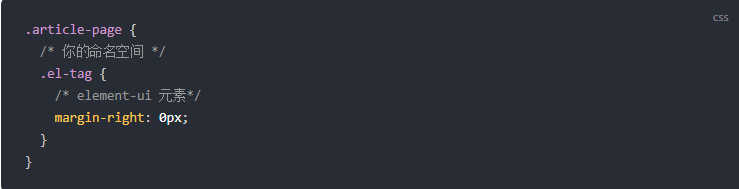
### 1.CSS Modules

#在样式开发过程中，有两个问题比较突出：

1. 全局污染 —— CSS 文件中的选择器是全局生效的，不同文件中的同名选择器，根据 build 后生成文件中的先后顺序，后面的样式会将前面的覆盖；
2. 选择器复杂 —— 为了避免上面的问题，我们在编写样式的时候不得不小心翼翼，类名里会带上限制范围的标示，变得越来越长，多人开发时还很容易导致命名风格混乱，一个元素上使用的选择器个数也可能越来越多，最终导致难以维护。
3. 新建页面时，不同级别的页面不要放同一级
4. 修改element样式时，尽量不要影响全局，如果没有办法规避，就到assets里面新建element组件相对应的scss文件，方便以后维护。

### 2.自定义 element-ui 样式

### 现在我们来说说怎么覆盖 element-ui 样式。由于 element-ui 的样式我们是在全局引入的，所以你想在某个页面里面覆盖它的样式就不能加 scoped，但你又想只覆盖这个页面的 element 样式，你就可在它的父级加一个 class，用命名空间来解决问题。

1. 

### 3.父组件改变子组件样式 深度选择器

1. 父组件改变子组件样式 深度选择器
2. 当你子组件使用了 scoped 但在父组件又想修改子组件的样式可以 通过 >>> 来实现：

<style scoped>

.a >>> .b { /\* ... \*/ }

</style>

1. 将会编译成

.a[data-v-f3f3eg9] .b {

/\* ... \*/}

1. 如果你使用了一些预处理的东西，如 sass 你可以通过 /deep/ 来代替 >>> 实现想要的效果。
2. 所以你想覆盖某个特定页面 element 的 button 的样式，你可以这样做：

.xxx-container >>> .el-button{

xxxx}

### 页面维护

1. bug优先维护自己做的，因为自己写的想对容易修改
2. 新建全局变量时要做注释
3. Css,js，也尽量做些注释，方便其他同学维护，修改