**智能中心开发指南与规范**

**(JavaScript代码规范篇)**

**V0.1**

2018年6月

**上海电气风电集团有限公司**

目录

文档类别使用对象 2

**1** **引言** 3

1.1 目的 3

1.2 范围 3

1.3 术语定义 3

1.4 参考资料 3

1.5 版序控制记录 3

1.6 版本更新记录 4

2 JavaScript代码规范 5

2.1 常量/变量/函数/类命名规范 5

2.2 空格与运算符 5

2.3 代码缩进 5

2.4 语句规则 5

2.5 变量声明 6

2.6 变量声明 6

2.7 比较操作 7

2.8 其他 7

2.9 注释规范 7

文档类别使用对象

*文档类别*

该文档是为上海电气风电集团智能中心软件开发代码与指南的规范性文件。(Note.JS部分)

*使用对象*

该文档使用对象为智能中心软件的设计、开发和外包软件开发人员以及智能中心的部门经理等管理人员。

## **引言**

## 目的

本文档是为规范公司智能中心软件开发设计、编码而制定的。

## 范围

本文档为智能中心软件开发人员、产品经理以及外包人员提供有关软件开发设计规范的相关内容，包括：

* JavaScript代码规范

注：针对ES6后续版本

## 术语定义

## 参考资料

## 版序控制记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版序状态 | 拟稿 | 审核 | 批准 | 发布日期 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## 版本更新记录

\***A** - 增加 **M** - 修改 **D** - 删除

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本/修订版 | 修改页码 | 修改记录 | 修改人 | 日期 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## JavaScript代码规范

## 常量/变量/函数/类命名规范

1. 应该是有意义而且易于理解的，最好是能够表达字段含义的英文单词或缩写。
2. **变量小驼峰命名**，首字母小写，之后每单词首字母大写，如：towerHeight 代表塔高度
3. **函数名采用小驼峰命名，如果没有特殊情况，必须使用动词或判断词汇**
4. 定义一次后固定不变的变量，必须是由const定义为常量
5. **常量名全部使用大写**，单词间用”\_”，如：TOWER\_HEIGHT 代表塔高度
6. **类名采用大驼峰命名。**
7. 禁止使用数据类型、保留关键字等作为变量/函数名，如：datetime，password，name，time，index等。

## 空格与运算符

通常运算符 ( = + - \* / ) 前后需要添加空格,建议存盘前使用工具自带格式化功能格式化后，再上传代码。

## 代码缩进

通常使用 4 个空格符号来缩进代码块，禁止使用Tab进行缩进！

## 语句规则

简单语句的通用规则:一条语句通常以分号作为结束符将左花括号放在第一行的结尾。

左花括号前添加一空格。

将右花括号独立放在一行。

不要以分号结束一个复杂的声明。分成多行声明。

## 变量声明

对象定义的规则:

将左花括号与类名放在同一行。

冒号与属性值间有个空格。

在Node中使用字符串最外层使用双引号（虽然单引号意义相同，但为了统一，都用双引号）。

将右花括号独立放在一行，并以分号作为结束符号。

逗号不要出现在行首。

var person = {  
    firstName : "John",  
    lastName: "Doe",  
    age: 50,  
    eyeColor: "blue"  
 };

每行代码字符小于 120

为了便于阅读每行字符建议小于数 120 个。

如果一个 JavaScript 语句超过了 120 个字符，建议在 运算符或者逗号后换行。

## 变量声明

永远在声明 (let或const)之后再使用变量，避免污染全局上下文。

没有特殊必要，尽可能避免使用var。

每行声明都应该带上(let或const)，而不是只有一个(let或const)：

推荐的代码：

let i=1;

let j=2;

不推荐的代码：

let i,j;

let i=1,j=2;

## 比较操作

如果没有特殊情况必须使用'==='代替'=='

尽量使用 `{} []`代替`new Object() new Array()`，不要使用`string bool number`对象类型，

即不要调用`new String new Boollean new Number`。

## 其他

如果没有特殊情况，禁止使用 with和evl

异步转同步执行尽量使用async/await进行

由于node是异步执行的，所以，尽量不要使用回调外Try/Catch，共通会捕获所有未知异常。

 回调函数总是把err参数作为第一个变量，如果参数中有回调函数，总是放在最后

比如  callback(err, param1, param2, callback)

对于复杂判断，建议使用有意义的变量定义判断条件，如：

var isValidPassword = passwordLength >= 4 && /^(?=.\*\d).{4,}$/.test(password);  
if (isValidPassword) {  
    console.log('winning');  
}

行末禁止存在空格，行间空行不得大于1行

## 注释规范

永远使用//的单行注释

每个文件必须拥有头注释，格式如下

// 系统模块：XXXX

// 子功能模块：XXXXX

// 文件功能概述：XXXXXXXX

// 履历 xxxx xxxx/xx/xx 做成

每个函数头必须有头注释，格式如下

// 函数名称（中文）： XXXX

// 概要说明：XXXX

// @param：XXX XXXX

// @return：XXX

单行注释建议注释独立一行便于统计，若实在影响阅读，可放在一行

判断注释放在{}中第一行。

If （a===1）{

//XXXX

}else{

//XXXX

}

注释率要求： 注释行数/总行数（不包括空行）>=20%

禁止出现无用代码注释，禁止出现错误的代码注释，代码完成后，请仔细检查自己的注释！

ajax使用规范

参数长度可能大于50个字符或者参数内存在敏感信息、安全要求较高的情况下，必须使用POST。