Java 技术管理文档（编码规范）

# 一、命名规范

1. 【强制】类名使用大驼峰命名法（PascalCase），如 `UserServiceImpl`

2. 【强制】方法名使用小驼峰命名法（camelCase），如 `getUserInfo()`

3. 【强制】变量名使用小驼峰命名法，尽量语义化

4. 【强制】常量全部大写，单词间以下划线分隔，如 `MAX\_LENGTH`

5. 【强制】包名一律小写，使用反域名结构，不使用下划线，如 `com.example.utils`

6. 【强制】枚举类命名使用大驼峰，枚举值全大写，如 `SUCCESS`, `FAILURE`

7. 【推荐】接口命名建议以 `I` 开头或以 `able`、`er` 结尾，如 `IUserService`、`Runnable`

8. 【推荐】boolean 变量建议以 `is`、`has`、`can` 开头，如 `isDeleted`

# 二、代码结构与格式

9. 【强制】每个 Java 文件只能有一个顶级公共类

10. 【强制】花括号不换行，方法定义 `{` 同行放置

11. 【强制】使用 4 个空格进行缩进，不使用 Tab

12. 【强制】所有成员变量声明放在类的顶部

13. 【强制】不允许在一行中声明多个变量，如 `int a = 0, b = 1;`

14. 【推荐】每行代码长度不超过 120 个字符

15. 【推荐】类、方法之间空一行，方法体内逻辑段落之间可适当空行

16. 【推荐】成员方法按照公共 → 保护 → 私有顺序排列

# 三、注释规范

17. 【强制】所有类、方法必须添加 Javadoc 注释

18. 【强制】注释内容要准确描述逻辑，避免“废话注释”

19. 【强制】方法注释包括功能描述、参数、返回值和异常说明

20. 【强制】避免过期注释，应及时删除无用代码

21. 【推荐】使用 `//` 单行注释优先于 `/\* \*/`

22. 【推荐】临时代码应使用 `TODO:` 或 `FIXME:` 标注，并写明负责人

# 四、代码实践规范

23. 【强制】不允许出现魔法值，应定义为常量

24. 【强制】不允许捕获 `Exception` 或 `Throwable`，应精确异常类型

25. 【强制】不允许空 catch 块，至少记录日志

26. 【强制】使用 `equals()` 比较对象，避免用 `==`

27. 【强制】字符串常量应放在前面进行 equals 判断，防止空指针，如 `"ok".equals(status)`

28. 【强制】禁止使用 `System.out.println` 进行日志输出，使用日志框架如 `SLF4J`

29. 【推荐】尽量使用 `final` 修饰不可变对象和方法参数

30. 【推荐】使用 `try-with-resources` 自动关闭资源

31. 【推荐】避免使用嵌套过深的 if-else，应使用早返回策略

32. 【推荐】建议使用 `Optional` 处理可能为空的返回值

33. 【推荐】控制每个方法的长度在 80 行以内

# 五、集合与泛型使用规范

34. 【强制】使用接口类型定义集合变量，如 `List<String> list = new ArrayList<>();`

35. 【强制】使用泛型时，避免原始类型，防止类型转换异常

36. 【推荐】使用 `Map` 时，避免使用 null 作为 key 或 value

37. 【推荐】初始化集合时尽可能指定容量

# 六、线程与并发

38. 【强制】禁止直接使用 `Thread` 创建线程，应使用线程池 `ExecutorService`

39. 【强制】并发场景下共享变量应使用 `volatile` 或同步机制

40. 【强制】使用线程安全的集合类，如 `ConcurrentHashMap` 代替 `HashMap`