# ERP系统前端编码规范

## HTML编码规范

1. 缩进使用tab（因为eslint里面缩进两格，建议将tab默认缩进修改为缩进两格）；
2. 嵌套的节点应该缩进；
3. 在属性上，使用双引号，不要使用单引号；
4. 属性名全小写，用中划线做分隔符；
5. 不要在自动闭合标签结尾处使用斜线（[HTML5 规范](http://dev.w3.org/html5/spec-author-view/syntax.html" \l "syntax-start-tag) 指出他们是可选的 --jsx语法除外）；
6. 不要忽略可选的关闭标签，例：</li> 和 </body>
7. 页面结构

* - 导航：nav
* - 页头：header
* - 容器：container
* - 内容：content
* - 页面主体：main
* - 页尾：footer
* - 侧栏：sidebar
* - 栏目：column
* - 面包屑：breadcrumbs
* - 页面外围控制整体布局宽度：wrapper
* - 左右中：left right center

1. 导航

* - 导航：nav
* - 主导航：mainbav
* - 子导航：subnav
* - 顶导航：topnav
* - 边导航：sidebar
* - 左导航：leftsidebar
* - 右导航：rightsidebar
* - 菜单：menu
* - 子菜单：submenu
* - 标题: title
* - 摘要: summary

1. 功能

* - 标志：logo
* - 广告：banner
* - 登陆：login
* - 登录条：loginbar
* - 注册：regsiter
* - 搜索：search
* - 功能区：shop
* - 标题：title
* - 加入：joinus
* - 状态：status
* - 按钮：btn
* - 滚动：scroll
* - 标签页：tab
* - 文章列表：list
* - 提示信息：msg
* - 当前的: current
* - 小技巧：tips
* - 图标: icon
* - 注释：note
* - 指南：guild
* - 服务：service
* - 热点：hot
* - 新闻：news
* - 下载：download
* - 投票：vote
* - 合作伙伴：partner
* - 友情链接：link
* - 版权：copyright

## **HTML5 doctype**

1. 在页面开头使用这个简单地doctype来启用标准模式，使其在每个浏览器中尽可能一致的展现；
2. 虽然doctype不区分大小写，但是按照惯例，doctype大写 （[关于html属性，大写还是小写](http://stackoverflow.com/questions/15594877/is-there-any-benefits-to-use-uppercase-or-lowercase-letters-with-html5-tagname" \t "http://alloyteam.github.io/CodeGuide/_blank)）。

## **lang属性**

1. 根据HTML5规范：
2. 应在html标签上加上lang属性。这会给语音工具和翻译工具帮助，告诉它们应当怎么去发音和翻译。
3. 更多关于 lang 属性的说明[在这里](http://www.w3.org/html/wg/drafts/html/master/semantics.html" \l "the-html-element)；
4. 在sitepoint上可以查到[语言列表](http://reference.sitepoint.com/html/lang-codes)；
5. 但sitepoint只是给出了语言的大类，例如中文只给出了zh，但是没有区分香港，台湾，大陆。而微软给出了一份更加[详细的语言列表](http://msdn.microsoft.com/en-us/library/ms533052(v=vs.85).aspx)，其中细分了zh-cn, zh-hk, zh-tw。

## **字符编码**

1. 通过声明一个明确的字符编码，让浏览器轻松、快速的确定适合网页内容的渲染方式，通常指定为'UTF-8'。

## **IE兼容模式**

1. 用 <meta> 标签可以指定页面应该用什么版本的IE来渲染；<meta http-equiv="X-UA-Compatible" content="IE=Edge">

## **属性顺序**

1. 属性应该按照特定的顺序出现以保证易读性；

* class
* id
* name
* data-\*
* src, for, type, href, value , max-length, max, min, pattern
* placeholder, title, alt
* aria-\*, role
* required, readonly, disabled

1. class是为高可复用组件设计的，所以应处在第一位；
2. id更加具体且应该尽量少使用，所以将它放在第二位。

## **CSS, SCSS**

1. 使用两个空格缩进，
2. 每个属性声明的末尾都要加分号。
3. 以下几种情况不需要空格：

* 属性名后
* 多个规则的分隔符','前
* !important '!'后
* 属性值中'('后和')'前
* 行末不要有多余的空格

1. 以下几种情况下需要空格：

* 属性值前
* 选择器'>', '+', '~'前后
* '{'前
* !important '!'前
* @else 前后
* 属性值中的','后
* 注释'/\*'后和'\*/'前

1. 以下几种情况需要空行：

* 文件最后保留一个空行
* '}'后最好跟一个空行，包括scss中嵌套的规则
* 属性之间需要适当的空行
* css属性书写顺序, 建议遵循:  布局定位属性-->自身属性-->文本属性-->其他属性

## **命名规范**

1. 文件命名规范

* 统一用翻译的英文命名
* 以功能命名
* 路由由左侧菜单来命名

1. 图片命名规范

* 图片的名称分为头尾两部分，用下划线隔开，头部表示此图片的大类性质，例如标志，菜单，按钮等
* banner：放置在页面顶部的广告，装饰图案等长方形的图片
* logo：标志性的图片
* button：在页面上位置不固定，并且带有链接的小图片
* menu：在页面中某一位置连续出现，性质相同的链接栏目的图片
* pic：装饰用的图片

例子：

* Banner\_sohu.gif, banner\_sina.gif
* Menu\_aboutus.gif, menu\_job.gif
* Title\_news.gif
* Logo\_police.gif
* Pic\_people.gif

1. JS命名

* 变量命名: 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;

1. Css命名

* class与id的使用: id是唯一的并是父级的, class是可以重复的并是子级的, 所以id仅使用在大的模块上, class可用在重复使用率高及子级 中; id原则上都是由我分发框架文件时命名的, 为JavaScript预留钩子的除外;

## **注释规范**

1. 文档注释

/\*

\* @desc 中文说明

\*/

1. 模块注释

/\* @module Header by 张三 \*/

/\* End Header \*/

3.单行注释

.class {

/\* 单行注释 \*/

...

}

1. 行尾注释

.class {

... /\* 行尾注释 \*/

}

二、 JS注释

1.文档注释

/\*

\* @func

\* @desc 一个带参数的函数

\* @param {string} a - 参数a

\* @param {number} b=1 - 参数b默认值为1

\* @param {string} c=1 - 参数c有两种支持的取值</br>1—表示x</br>2—表示xx

\* @param {object} d - 参数d为一个对象

\* @param {string} d.e - 参数d的e属性

\* @param {string} d.f - 参数d的f属性

\* @param {object[]} g - 参数g为一个对象数组

\* @param {string} g.h - 参数g数组中一项的h属性

\* @param {string} g.i - 参数g数组中一项的i属性

\* @param {string} [j] - 参数j是一个可选参数

\*/

function foo(a, b, c, d, g, j) {

...

}

2.多行注释

最少三行, '\*'后跟一个空格，具体参照右边的写法；

建议在以下情况下使用：

难于理解的代码段

可能存在错误的代码段

浏览器特殊的HACK代码

业务逻辑强相关的代码

/\*

\* one space after '\*'

\*/

var x = 1

3.单行注释

双斜线后，必须跟一个空格；

缩进与下一行代码保持一致；

可位于一个代码行的末尾，与代码间隔一个空格。

1. JavaScript注释, 单行注释使用'//这儿是单行注释' ,多行注释使用 /\* 这儿有多行注释 \*/;
2. 其他js规范使用eslint规范；

## **其他规范** 开发过程中严格按分工完成页面, 以提高css复用率, 避免重复开发; 减小沉冗代码, 书写所有人都可以看的懂的代码. 简洁易懂是一种美德. 为用户着想, 为服务器着想.