

(一) 命名风格

- 1.所有编程相关的命名均不能以下划线或美元符号开始，也不能以下划线或美元符号结束。
- 2.所有编程相关的命名严禁使用拼音与英文混合的方式，更不允许直接使用中文的方式。
- 3.方法名、参数名、成员变量、局部变量都统一使用 lowerCamelCase 风格 (驼峰命名法)。
- 4.常量命名应该全部大写，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。
- 5.避免在子父类的成员变量之间、或者不同代码块的局部变量之间采用完全相同的命名，使可理解性降低。
- 6.杜绝完全不规范的英文缩写，避免望文不知义。

(二) 代码格式

- 1.若函数或过程中的参数较长，则要进行适当的划分。
- 2.注释的双斜线与注释内容之间有且仅有一个空格。
- 3.方法参数在定义和传入时，多个参数逗号后面必须加空格。
- 4.程序块要采用缩进风格编写，缩进的空格数为 4 个，避免使用 Tab 进行缩进。
- 5.较长的语句 (>80 字符) 要分成多行书写，长表达式要在低优先级操作符处划分新行，操作符放在新行之首，划分出的新行要进行适当的缩进，使排版整齐，语句可读。
- 6.循环、判断等语句中若有较长的表达式或语句，则要进行适应的划分，长表达式要在低优先级操作符处划分新行，操作符放在新行之首。
- 7.不允许把多个短语句写在一行中，即一行只写一条语句。
- 8.if、for、do、while、case、switch、default 等语句自占一行，且 if、for、do、while 等语句的执行语句部分无论多少都要加括号 {}。

(三) 注释规约

1.类、类属性、类方法的注释必须使用 Javadoc 规范，使用 `/** 内容 */` 格式，不得使用 `// xxx` 方式。

2.所有的抽象方法（包括接口中的方法）必须要用 Javadoc 注释、除了返回值、参数异常说明外，还必须指出该方法做什么事情，实现什么功能。

3.方法内部单行注释，在被注释语句上方另起一行，使用 `//` 注释。方法内部多行注释使用 `/* */`注释，注意与代码对齐。

4.代码修改的同时，注释也要进行相应的修改，尤其是参数、返回值、异常、核心逻辑等。

5.好的命名、代码结构是自解释的，注释力求精简准确、表达到位。避免出现注释的另一个极端：过多过滥的注释，代码的逻辑一旦修改，修改注释又是相当大的负担。

6.选择熟悉的、可以正确表达代码含义的语言进行注释，不必追求英文注释。

(四) 可读性

1.注意运算符的优先级，并用括号明确表达式的操作顺序，避免使用默认优先级。

2.避免使用不易理解的数字，用有意义的标识来替代。

3.源程序中关系较为紧密的代码应尽可能相邻。