

# Java 技术规范 (50 条)

## 1. 命名规范

编号	规范内容	等级
1	类名使用 Pascal 命名法, 如 OrderServiceImpl	a
2	方法名、变量名使用 camelCase, 如 getUserById	a
3	常量命名使用全大写加下划线, 如 DEFAULT_TIMEOUT	a
4	接口命名应以 I 开头或 able/er 结尾, 如 IUserService	b
5	枚举类名使用 Pascal 命名法, 枚举值用全大写	b
6	包名全小写, 不使用大写字母或下划线	a
7	Boolean 变量以 is、has、can 开头, 如 isActive	b
8	DTO、VO、BO 等后缀必须清晰标识用途	a
9	不允许使用拼音、数字、简写命名变量	a
10	泛型类使用 T, E, K, V 等通用命名	b

## 2. 代码结构与格式

编号	规范内容	等级
11	每行字符不超过 120, 超过换行	b
12	逻辑相关代码块之间用空行分隔	b
13	每个类最多不超过 1000 行	b
14	每个方法不超过 80 行, 控制逻辑清晰	b
15	每个文件只允许出现一个顶级类	a
16	成员变量必须放在类开头, 方法分组有序	b
17	public 方法在前, private 方法在后	b
18	@Override 必须加在被覆盖方法上	a
19	缩进统一使用 4 个空格, 不使用 Tab	a
20	块注释 (/** */) 用于方法与类, 行注释 (//) 用于逻辑解释	b

## 3. 注释规范

编号	规范内容	等级
21	所有 public 方法必须使用 Javadoc 注释	a
22	注释必须准确、更新及时，不得过期误导	a
23	对于复杂逻辑，必须加注解解释其目的与边界	a
24	使用 TODO/FIXME 必须标明责任人	b
25	不允许大量注释代码长期保留，需清理	a

#### 4. 编程规范

编号	规范内容	等级
26	禁止使用魔法值，应定义为常量	a
27	所有集合必须指定泛型，如 List<String>	a
28	== 不能用于对象比较，应使用 equals()	a
29	字符串比较建议用 "a".equals(str) 防止 NPE	a
30	禁止空的 catch 块，必须处理或记录日志	a
31	异常应具体到类型，不使用 Exception 捕获所有	b
32	禁止在业务代码中使用 System.out.println	a
33	必须使用日志框架如 SLF4J，避免直接依赖 log4j	a
34	switch 语句必须有 default 分支	a
35	所有资源必须在 finally 或 try-with-resource 中关闭	a

#### 5. 控制流程规范

编号	规范内容	等级
36	所有 if/for/while 必须加花括号 {}	a
37	三元表达式嵌套不超过 1 层	b
38	不允许在 if 判断中做赋值操作	a
39	return 不得过多，除非清晰表示不同场景	b
40	避免在循环中使用 break、continue	b

#### 6. 面向对象规范

编号	规范内容	等级
41	类应遵循单一职责原则，不得大而全	a
42	类属性必须 private，通过 getter/setter 访问	a
43	禁止在构造函数中执行复杂逻辑	b
44	禁止在构造函数中调用非 final 方法	a

编号	规范内容	等级
45	Controller 不得包含业务逻辑，仅作分发	a

7. 工具类与框架使用

编号	规范内容	等级
46	工具类必须声明为 final，构造器私有化	a
47	工具类中方法必须为 static，不能依赖上下文	a
48	工具类不得包含业务逻辑，保持纯净	a
49	不允许硬编码依赖框架注入，如 new MapperImpl()	a
50	禁止 Controller 直接依赖 DAO，必须通过 Service 层	a