1. 强制（必须遵循的规范）

1.类名使用大驼峰命名法（PascalCase），如 UserService。

2.方法名、变量名使用小驼峰命名法（camelCase），如 getUserById。

3.常量命名必须全部大写，使用下划线分隔，如 MAX\_RETRY\_COUNT。

4.每个类必须有明确的访问修饰符（如 public / package-private），不可省略。

5.文件名必须与公共类名完全一致，如 UserService.java。

6.缩进必须为4个空格，禁止使用Tab字符。

7.每条语句结束必须加分号。

8.禁止类中出现超过100行未拆分的方法。

9.禁止在代码中出现中文注释以外的中文字符（除特殊需求）。

10.禁止捕获 Exception 顶级异常，应细化异常类型。

11.数据库操作必须使用预编译语句（PreparedStatement）防止 SQL 注入。

12.禁止代码中出现System.out.print，应统一使用日志框架如 SLF4J。

13.所有公开接口必须有注释说明，包含功能、参数、返回值。

14.禁止魔法数字（magic numbers）直接出现在代码中，必须提取为常量。

15.严禁使用==比较字符串，应使用equals方法。

16.严禁在多线程场景下使用非线程安全的类，如SimpleDateFormat。

17.禁止在循环中进行对象创建、数据库连接、文件打开等重操作。

18.严禁方法中使用 return null 返回集合，应返回空集合。

19.禁止在公共类中使用默认包（default package）。

1. 所有枚举类必须显式定义构造函数及属性。
2. 推荐（建议遵循，提高可读性与可维护性）

21.类文件中结构顺序建议为：常量、属性、构造函数、方法。

22.建议每个方法不超过50行代码，超过需重构。

23.推荐使用 Lombok 简化 Getter/Setter，但应谨慎使用 @Data。

24.建议使用 JDK8 以上的 Stream API 提高代码简洁性和可读性。

25.建议控制类文件长度在1000行以内，超过应拆分类职责。

26.推荐使用日志占位符{}而非字符串拼接进行日志打印。

27.建议所有属性使用private修饰并提供getter/setter。

28.推荐 Controller 层返回统一的响应结构（如 Result、ResponseWrapper）。

29.推荐使用Optional替代可能为 null 的返回值，避免 NPE。

30.建议使用 final 修饰不变变量或参数，提高代码可读性。

31.推荐单元测试覆盖率在80%以上，并使用 Mock 工具进行隔离测试。

32.建议为每个模块配套 README 文件说明使用方式与注意事项。

33.建议代码使用英文注释，必要时辅以简要中文说明。

34.推荐使用构建工具（如 Maven/Gradle）管理依赖。

35.建议用 enum 替代常量类中的状态码定义。

36.推荐使用异常类区分系统异常与业务异常（如 BizException）。

37.建议控制类的成员变量不超过7个，超出建议拆分。

38.推荐注释使用 Javadoc 风格，统一格式。

39.建议用 MapStruct 等工具进行对象转换，避免手动拷贝。

1. 推荐引入代码静态分析工具（如 Checkstyle、PMD、SonarQube）保证规范
2. 允许（允许使用但应谨慎）

41.可以使用 var（JDK10+）声明局部变量，但仅限于简洁和可读性不受影响时。

42.可以使用反射技术，但应明确边界，避免过度依赖。

43.可以使用匿名内部类，但优先推荐使用 Lambda 表达式。

44.可使用单例模式，但推荐使用枚举单例（Enum Singleton）实现。

45.可在工具类中使用静态方法，但应保持方法无状态。

46.可以使用 JSON 序列化/反序列化，但需注意字段兼容性和安全性。