

目录

- 目录
- 通用规则
 - 文件编码
 - 文件、文件夹命名规则
 - 缩进
- html代码约定
 - 类命名
- css代码约定
- js代码约定
 - 基础约定
 - js函数命名约定
 - js变量命名约定

通用规则

文件编码

- 所有文件，包括 .html、.css、.js 文件，全部使无BOM头的UTF-8编码。

文件、文件夹命名规则

- 全部采小写形式，以中划线“-”分隔。
- 有复数结构时，采用复数命名法。
- 简洁且有习惯性缩写的单词必须采用容易理解的缩写。
- 例如：my-project/src/my-parser/html-parser.js

缩进

- 使用soft-tab。
- 项目中所有源码使用同样的缩进，使用4个或者2个空格，同一项目保持一致即可，推荐使用2个空格。

html代码约定

类命名

- 标签名必须使用小写字母。
- 属性名必须使用小写字母，属性值必须用双引号包围。
- class 必须单词全字小写，单词间以中划线“-”分隔。
- id必须单词全字小写，单词间以下划线“_”分隔。
- 同一页面id必须保证唯一。
- script标签必须以斜杆尖括号<script/>结尾。
- 例如：

```
<p class='article-paragraph' id='person_desc'>这是我的跟人简介</P>
```

- 建议class 必须代表相应模块或部件的内容或功能，避免直接以样式信息进行命名。
- js引用放在文档末尾，或采用异步加载。
- CSS引用放在head中。
- 例如：

```
<html>
  <head>
    <meta charset="UTF-8">
    <title>我是标题</title>
    <link rel="stylesheet" href="page.css">
  </head>
  <body>
    <!-- a lot of elements -->
    <script src="init.js"/>
  </body>
</html>
```

CSS代码约定

- 选择器与{之间必须包含空格。
- 属性名与之后的冒号“:”之间不允许包含空格，冒号与属性值之间必须包含空格。
- 例如：

```
.selector {
  font-family: Arial, sans-serif;
}
```

- 长度为0时须省略单位，例如：

```
body {
  padding: 0 5px;
}
```

js代码约定

基础约定

- 所有的控制结构都要求使用花括号，例如if，else，for，while以及其他，即使花括号中只有一条语句。例如：

```
var flag = 'red';
if(code === '0') {
  flag = 'blue';
}
```

- 代码块可为空，但应添加相关注释说明。
- 变量必须先声明再使用，尽量遵守就近声明原则。
- 不要在数组的最后一个元素之后添加逗号(在IE的某些版本下会报错或导致数组的长度不正确)。
- 对象中的属性，一般不使用引号，只有当其中包含非法字符时才使用引号，例如：

```
var good = {
  foo: 3,
  bar: 4,
  'data-blah': 5
};
```

- 使用===和!==代替==和!=。
- 在if的判断条件中，对于布尔类型可以不使用比较操作符来判断，但对于字符串和数字，要使用比较操作符来判断。
- 不要将函数的参数命名为arguments,这会覆盖函数作用域提供的内置同名变量。

js函数命名约定

- handleXXX：所有DOM元素事件处理函数以handle开头进行命名；
- _functionName：以下划线开头进行命名的函数，表示页面中非事件处理函数，且使用范围局限于当前页面，可以理解为私有函数；
- functionName：直接以小驼峰命名的函数视为对外提供的公有函数，例如，vue组件外可以通过this.\$refs.refName.functionName()方式调用此类函数；

js变量命名约定

- 函数中的变量定义遵从就近原则，哪里使用在哪里定义；
- 类中的属性统一在类开头定义；