

NMSI 小组

Code Style 规范文档

涉及语言：Java，Java Script，Html，Css

本篇规范：Java

1. 文件名格式

| 文件类别 | 文件后缀 |
|-----------|--------|
| Java源文件 | .java |
| Java字节码文件 | .class |

2. 缩进

以 4 个空格为缩进的一个单位。

3. 行长度

尽量避免一行的长度超过 80 个字符。

编写技术文档中的代码解释一行不超过 70 个字符。

4. 换行

当一个表达式无法容纳在同一行内时，依据以下规则断开

- a) 在一个逗号后面断开
- b) 在一个操作符前面断开
- c) 新行应该与上一行同一级别表达式式的开头对齐
- d) 若以上规则导致代码排版混乱，就代之以 8 个空格。

5. 注释

a) 块注释

- i. 块注释之首应该有一个空行，例：

```
/*  
 * Here is a block comment.  
*/
```

b) 单行注释

- i. 短注释与其后的代码具有一样的缩进层次，例：

```
if (condition) {  
  
    /* Handle the condition. */  
    ...  
}
```

c) 尾端注释

i. 极短的注释可以与描述的代码位于同一行，但应留有足够的空白来分离，且若有多个极短的尾端注释，它们应该有相同的缩进，例：

```
if (a == 2) {  
    return TRUE;           /* special case */  
} else {  
    return isPrime(a);      /* works only for odd a */  
}
```

6. 声明

a) 一行一个声明，例：

```
int level; // indentation level  
int size;  // size of table
```

b) 在代码块开始处声明变量，例：

```
void myMethod() {  
    int int1 = 0;           // beginning of method block  
  
    if (condition) {  
        int int2 = 0;      // beginning of "if" block  
        ...  
    }  
}
```

c) 类和接口

- i. 在方法名与其参数列表之前的左括号间不留空格。
 - ii. 左大括号位于声明语句同行的末尾。
 - iii. 右大括号另起一行，与相应的声明语句对齐。
 - iv. 空语句则{}相邻。
- 例：

```
class Sample extends Object {  
    int ivar1;  
    int ivar2;  
  
    Sample(int i, int j) {  
        ivar1 = i;  
        ivar2 = j;  
    }  
  
    int emptyMethod() {}  
  
    ...  
}
```

7. 语句

a) 简单语句

i. 每行至多一条，例：

```
argv++;           // Correct  
argc--;           // Correct  
argv++; argc--;   // AVOID!
```

b) 复合语句

- i. 被括其中的语句应该较之复合语句缩进一个层次
- ii. 左大括号"{"应位于复合语句起始行的行尾；右大括号"}"应另起一行并与复合语句首行对齐。
- iii. 大括号可以被用于所有语句，包括单个语句，只要这些语句是诸如 **if-else** 或 **for** 控制结构的一部分。这样便于添加语句而无需担心由于忘了加括号而引入 **bug**。

c) 返回语句

- i. 一个带返回值的语句不使用小括号()

d) **If, if-else, if else-if else** 语句

- i. 应具有如下格式

```
if (condition) {  
    statements;  
}
```

```
if (condition) {  
    statements;  
} else {  
    statements;  
}
```

```
if (condition) {  
    statements;  
} else if (condition) {  
    statements;  
} else {  
    statements;  
}
```

e) **for** 语句

- i. 应具有如下格式

```
for (initialization; condition; update) {  
    statements;  
}
```

- ii. 一个空的 **for** 语句应具有如下格式

```
for (initialization; condition; update);
```

f) **While** 语句

- i. 应具有如下格式

```
while (condition) {  
    statements;  
}
```

一个空的**while**语句应该具有如下格式:

```
while (condition);
```

g) **Do-while** 语句

```
do {
    statements;
} while (condition);
```

h) Switch 语句

i. 应具有如下格式

```
switch (condition) {
case ABC:
    statements;
    /* falls through */
case DEF:
    statements;
    break;

case XYZ:
    statements;
    break;

default:
    statements;
    break;
}
```

i) try-catch 语句

i. 应具有如下格式

```
try {
    statements;
} catch (ExceptionClass e) {
    statements;
}
```

8.空行

a) 一个空行

- i. 一个源文件的两个片段间
- ii. 类声明和接口声明之间

b) 两个空行

- i. 两个方法之间
- ii. 方法内的局部变量和方法的第一条语句之间
- iii. 块注释或单行注释之前
- iv. 一个方法内的两个逻辑段之间，用以提高可读性

c) 空格

- i. 一个紧跟着括号的关键字应该被空格分开，例：

```
while (true) {
    ...
}
```

- ii. 空白应该位于参数列表中逗号的后面
- iii. 所有的二元运算符，应该使用空格将之与操作数隔开

```
a += c + d;
a = (a + b) / (c * d);

while (d++ = s++) {
    n++;
}
printSize("size is " + foo + "\n");
```

- iv. For 语句中的表达式应该被空格隔开，例：

```
for (expr1; expr2; expr3)
```

- v. 强制类型转换后应该跟一个空格，例：

```
myMethod((byte) aNum, (Object) x);
myMethod((int) (cp + 5), ((int) (i + 3)) + 1);
```

9.编程惯例

- a) 提供实例以及类变量的访问控制
 - i. 若无足够的原因，不把实例与变量声明为公有
- b) 常量
 - i. 位于 for 循环中作用计数器值得数字常量，除-1, 0,1 外，不应被直接写入代码
- c) 变量赋值
 - i. 避免在一个语句中给多个变量赋值
 - ii. 不使用内嵌赋值运算符，例：

```
d = (a = b + c) + r;           // AVOID!
```

应该写成

```
a = b + c;
d = a + r;
```

- d) 圆括号
 - i. 在含有多种运算符的表达式中使用圆括号来避免运算符优先级问题，例：

```
if (a == b && c == d)           // AVOID!
if ((a == b) && (c == d))       // RIGHT
```

NMSI 小组

2017 年 4 月 8 日

NMSI 小组

Code Style 规范文档

涉及语言：Java，Java Script

本篇规范：JavaScript

5.变量与声明

- a) 每个变量声明都要加上 `var` 关键字
- b) 变量再函数内部或循环控制条件之前声明
- c) 私有变量函数前置 `_` 标识
- d) 变量命名规则具象化,布尔值需带 `is,has,can`,单位值带 `_ms,_s,_px` 等
- e) 构造器首字母大写，方法变量需要遵循驼峰命名规则

6.常量

- a) 常量需要大写
- b) 常量需要定义在文件头部，并使用 `_` 分割

7.分号与逗号

- a) 赋值，定义，返回值，方法调用后强制需要加分号
- b) 定义变量时，不使用逗号
- c) `objectMap` 最后一个属性不带逗号

8.函数

- a) 函数内部不允许嵌套函数。
- b) 函数内部不允许使用 `arguments.callee` 和 `arguments.caller`
- c) 函数参数不得超过 5 个（多于 5 个使用 `objectType` 代替）

9.异常

- a) 函数参数不得超过 5 个（多于 5 个使用 `objectType` 代替）

10.循环

- a) 循环使用 `for`，不在非必要条件下不使用 `while`
- b) 循环一致使用自增
- c) 循环内不允许定义函数和变量

11.控制逻辑

- a) 控制条件内不要使用 `var` 定义变量
- b) 分支超过 5 种的判断使用 `switch case`，`switch case` 需要带 `default`
- c) 判断条件过长，使用变量标识
- d) 判断中禁止使用 `yoda` 表达式
- e) 判断使用严格类型判断 `0,null,undefined`，固定字符使用 `===`

12.字符串

- a) 允许多行字符，使用 `+` 或者 `\` 分割

13.原型

- a) 禁止对顶级对象进行原型修改
- b) 原型上不允许保存属性，只能保存方法

14.注释

- a) 文件开头必须要有文件说明注释，时间，作者
- b) 多行单行注释都被允许

NMSI 小组

2017 年 4 月 9 日