

# 项目开发编码标准

日期	修订版本	修改描述	作者	审核
2019-5-1	1.0	草拟	田佳乐	李佩如

作者： 田佳乐

版本： 1.0

日期： 2019-5-1

# 目录

一、 目的.....	3
二、 概述.....	3
三、 编码标准.....	3
3.1 有关命名.....	3
3.1.1 包命名.....	3
3.1.2 类命名.....	4
3.1.3 方法命名.....	4
3.1.4 常量命名.....	4
3.1.5 变量命名.....	4
3.1.6 资源文件命名.....	5
3.2 有关注释.....	5
3.2.1 程序头注释.....	5
3.2.2 关键点注释.....	5
四、 格式规范.....	6
4.1 缩进.....	6
4.2 换行.....	6
4.3 对齐.....	7
五、 写在后面.....	7
六、 补充.....	7

## 一、目的

为了使软件开发过程顺畅，保证软件质量，于是我们在进行开发之前编撰了这份编码标准文档。

## 二、概述

- \*项目以功能模块来划分分工；
- \*数据库不能随意修改，本地数据库须统一；
- \*有标准的 Java 代码风格和前端代码风格，有良好习惯；
- \*时刻考虑代码的可复用性；
- \*根据项目进度安排，每个人在需要时应提供每周项目进度报告；

## 三、编码标准

### 3.1 有关命名

总体原则上，变量名与方法命名时应该遵循良好的命名习惯，尽量使得命名能够体现变量或者方法本身的含义。

#### 3.1.1 包命名

- (1) 包名采用全小写命名；
- (2) 包名应符合“com.项目名.包内文件用图”的基本范式，例如“com.jwglxt.data”表示项目代号为 jwglxt 的项目中用来处理数据的类组成的包；

### 3.1.2 类命名

(1) 所有类命名必须使用正确的英文和语法，做到见名知意。严禁使用拼音和英文混合的方式，更不允许直接使用中文的方式。正确的英文拼写可以让读者更易于理解，避免歧义。注意：即使使用纯拼音命名方式也要避免采用。

(2) 所有 Servlet 类采用 lowerCamelCase 风格，必须遵从驼峰形式。

例如：addStudent.java

(3) 所有除 Servlet 类外的类均以大写字母开头并遵从驼峰形式。

例如：Student.java

### 3.1.3 方法命名

方法命名统一采用 lowerCamelCase 风格，必须遵从驼峰形式。

例如：showMyScore()

### 3.1.4 常量命名

常量名全部大写，单词间用下划线隔开，力求语义表达完整清楚，不要嫌名字长。

例如：MAX\_STOCK\_COUNT

### 3.1.5 变量命名

变量名（成员变量、局部变量）、参数名都统一使用 lowerCamelCase 风格，必须遵从驼峰形式。

例如：int numOfStudent

### 3.1.6 资源文件命名

所有资源文件，包括.java,.css,.js 等，均需要使用英文表述，做到见名知意。

## 3.2 有关注释

团队成员都应该形成良好的写注释的习惯，方便以后阅读，以及为了后期生成可读性良好的代码。

### 3.2.1 程序头注释

每个代码文件头部第一行必须将该代码的作用加以注释，以方便其他读者更快地明白这段代码所做的工作。注释格式可自由选择。

### 3.2.2 关键点注释

应该包含如下信息：

- \* 一些程序关键的地方 ；
- \* 一些程序不易读的地方 ；
- \* 编写代码过程中遇到问题的地方 ；
- \* 需要提示读者的地方；
- \* 注释易懂 ；

## 四、格式规范

### 4.1 缩进

\*应注意使用 format 来格式化代码，使用 Tab 键来缩进代码，相当于 4 个空格。

折行使用 120 字符宽度；

\*html 文件内，子元素和父元素之间应有一个标签位的缩进

### 4.2 换行

\*{}花括号应该另起一行，左花括号与方法名、类名在同一行。(除了数组初始化时的花括号)；

\*if、while 等语句，假如体内只有一句代码也不要省略{}，为了方便以后的增删；

\*字符串过长考虑拆分成多行；

\*{}括号等应该对齐；

\*类和方法的块注释必须紧贴类和方法；

\*单独起行的//注释必须对齐被注释语句；

\*对某一代码块进行解释说明的注释最好使用//另起一行，并和所注释的代码块第一行代码缩进相同；

\*html 文件内，成对出现的标签内的内容过长建议将标签的起始和结束单独分为两行，代码单独作为一行；

\*html 文件内，单独出现的标签在<和>之间不宜进行换行；

## 4.3 对齐

\*html 文件内，同级别和相似代码缩进位置应该对齐；

\*}应与代码块起始字母对齐；

\*统一代码块中的语句的缩进应该对齐；

\*方法之间缩进应该对齐；

\*类中属性之间应对齐，方法之间也应该对齐；

## 五、写在后面

希望各位成员遵守这份开发规范文档，养成良好的开发习惯；

## 六、补充

如果有问题，请及时反馈。