前台编码规范

密级：内部

1. [引言 3](#_bookmark0)
   1. [编写目的 3](#_bookmark1)
   2. [变更记录 3](#_bookmark2)
   3. [定义 3](#_bookmark3)
   4. [参考资料 3](#_bookmark4)
2. [HTML 3](#_bookmark5)
   1. [命名 3](#_bookmark6)
   2. [语法 4](#_bookmark7)
   3. [排版 4](#_bookmark8)
3. [JS 4](#_bookmark10)
   1. [基本规范 4](#_bookmark11)
   2. [命名 5](#_bookmark12)
   3. [变量 5](#_bookmark13)
   4. [方法 5](#_bookmark14)
   5. [对象及控件 5](#_bookmark15)
   6. [第三方库 5](#_bookmark16)
   7. [排版 5](#_bookmark17)
4. CSS 6
   1. 命名 6
   2. 兼容性 6
   3. 第三方库 7
   4. 排版 7

**前台编码规范**

# 引言

## 编写目的

帮助开发团队形成一致的前台编码风格，并指出某些容易引发 BUG 的不良编码习惯， 为了提高工作效率，便于后台人员添加功能及前端后期优化维护，输出高质量的文档，在网站建设中，使结构更加清晰，代码简明有序，有一个更好的前端架构。

## 变更记录

|  |  |  |  |
| --- | --- | --- | --- |
| **版本号** | **作者** | **日期** | **备注** |
| 1.0 | 孙培瑶、田靖 | 2021.9.17 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## 定义

|  |  |  |
| --- | --- | --- |
| **术语** | **定义** | **备注** |
| HTML | Hypertext Markup Language |  |
| JS | Javascript |  |
| CSS | Cascading Style Sheet |  |
|  |  |  |
|  |  |  |

## 参考资料

# HTML

## 命名

* + - 文件名称统一用小写的英文字母、数字和下划线的组合,其中不得包含汉字、空格和特殊字符;
    - 文件名应见名知义，能清晰描述页面主要功能，例如“AddUser.cshtml”，尽量避免使用拼音。
    - 如需缩写单词，则应使用约定俗成的缩写形式，如 btn、nav、num、img 等，不能自造单词，以免引起歧义。

## 语法

* + - 编码：所有编码均采用xhtml/html，标签必须闭合，编码统一为UTF-8，在多语言的网站建议添加<html lang="zh-CN">，说明内容是以中文显示和阅读为基础的。
    - 用两个空格来代替制表符（tab） -- 这是唯一能保证在所有环境下获得一致展现的方法。
    - 对于属性的定义，确保全部使用双引号，绝不要使用单引号。
    - 不要在自闭合（self-closing）元素的尾部添加斜线 -- HTML5 规范中明确说明这是可选的。
    - 不要省略可选的结束标签（closing tag）（例如，</li> 或 </body>）。

## 排版

* + - 嵌套元素应当缩进一次（即两个空格）。
    - HTML 标记内部，不允许出现空行及无意义的空格。
    - 应正确使用缩进来呈现嵌套关系，子元素应比父级元素缩进一个制表位。但仅有内容无子标签的元素，可在同一行书写，例如“<button>Save</button>”。
    - 字符编码在HTML中指定字符编码，让浏览器轻松、快速的确定适合网页内容的渲染方式。字符编码通常设为 UTF-8 。
    - 如标签属性过多过长，可适当在标签内部换行。

# JS

## 基本规范

* + - 阅读性好：如良好的注释和命名规范，有文档。
    - 代码的松耦合，高度模块化：将页面内的元素视为一个个模块，相互独立，尽量避免耦合过高的代码，从html,css,js三个层面都要考虑模块化  
      严格按照规范编写代码。
    - 具有一致性：看起来如同一个人编写。
    - 在页面中引用 JS 文件时，应将引用标签写在<body>标签之后。

## 命名

* + - 文件名全部小写（ 词汇缩写也小写）， 各单词间用“ - ” 为连接符。 例如“jquery-overlay-1.0.js”
    - 驼峰式命名：原生JavaScript变量要求是纯英文字母, 首字母须小写,如iTaoLun;jQuery变量要求首字符为'\_',其他与原生JavaScript规则相同,如:\_iTaoLun;另,要求变量集中声明,避免全局变量。
    - 类命名:首字母大写,驼峰式命名，如ITaoLun;
    - 函数命名:首字母小写驼峰式命名，如iTaoLun();
    - 命名语义化,尽可能利用英文单词或其缩写;
    - 所有变量及方法名都应见名知义，除 i, j ,k 等约定俗成用于集合遍历的临时变量名以外，不允许使用无意义的单个字母做为变量名或方法名。

## 变量

* + - JS 有着非常糟糕的内存及作用域管理机制，如非绝对必要，不要声明全局变量。

## 方法

* + - 为提高兼容性，尽可能少使用 return false; 应以 e.preventDefault(); 代替。

## 对象及控件

* + - 公司目前无专职前台工程师，JS 平均水平偏低。而 JS 自定义对象及控件语法比较复杂，为增强项目可维护性，应尽量避免使用自定义对象及控件，在必要情况下， 也应先与整个团队协商。

## 第三方库

* + - 为增强可维护性，原则上各项目都采用 Jquery 或 AngularJS framework 为基础库，以确保所有人都能看懂相关代码。
    - 原则上仅引入jQuery库, 若需引入第三方库, 须与团队其他人员讨论决定，并认真评估兼容性。
    - 如无绝对正当的理由，同一个项目中，不应该出现同一 JS 库的不同版本。
    - 项目中引用的任何第三方 JS 文件均应存放于独立文件夹中，并不得直接修改原始文件，如果需要扩充或修改第三方 API，应写在另外的文件中。如引用的第三方 JS 库较多，应在项目根目录下维护一份清单。
    - 应尽可能使用CDN 来引用第三方库，以改善访问体验。

## 排版

* + - JS 代码块中不应该出现无意义的空行，尤其是不允许毫无规律地随意换行。
    - 为增强兼容性，每一个完整的语句末尾都应加上分号。
    - 后期优化中,JavaScript非注释类中文字符须转换成unicode编码使用, 以避免编码错误时乱码显示;
    - 在进行链式方法调用时，推荐采用一行一个方法的书写方式，以增强可读性。
    - 代码结构明了,加适量注释，提高函数重用率。

# CSS

**4.1 命名**

* + - 文件名全部小写（词汇缩小也小写），各单词间用“-”为连接符。例如“all-pages.css”
    - css最好用Class来命名，以中划线分隔；命名应反映该元素的功能或使用通用名称，而不要用抽象的晦涩的命名。
    - CSS中的变量和placeholder 使用小写字母，中划线分隔；
    - 标签名，属性名，class名全部用小写，各单词间用“-”为连接符。
    - CSS中的函数、混合采用小驼峰式命名。

**4.2 兼容性**

* + - 为增强兼容性，应尽可能减少使用 CSS3 特性。例如“first-nth-child”属性在 IE8以下浏览器无法加载 。
    - 为增强兼容性，应尽可能减少使用 important 声明。
    - 为增强兼容性，应尽量避免使用标签类型选择器。
    - 避免兼容性属性的使用, 比如text-shadow||css3的相关属性。

**4.3 第三方库**

* + - 推荐使用 bootstrap 为前台 css 框架，当前最新稳定版本为 3.0 。
    - 项目中引用的任何第三方 CSS 文件均应存放于独立文件夹中，并不得直接修改原始

文件，如果需要扩充或修改第三方样式，应写在另外的文件中。如引用的第三方 CSS 库较多，应在项目根目录下维护一份清单。

* + - 应尽可能使用 CDN 来引用第三方库，以改善访问体验。

**4.4 排版**

* + - CSS样式可细分为3类:自定义样式、重新定义HTML样式、链接状态样式。
    - 样式为设计师自定义的新CSS样式,影响被使用本样式的区域,用于完成网页中局部的样式设定。样式名“.”+“相应样式效果描述的单词或缩写”例:“.shadow”文字样式样式名“.no”+“字号”+“行距”+“颜色缩写”例:“.no12”、“.no12-24 ”
    - 重新定义HTML样式为设计师重新定义已有的HTML标签样式,影响全部的被设定标签样式,用于统一网页中某一标签的样式定义。样式名“HTML标签”例:hr { border: 1px dotted #333333 }
    - 连接状态样式为设计师对链接不同状态设定特殊样式,影响被使用本样式区域中的链接。该样式写法有2种:a.nav:link、nav.a:link。第一种只能修饰<a>标签中;第二种可以修饰所有包含有<a>标签的其他标签。
    - 页面内的样式加载必须用链接方式<link rel="stylesheet" type="text/css" href="style/style.css">
    - CSS 文件中不应出现无意义的换行或空格