

Java 技术管理规范文档

一、命名规范

编号	规则内容	级别
1	类名使用 PascalCase（如 UserServiceImpl）	a
2	方法名使用 camelCase（如 getUserById）	a
3	常量全大写，用下划线分隔（如 MAX_SIZE）	a
4	包名全小写，禁止使用下划线（如 com.example.service）	a
5	Boolean 类型变量以 is 或 has 开头（如 isActive）	b
6	枚举命名使用单数形式（如 Status）	b
7	接口名以 I 开头或以 able 结尾（如 IService, Runnable）	b
8	临时变量避免使用 temp、data 等模糊命名	b
9	方法参数命名不得使用缩写，如 usr、val	a
10	单字符命名仅用于常见循环变量（如 i, j）	a

二、代码风格

编号	规则内容	级别
11	每行代码长度不超过 120 个字符	a
12	缩进使用 4 个空格，禁止 Tab	a
13	if/else/for 等语句必须加 {}	a
14	switch 必须包含 default 分支	a
15	类、方法、字段之间加空行分隔	b

编号	规则内容	级别
16	一行只写一条语句，禁止多语句同行	a
17	Lambda 表达式优先使用简洁语法	b
18	Java 文件结构顺序建议：常量 > 属性 > 构造方法 > 方法	b
19	不允许方法嵌套超过 3 层逻辑嵌套	b
20	禁止直接在方法中写测试代码	a

三、注释规范

编号	规则内容	级别
21	公共类和方法必须写 Javadoc 注释	a
22	类注释必须描述用途和作者	a
23	注释与代码保持一致，不得过期	a
24	TODO/FIXME 注释必须注明处理人和时间	b
25	不得用注释“注释掉无用代码”并长时间保留	a
26	单行注释前空一格，避免与代码粘连	b
27	多行注释推荐使用 <code>/** */</code> 风格	b

四、异常与日志

编号	规则内容	级别
28	禁止抛出或捕获顶层异常 <code>Exception</code> , <code>Throwable</code>	a
29	必须使用日志记录异常，不得使用 <code>e.printStackTrace()</code>	a
30	日志框架统一使用 SLF4J 接口 + 实现（如 logback）	a

编号	规则内容	级别
31	禁止日志中打印敏感信息（如密码、手机号）	a
32	建议使用占位符 {} 记录日志，不使用字符串拼接	b
33	异常信息应包含上下文信息（如参数、ID）	a
34	捕获异常后应适当处理或向上抛出，不得空 catch	a

五、设计结构规范

编号	规则内容	级别
35	控制器只做参数接收与响应封装，不包含业务逻辑	a
36	Service 层方法不应过长，超过 100 行建议拆分	b
37	工具类应为 final 且私有构造方法防止实例化	a
38	使用常量代替硬编码 Magic Number	a
39	多参数构造函数优先考虑使用 Builder 模式	b
40	禁止循环嵌套中创建对象、执行数据库操作	a

六、集合与并发

编号	规则内容	级别
41	不允许将 null 作为集合元素或键值对的 key	a
42	推荐使用增强 for 循环遍历集合	b
43	HashMap 不可用于多线程环境，使用 ConcurrentHashMap	a
44	禁止使用过时集合类（如 Vector、HashTable）	a

七、数据库与 IO

编号	规则内容	级别
45	禁止在代码中拼接 SQL 字符串，应使用 MyBatis 等框架	a
46	SQL 查询禁止使用 SELECT *，必须指定字段	a
47	数据库操作必须在 DAO 层实现，不得跨层访问	a
48	IO 操作必须使用 try-with-resources 保证释放资源	a

八、安全性与其他

编号	规则内容	级别
49	前端传参需进行合法性校验，后端仍需二次校验	a
50	所有类应加上 @Override 注解（若方法覆盖父类）	b