

Java 技术管理规范文档

文曼谕 2022141461053

1.代码风格

a) 强制

1. 类名必须使用大驼峰式命名法 (UpperCamelCase)
2. 方法名、变量名必须使用小驼峰式命名法 (lowerCamelCase)
3. 常量名必须全部大写, 单词间用下划线分隔
(UPPER_CASE_WITH_UNDERSCORES)
4. 每行代码长度不得超过 120 个字符
5. 必须使用 4 个空格进行缩进, 禁止使用 Tab 键
6. 大括号必须遵循 K&R 风格 (左大括号不换行)
7. 所有覆写的方法必须加上@Override 注解
8. 禁止使用魔鬼数字, 必须定义有意义的常量
9. 所有异常必须被捕获或声明抛出, 禁止捕获 Exception 基类

b) 推荐

10. 包名全部小写, 使用反转域名格式 (如 com.company.project)
11. 方法参数不超过 5 个, 过多时应考虑重构为对象
12. 类成员变量应该放在类的顶部, 方法之间用空行分隔
13. 使用空格而非 Tab 进行缩进
14. 使用空行分隔逻辑相关的代码块
15. 使用 JavaDoc 为公开的类和方法添加注释
16. 优先使用接口而非具体实现类声明变量

c) 允许

17. 在简单的 getter/setter 方法上使用 lombok 的@Data 注解
18. 在简单的循环中使用单行 if 语句 (但必须包含大括号)
19. 在测试代码中使用较短的变量名 (如 i, j, k)

20. 在 lambda 表达式中使用单字母参数名

2. 编码实践

a) 强制

- 21. 必须使用 try-with-resources 处理所有实现 AutoCloseable 的资源
- 22. 必须对用户输入、外部系统调用进行参数校验
- 23. 必须处理所有受检异常，不能简单忽略或打印堆栈
- 24. 禁止在循环中进行数据库连接操作
- 25. 必须使用线程安全的日期格式化类（如 DateTimeFormatter）
- 26. 集合返回必须使用 Collections.emptyList() 而非 null
- 27. 必须为所有序列化类定义 serialVersionUID

b) 推荐

- 28. 优先使用 Java 8 的日期时间 API（java.time 包）
- 29. 使用 Optional 代替 null 作为方法可能无值的返回
- 30. 使用 StringBuilder 进行字符串拼接
- 31. 使用枚举代替整型常量
- 32. 使用 @VisibleForTesting 限制测试专用方法的可见性
- 33. 优先使用不可变对象（Immutable Objects）
- 34. 使用静态工厂方法代替多个构造器

c) 允许

- 35. 在性能关键路径使用原生类型而非包装类
- 36. 在内部工具类中使用方法链（Method Chaining）
- 37. 在特定场景使用双重检查锁定模式
- 38. 在明确线程安全的上下文中使用非线程安全集合

3. 工具与框架

a) 强制

- 39. 必须使用统一的构建工具（Maven 或 Gradle）

- 40. 必须配置代码格式化模板并统一团队使用
- 41. 必须使用静态代码分析工具（如 SonarQube）
- 42. 必须使用版本控制系统（Git）并遵循分支策略

b) 推荐

- 43. 使用日志框架（SLF4J+Logback）而非 System.out
- 44. 使用 Guava 或 Apache Commons 等经过验证的工具库
- 45. 使用 JUnit 5 作为单元测试框架
- 46. 使用 Mock 框架（如 Mockito）进行测试隔离

c) 允许

- 47. 在特定场景使用 Lombok 减少样板代码
- 48. 在性能分析时使用 JMH 进行微基准测试
- 49. 在复杂业务逻辑中使用领域特定语言（DSL）
- 50. 在适当场景使用函数式编程风格