

Java 编程军规

- 1 public方法参数的合法性检查应由方法本身负责，每个public方法必须保证自己的健壮性。
- 2 调用方法获取返回结果后必须进行有效性校验，以确保后续代码在运行过程中不会抛出异常或产生逻辑错误。
- 3 在进行数据库操作或IO操作时，必须确保资源在使用完毕后得到释放，并且必须确保释放操作在finally中进行。
- 4对于if...else if...(后续可能有多个else if...)这种类型的条件判断，最后必须包含一个else分支，避免出现分支遗漏，造成错误；每个switch-case语句都必须保证有default，避免出现分支遗漏，造成错误。
- 5 在使用 Timer 或者 ScheduledThreadPoolExecutor 执行周期性任务时，实现Runnable接口必须在方法内捕获异常，避免因异常抛出导致周期性任务失效，后续不会继续执行。
- 6 确定需要覆盖对象equals()方法时，必须同时覆盖hashCode()方法。
- 7 在进行精确计算时(例如：货币计算)避免使用float和double,浮点数计算都是不精确的，必须使用BigDecimal或者将浮点数运算转换为整型运算。
- 8 确保程序不再持有无用对象的引用，避免程序内存泄露。
- 9 对象比较必须使用equals方法，而不是“==”。
- 10 访问数组、List等容器内的元素时，必须首先检查下标是否越界，以杜绝下标越界异常的发生。
- 11 将对象存入HashSet，或作为key存入HashMap（或HashTable）后，必须确保该对象的hashCode值不变，避免因hashCode值的变化导致不能从容器内删除该对象，进而引起内存泄露的问题。
- 12 在编码过程中，必须在适当的位置，以适当的级别打印日志，便于代码出现问题后进行定位分析。例如：方法入口，出口以Debug打印；异常分支使用ERROR,关键参数打印使用INFO。
- 13 避免在程序中使用魔鬼数字，必须用有意义的常量来标识。
- 14 将字符串转换为数字时必须处理NumberFormatException异常。
- 15 在进行三个字符串（不包含三个）以上的串联操作时必须使用StringBuilder或StringBuffer，禁止使用“+”。
- 16 根据应用场景选择最适合的容器，避免因容器选择不当造成程序性能问题。

17 必须在进行I/O操作时使用缓存。

18 在程序中必须考虑对象重用，避免创建不必要的垃圾对象。

19 对多线程访问的变量、方法，必须加锁保护，避免出现多线程并发访问引起的问题。

20 新起一个线程时，都要使用Thread.setName(...)设置线程名。

21 线程使用时，要在代码框架中使用线程池，避免创建不可复用的线程。禁止在循环中创建新线程，否则会引起JVM资源耗尽。