**Java技术管理规范**

**级别说明**  
a) **强制**：必须遵守，代码审查中一票否决  
b) **推荐**：团队统一执行，特殊情况需说明  
c) **允许**：根据场景灵活选择

**一、基础规范**

1. **包名全小写** (a)  
   **com.example.project** 禁止大写或驼峰
2. **类/接口名大驼峰** (a)  
   **StudentService** 而非 **studentService**
3. **方法/变量名小驼峰** (a)  
   **calculateScore()** 而非 **CalculateScore()**
4. **常量全大写+下划线** (a)  
   **MAX\_RETRY\_COUNT = 5**
5. **禁用魔法数字** (b)  
   需定义常量：**if (status == ORDER\_PAID)**

**二、编码实践**

1. **使用@Override注解** (a)  
   重写方法必须添加：**@Override public void run(){...}**
2. **StringBuilder拼接字符串** (b)  
   循环内禁止**+**拼接：**StringBuilder sb = new StringBuilder()**
3. **集合声明用接口** (b)  
   **List<String> list = new ArrayList<>()**
4. **泛型需显式声明** (a)  
   禁止原生类型：**List<String>** 而非 **List**
5. **equals()防空指针** (a)  
   **"text".equals(variable)** 而非 **variable.equals("text")**

**三、异常处理**

1. **捕获具体异常** (a)  
   禁止**catch (Exception e)**
2. **异常信息包含上下文** (b)  
   **throw new IOException("Failed to read file: " + fileName)**
3. **finally块释放资源** (a)

java

复制

下载

try (FileInputStream fis = new FileInputStream(file)) {

*// ...*

} *// 自动关闭*

1. **禁止吞没异常** (a)  
   **catch**块至少打印日志：**log.error("Error", e)**

**四、并发安全**

1. **volatile保证可见性** (b)  
   多线程共享变量：**private volatile boolean flag;**
2. **同步代码块最小化** (b)  
   **synchronized**仅包裹必要部分
3. **使用线程池** (b)  
   禁止**new Thread()**：**ExecutorService executor = Executors.newFixedThreadPool(4)**
4. **ConcurrentHashMap替代同步Map** (b)  
   高并发场景优先使用并发集合

**五、资源管理**

1. **IO流显式关闭** (a)  
   必须使用try-with-resources
2. **数据库连接池化** (b)  
   使用HikariCP等连接池
3. **释放资源在finally块** (a)  
   非try-with-resources需手动关闭

**六、代码风格**

1. **缩进=4空格** (a)  
   禁止Tab键
2. **行宽≤100字符** (b)  
   超长需换行对齐
3. **大括号K&R风格** (a)

java

复制

下载

if (condition) { *// 左大括号不换行*

*// ...*

}

1. **无意义的空行≤2行** (c)  
   方法间保留1行空行

**七、OOP原则**

1. **字段必须private** (a)  
   通过getter/setter访问
2. **组合优于继承** (b)  
   慎用**extends**，优先**has-a**关系
3. **接口单一职责** (b)  
   每个接口≤3个方法
4. **避免静态方法** (c)  
   工具类除外（如MathUtils）

**八、测试规范**

1. **测试用例命名规范** (b)  
   **shouldReturnTrueWhenInputValid()**
2. **覆盖率≥70%** (c)  
   核心业务需达80%
3. **Mock外部依赖** (b)  
   使用Mockito模拟数据库/网络调用

**九、安全规范**

1. **SQL参数化查询** (a)  
   禁止拼接SQL：**PreparedStatement**
2. **敏感数据不打印日志** (a)  
   密码、token等禁止**System.out**
3. **文件上传校验类型** (b)  
   检查MIME类型及后缀名

**十、性能优化**

1. **避免频繁装箱拆箱** (b)  
   优先**int**而非**Integer**
2. **集合初始化指定容量** (b)  
   **new ArrayList<>(100)**
3. **正则表达式预编译** (b)  
   **private static final Pattern PATTERN = Pattern.compile("...")**

**十一、文档规范**

1. **类/接口必须JavaDoc** (b)

java

复制

下载

*/\*\**

*\* 学生成绩计算器*

*\* @author YourName*

*\*/*

public class ScoreCalculator { ... }

1. **公共方法注释参数/返回值** (b)

java

复制

下载

*/\*\**

*\* 计算加权平均分*

*\* @param scores 成绩列表*

*\* @param weights 权重列表*

*\* @return 平均分，-1表示输入无效*

*\*/*

public double calculateAverage(List<Double> scores, List<Double> weights)