软件工程

OOP 面向对象编程（Object Oriented Programming，OOP，面向对象程序设计）是一种计算机编程架构。OOP 的一条基本原则是计算机程序是由单个能够起到子程序作用的单元或对象组合而成。

面向对象编程(Object-Oreinted Programming) 是一种编程范式。指在设计程序时大量运用类实例对象的方式。OOP一旦在项目中被运用，就成了时刻要考虑的东西。

SOA 面向服务的架构（SOA）是一个组件模型，它将应用程序的不同功能单元（称为服务）通过这些服务之间定义良好的接口和契约联系起来。

面向服务架构(Service-Oreinted Architecture) 是将软件设计成一组可互操作的服务的一套原则或方法论。通常在考虑系统架构时才会触及SOA。

基于组件开发(Component-Based Development) 是一种软件工程实践，设计时通常要求组件之间高内聚，松耦合。其接口可能是OO的，调用方式可能是以Service的方式。基于组件开发关注系统层次、子系统边界和子系统间通讯的的设计，处于代码层面但不像OOP的一样是时刻需要运用的东西。

三者身处软件开发的不同层面，因此说他们用于“哪些领域”并不恰当。不论是哪个领域的软件开发，都可能要同时面对OOP、SOA和CBD。

编程思维

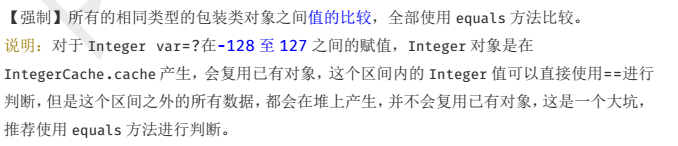
1抽象化 封装 ->降低程序复杂度

2 产品化 复用 回馈完善

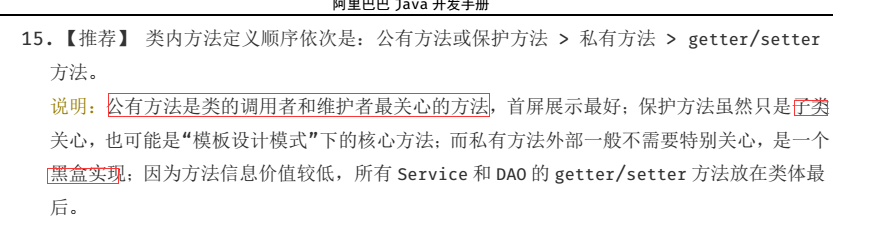
3 设计模式 适配器模式，插件式低耦合

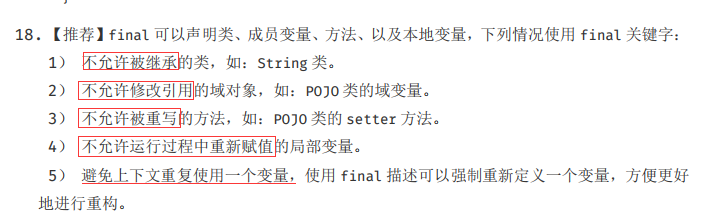
Java编程规范笔记

1不要用复杂的逻辑判断 增加可读性

2

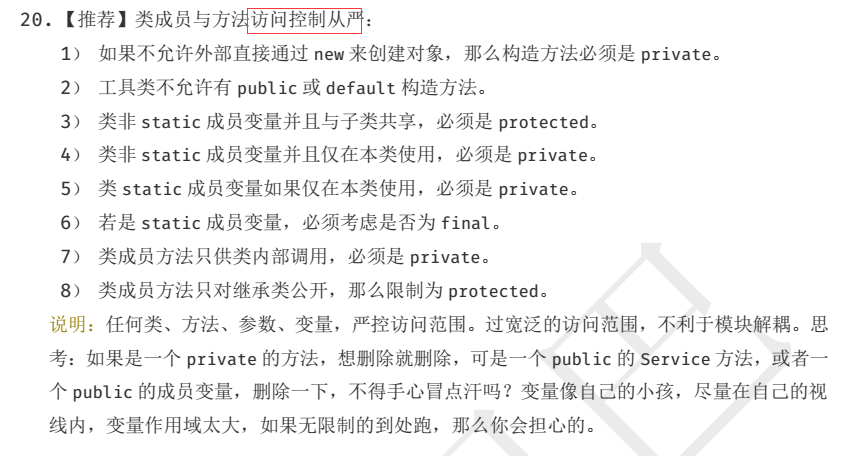
3

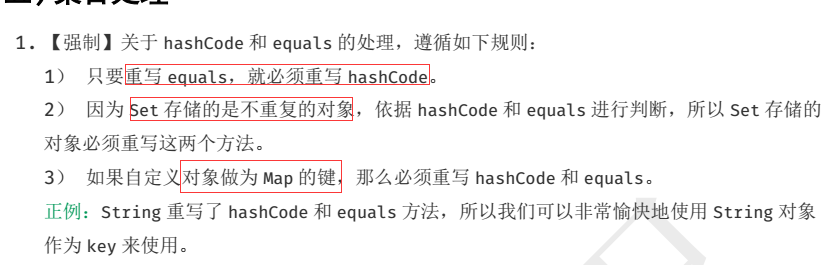
4

5

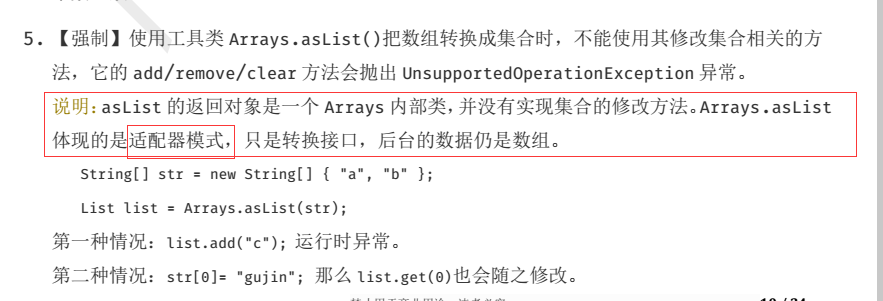
6

public protected private

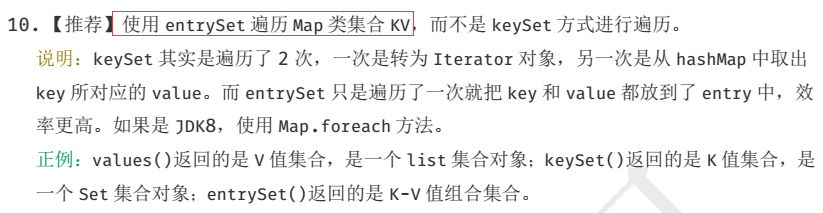


7

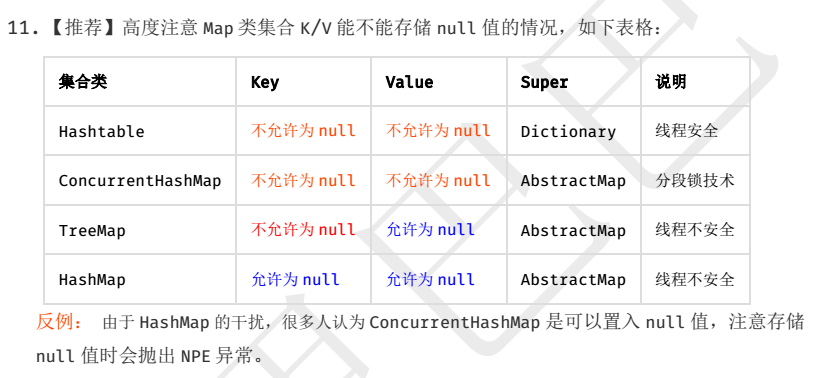
8



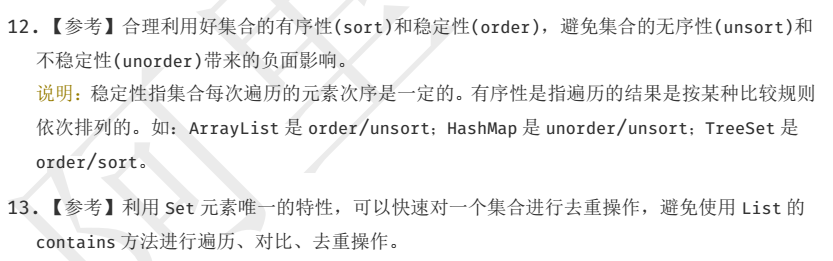
9



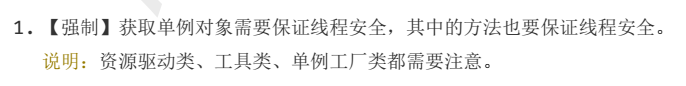
10



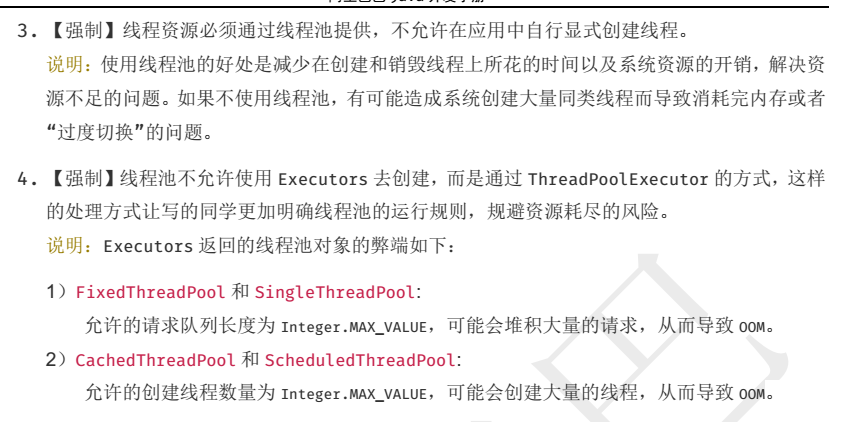
11

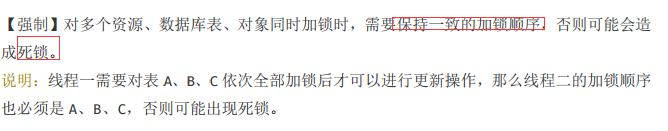
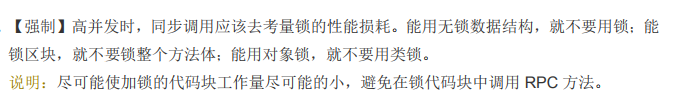


12

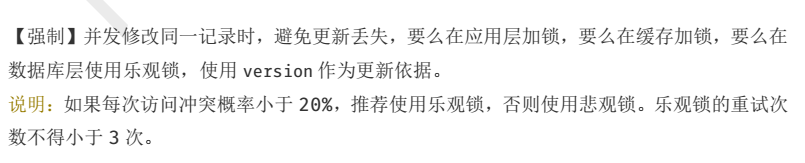


13

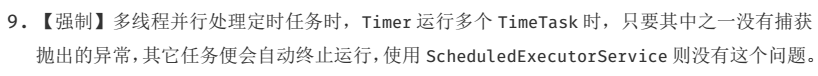


14

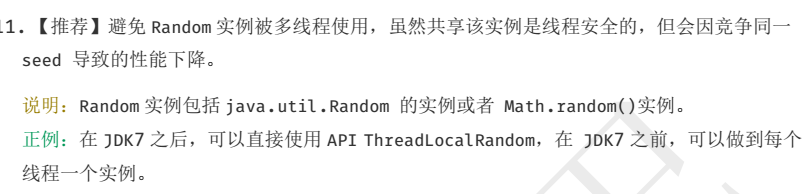
15



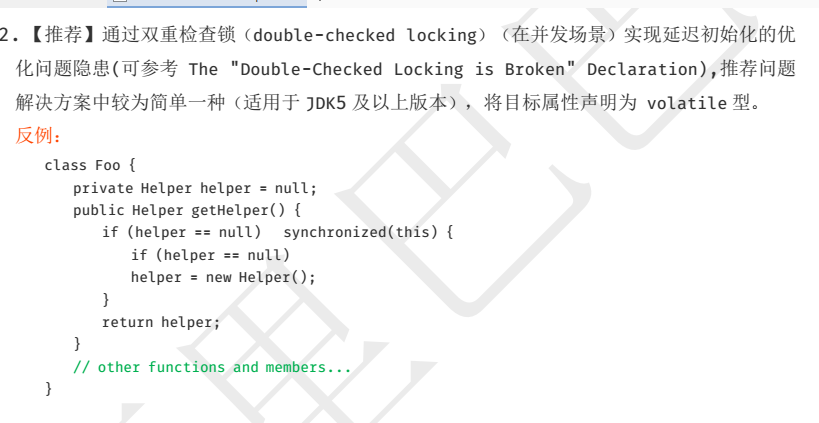
16



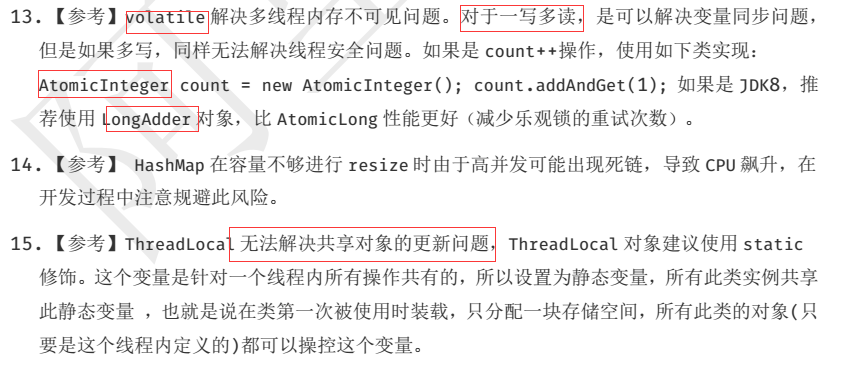
17



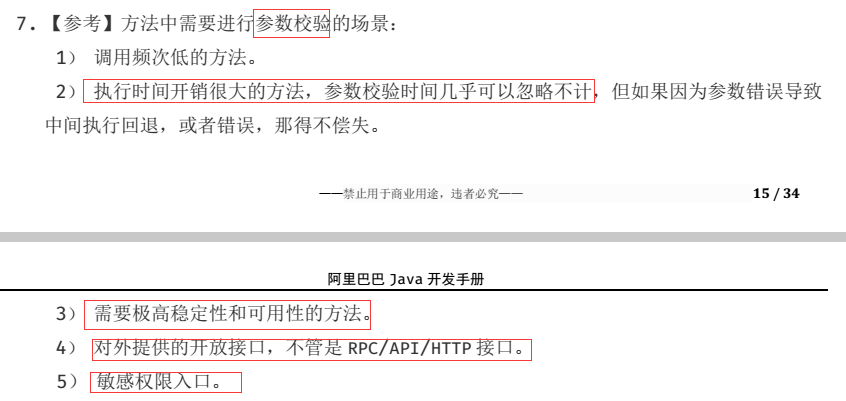
18



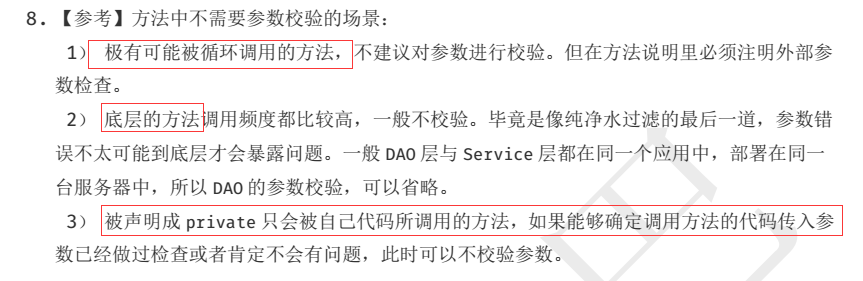
19



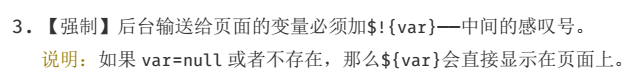
20



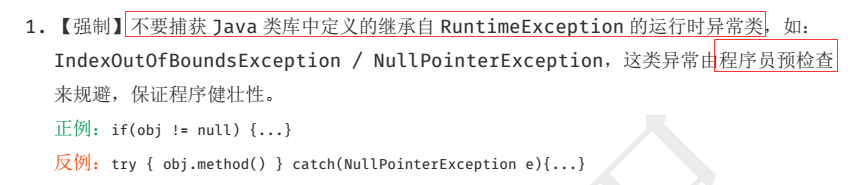
21

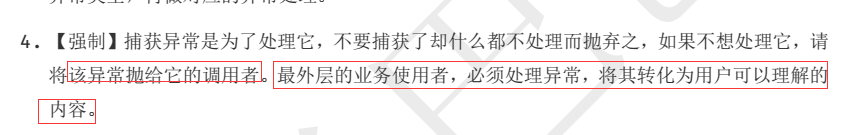


22

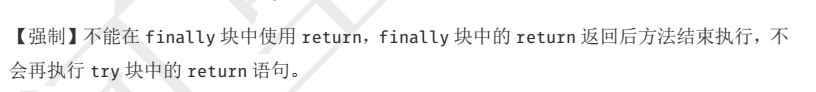


23

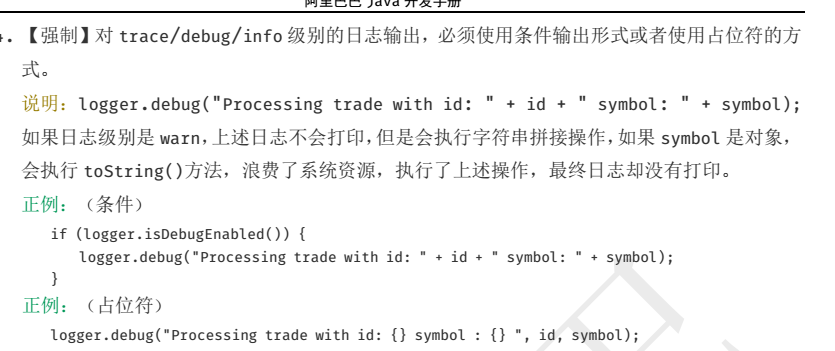


24

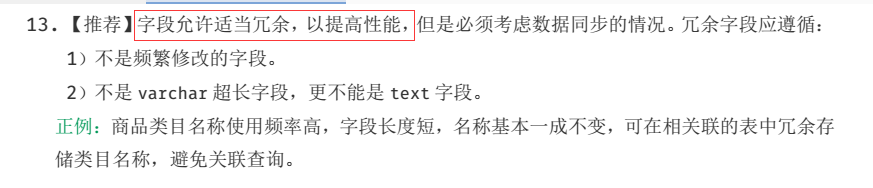
25



26



27



28

