

Java 技术管理规范文档

a. 强制规则（违反将导致功能缺陷/安全风险，CI 流程必须拦截）

1. 类名使用 **UpperCamelCase**，方法/变量使用 **lowerCamelCase**，常量使用 **UPPER_SNAKE_CASE**
2. 包名全小写，不使用下划线（如 **com.example.project**）
3. 大括号与代码同行，**if/for** 等单行逻辑必须加 **{}**
4. 禁止捕获 **Throwable** 或 **Exception**（需明确异常类型）
5. 自定义异常需继承 **RuntimeException** 或标准异常类
6. **finally** 块中禁止 **return** 或抛出异常
7. 线程池必须通过 **ThreadPoolExecutor** 显式创建（禁用 **Executors**）
8. 共享资源访问需加锁（**synchronized** 或 **ReentrantLock**）
9. 集合初始化需指定容量（如 **new ArrayList<>(16)**）
10. 禁止在 **foreach** 循环中修改集合（用 **Iterator.remove()**）
11. 使用 **try-with-resources** 管理所有 **Closeable** 资源
12. **finally** 块中必须释放锁、连接等资源
13. 公有类/方法必须用 **Javadoc** 注释（含参数、返回值说明）
14. 提交代码时禁止包含 **TODO** 或 **FIXME** 注释
15. 正则表达式需预编译（**Pattern.compile()**复用）
16. 类职责单一（方法行数 ≤ 80 行）
17. 禁止 **System.out** 直接输出（统一 SLF4J 日志）
18. 单元测试覆盖率 $\geq 80\%$ （核心逻辑需 100%）
19. 集成测试需标记 **@IntegrationTest** 并隔离运行
20. 用户输入必须校验（防 SQL 注入、XSS）
21. 外部服务调用需超时控制（如 **Feign.clientTimeout**）

b. 推荐规则（Code Review 重点检查）

1. 缩进统一用 4 个空格，禁用 **Tab** 字符
2. 单行长度 ≤ 120 字符，超长需换行对齐
3. 异常信息需包含上下文（如 `throw new UserNotFoundException("UserID: "+id)`）
4. 使用 **AtomicXXX** 类替代 **volatile** 基础类型
5. 泛型需显式声明类型（禁用 **Raw Type**）
6. 数据库连接、HTTP 客户端等资源必须复用（连接池化）
7. 废弃代码用 **@Deprecated** 标注并说明替代方案
8. 字符串拼接用 **StringBuilder**（循环内禁用+）
9. 日志输出需判断级别（`logger.isDebugEnabled()`避免拼接开销）
10. 接口与抽象类优先于具体类（依赖倒置）
11. 测试类名以 **Test** 结尾（如 **UserServiceTest**）
12. 敏感数据（密码、密钥）禁止硬编码或日志输出
13. 序列化接口需显式定义 **serialVersionUID**

c. 允许规则（视场景灵活使用）

1. 优先使用 JDK 内置异常（如 **IllegalArgumentException**）
2. 高并发场景优先使用 **ConcurrentHashMap** 而非 **Collections.synchronizedMap()**
3. 优先 **Arrays.asList()**替代循环初始化集合
4. 复杂算法需添加行内注释（解释意图而非复述代码）
5. 避免频繁拆装箱（优先原始类型集合如 **IntStream**）
6. 使用枚举替代常量类（如 **enum Status{ACTIVE,INACTIVE}**）
7. 使用 **@ParameterizedTest** 覆盖多边界条件
8. 使用 **Objects.requireNonNull()**校验参数非空

参考：Google Java Style Guide, Alibaba Java Coding Guidelines, Oracle Java Conventions