# Java 语言规范文档

## 一、命名规范

### a、强制

1. 类名必须使用大驼峰命名法（PascalCase），如 UserService
2. 方法名必须使用小驼峰命名法（camelCase），如 getUserById
3. 常量名必须全部大写，单词间用下划线分隔，如 MAX\_VALUE
4. 变量名必须使用小驼峰命名法，避免使用单字母变量
5. 包名必须全部小写，连续的单词之间无分隔符，如 com.example.project

### b、推荐

1. 接口名推荐使用形容词或动词短语，如 Serializable 或 EventListener
2. 枚举类名推荐使用单数形式，如 Color 而非 Colors
3. 泛型类型参数推荐使用单个大写字母，如 T（Type）、E（Element）、K（Key）、V（Value）

### c、允许

1. 可以使用缩写，但必须是公认的缩写，如 XML、HTML
2. 测试类可以使用 Test 后缀，如 UserServiceTest

## 二、代码格式

### a、强制

1. 每行代码长度不超过120个字符
2. 左大括号不换行，右大括号单独一行
3. 方法参数间的逗号后必须有一个空格
4. 控制语句（if、for、while等）的圆括号与关键字之间必须有一个空格
5. 二元运算符（如 +、-、=）两边必须有一个空格

### b、推荐

1. 推荐使用4个空格缩进，而非Tab
2. 方法之间空一行，不同逻辑块之间空一行
3. 类的成员变量按照字段、构造方法、方法的顺序排列

### c、允许

1. 允许在方法参数较多时进行适当换行，保持格式美观
2. 允许在表达式较长时进行适当换行，保持可读性

## 三、注释规范

### a、强制

1. 类和接口必须有Javadoc注释，说明用途
2. 方法必须有Javadoc注释，说明功能、参数和返回值
3. 常量必须有注释，说明用途和取值范围
4. 复杂的业务逻辑必须有注释说明

### b、推荐

1. 推荐在关键代码段添加注释，解释代码意图
2. 推荐在修改代码时更新相关注释
3. 推荐使用TODO、FIXME等标记未完成或需要修复的代码

### c、允许

1. 允许使用单行注释（//），但必须简洁明了
2. 允许使用块注释（/\* \*/），但不推荐用于代码注释

## 四、编程规约

### a、强制

1. 禁止使用未经检查的类型转换
2. 禁止使用过时的API
3. 禁止在finally块中使用return语句
4. 禁止在循环中创建对象引用
5. 禁止使用==比较包装类型，必须使用equals方法
6. 必须使用try-with-resources语句处理资源
7. 必须对敏感信息进行加密存储
8. 必须处理异常，不能简单地捕获后不做处理

### b、推荐

1. 推荐使用接口而非具体实现类型声明变量
2. 推荐使用不可变对象
3. 推荐使用Stream API处理集合操作
4. 推荐使用Optional类处理可能为空的值
5. 推荐使用构造器模式创建复杂对象

### c、允许

1. 允许使用静态导入，但必须谨慎使用
2. 允许使用Lambda表达式简化代码
3. 允许使用var关键字声明局部变量，但必须确保类型明确

## 五、异常处理

### a、强制

1. 必须捕获并处理受检异常
2. 必须在方法签名中声明未处理的受检异常
3. 必须记录异常信息，包括堆栈跟踪

### b、推荐

1. 推荐使用自定义异常类表示特定业务异常
2. 推荐在finally块中释放资源

### c、允许

1. 允许在测试代码中使用更宽松的异常处理

## 六、单元测试

### a、强制

1. 必须为公共方法编写单元测试
2. 必须测试边界条件和异常情况

### b、推荐

1. 推荐使用测试框架如JUnit和Mockito
2. 推荐编写集成测试验证系统功能

### c、允许

1. 允许使用数据驱动测试处理多种输入场景

## 七、性能优化

### a、强制

1. 必须避免在循环中使用字符串拼接，应使用StringBuilder
2. 必须避免创建不必要的对象

### b、推荐

1. 推荐使用局部变量而非实例变量
2. 推荐使用缓存机制提高性能

### c、允许

1. 允许使用并行流处理大数据集
2. 允许在性能敏感的代码中使用原始类型而非包装类型

## 八、安全规范

### a、强制

1. 必须对用户输入进行验证和过滤
2. 必须使用安全的随机数生成器
3. 必须避免SQL注入和XSS攻击

### b、推荐

1. 推荐使用HTTPS协议传输敏感数据
2. 推荐使用安全的密码存储方式

### c、允许

1. 允许在非生产环境中使用较宽松的安全策略