

**码出水平**

**梦嘉集团Web前端开发手册 内部测试版（1.0.0）**

1. **一般规范**
2. 外部文件名规范

正例：

1、my-script.js

2、my-camel-case-name.css

3、i-love-underscores.html

4、thousand-and-one-scripts.js

5、my-file.min.css

反例：

1、MyScript.js

2、myCamelCaseName.css

3、i\_love\_underscores.html

4、1001-scripts.js

1. my-file-min.css

2、文件命名规范

‘驼峰形式’表示一个完整的命名，‘\_’下划线表示层级

（1）vue文件统一用lowerCamelCase首字母小写的驼峰形式命名。若有子页面，则用‘\_’连接。

例：文章列表页,文章列表的详情页。(两个页面有层级关系的)

articleList.vue

articleList\_detail.vue

3、class命名规范

（1）文件的<nv-loading >标签下面的第一个div的class = 该文件名 = export default 里面的name

（2）<div class='文件名'>下面的第一层div的class =文件名\_xxx

（3）下面的子模块的class可以根据用途命名，必须遵守lowerCamelCase。

4、路由中的命名规则

const 文件名= r => require.ensure([], () => r(require('../.. /文件名')), '文件名')

{

name: '文件名',

meta: {

title: '资讯详情'

},

path: '/文件名/:id',

component: 文件名,

}

5、协议不要指定引入资源所带的具体协议。当引入图片或其他媒体文件，还有样式和脚本时，URLs 所指向的具体路径，不要指定协议部分（http:, https:），除非这两者协议都不可用。不指定协议使得 URL 从绝对的获取路径转变为相对的，在请求资源协议无法确定时非常好用，而且还能为文件大小节省几个字节。

正例

1、<script src="//cdn.com/foundation.min.js"></script>

2、background: url(//static.example.com/images/bg.jpg)

反例：

1、<script src="http://cdn.com/foundation.min.js"></script>

2、background: url(http://static.example.com/images/bg.jpg)

3、脚本加载，为了减少脚本的阻塞影响，需要增加异步方法async。

正例：

<html>

<head>

<link rel="stylesheet" href="main.css">

<script src="main.js" async></script>

</head>

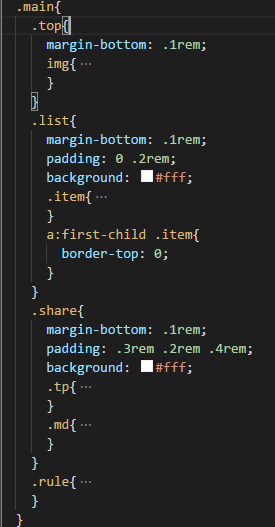
<body>

<!-- body goes here -->

</body>

</html>

1. **HTMl页面规范**
2. 【强制】页面所有层级，尽量使用HTML5新的特性，更佳语义化，例如header、nav、main、article、section、aside、footer、small、strong、em、mark等新增属性，尽量少使用div进行布局。
3. 【强制】在样式定义中，尽量不要使用内联样式，不要在html结构里定义样式，会使页面结构不清晰，如<footer style=”color:#fff”>等类似的结构，在页面完成后，删除冗余代码，注释掉的代码要删除，在html结构里，尽量把逻辑结构放在javascript里面写。
4. **CSS结构规范**
5. 【强制】vue项目中使用的是less，在项目中css的结构要和html整体结构要保持一致，形成梯度，在折叠代码的时候，整体结构清晰。



1. 使用rem进行样式编码的时候，对于0.2rem统一写成 .2rem，加快编码速度，所有颜色都须使用16进制进行表示，如white和#fff，不建议使用white来表示，统一代码结构。
2. 【强制】页面布局中尽量少使用position进行定位，增加页面的扩展性。

4.【强制】ID和class(类)名总是使用可以反应元素目的和用途的名称，或其他通用名称。代替表象和晦涩难懂的名称。

应该首选具体和反映元素目的的名称，因为这些是最可以理解的，而且发生变化的可能性最小。

5.【强制】省略“0”值后面的单位。不要在0值后面使用单位，除非有值。

**（三）Javascript逻辑规范**

1.【强制】代码中的命名不能以下划线或美元符号开始，也不能以下划线或美元符号结束。

反例：\_name / $Object / name\_ / name$/ Object$

2.【强制】代码中的命名严禁使用拼音与英文混合的方式，更不允许直接使用中文的方式。

说明：正确的英文拼音和语法可以让阅读者易于理解，避免歧义。注意，即使纯拼音命名方式也要避免采用。

正例：youtube 国际通用的名称，可视同英文。

反例：dazhe 打折

4.【强制】方法名、参数名、成员变量、局部变量都统一使用lowerCamelCase,必须遵从驼峰形式。

正例：userDetail/tcpUdpDeal

5.【强制】常量命名全部大写，单词间用下划线隔开，力求语义表达完整清楚，不要嫌名字长。

正例：MAX\_STOCK\_COUNT

反例：max\_count

6.【强制】使用多行注释解释函数用途。

例：

/\*\*

\*（有一个空格）设置title的值

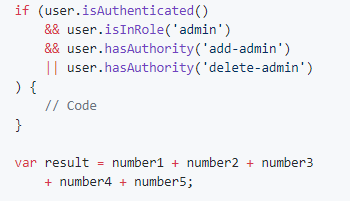
\*/

function setTitle(){};

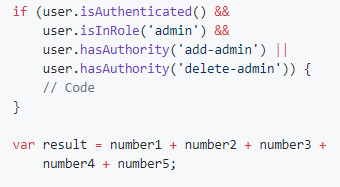
7.【强制】每个独立语句结束后必须换行；每行不得超过120个字符；运算符处换行时，运算符必须在新行的行首。

（解释：超长的不可分割的代码允许例外，比如复杂的正则表达式。长字符串不在例外之列。）

正例：



反例：



8.【强制】不得省略语句结束的分号。

正例：function A(name){

this.name=name;

}

反例：function A(name){

this.name=name

}