# Java技术管理文档（48条）

## 格式化规则（12条）

### a. 强制

1. 每个新的代码块或类似代码块的构造打开时，缩进增加两个空格。
2. 每条语句后必须换行。
3. Java 代码每行字符限制为 100 个，除特殊情况外，超出此限制的行必须换行。
4. 不允许使用制表符缩进，只能使用空格。
5. 在类的每个方法之间必须添加一个空行，以区分不同的逻辑单元。
6. 大括号 {} 后必须换行，不允许在大括号后直接写代码，除非是空代码块。
7. 块注释的每一行必须左对齐，且注释内容与注释符号之间必须有一个空格。

### b. 推荐

1. 尽量避免一行代码过长，即使在 100 字符限制内，若逻辑复杂也应适当换行以提高可读性。
2. 在逻辑上相关的代码块之间可以添加一个空行，以增强代码的可读性。
3. 对于较长的表达式（如方法调用、条件语句等），应在逻辑操作符后换行，并将后续代码缩进两个空格。

### c. 允许

1. 在某些特殊情况下，如长 URL 或特定注释中的命令行，可以超出 100 字符限制。
2. 对于单行注释，可以在同一行中直接跟在代码后面，但需确保注释内容与代码之间有一个空格。

## 命名规则（12条）

### a. 强制

1. 包名和模块名只能使用小写字母和数字，连续单词直接拼接，不使用下划线。
2. 类名使用 UpperCamelCase，通常是名词或名词短语。
3. 方法名使用 lowerCamelCase，通常是动词或动词短语。
4. 常量名使用 UPPER\_SNAKE\_CASE，所有大写字母，单词之间用单个下划线分隔。
5. 非常量字段名使用 lowerCamelCase。
6. 参数名使用 lowerCamelCase，公共方法的参数名避免使用单字符。
7. 接口名必须以 I 开头，后跟 UpperCamelCase，例如 IUserRepository。
8. 自定义异常类名必须以 Exception 结尾，例如 InvalidInputException。
9. 枚举类名必须使用 UpperCamelCase，且应具有明确的语义，例如 Weekday。

### b. 推荐

1. 测试类名以 Test 结尾，如果是针对单个类的测试，则为该类名加 Test 后缀。
2. 方法名尽量使用描述性的动词短语，避免使用过于简略的命名。

### c. 允许

1. 对于局部变量，可以在不引起歧义的情况下使用较短的命名。

## 编程实践（10条）

### a. 强制

1. 必须使用大括号 {}，即使是空的代码块或只有一条语句。
2. switch 语句必须包含 default 分支。
3. 静态成员变量和方法的修饰符必须按照 Java 语言规范的推荐顺序出现。
4. 不允许使用通配符导入。
5. 不要静默失败，应始终报告错误。

### b. 推荐

1. 方法重载应连续出现，不要被其他代码分隔。
2. 尽量限制变量的作用域。
3. 在复杂的代码段添加注释，说明其意图和逻辑。
4. 遵循单一职责原则，将代码分解为更小的方法和类。

### c. 允许

1. 在某些情况下，可以使用匿名内部类来简化代码，但需注意可读性。

## 注释和文档（5条）

### a. 强制

1. 类和方法必须有 Javadoc 注释，且格式符合规范。
2. 注释中不得包含误导性或错误的信息。

### b. 推荐

1. 在代码中适当添加注释，解释复杂的逻辑。
2. 对于重要的设计决策或算法，可以在注释中提供详细的说明。

### c. 允许

1. 对于非常简单的代码，可以省略注释，但需确保代码的自描述性。

## 错误处理（5条）

### a. 强制

1. 不要捕获异常后直接忽略，至少应记录日志。
2. 不要隐藏错误的根本原因，应提供具体的错误信息。

### b. 推荐

1. 在可能的情况下，尽早抛出错误。
2. 使用标准的错误代码和消息，便于监控和调试。

### c. 允许

1. 在某些低风险的场景下，可以简化错误处理逻辑，但需确保不会影响系统的稳定性。

## 其他（4条）

### a. 强制

1. 不得使用 C 风格的数组声明。
2. 文件编码必须使用 UTF-8。

### b. 推荐

1. 在类中使用 final 修饰符，避免不必要的继承。

### c. 允许

1. 在不影响代码可读性和可维护性的情况下，可以适当使用一些简化的代码结构