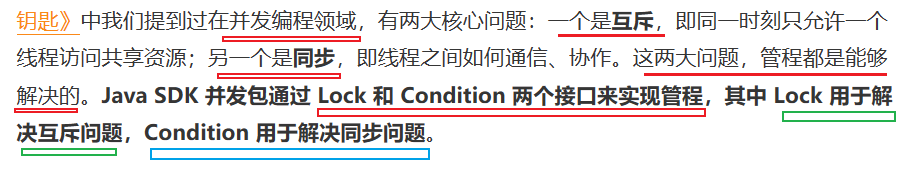
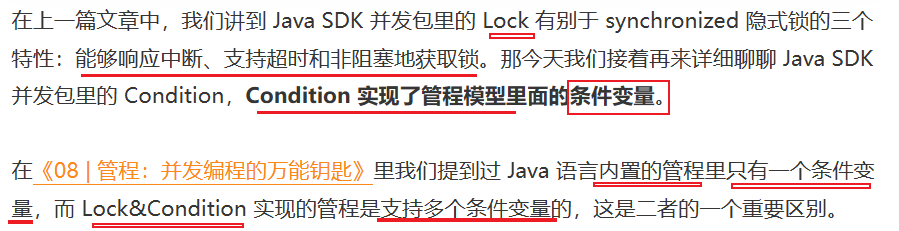
# Lock（上）

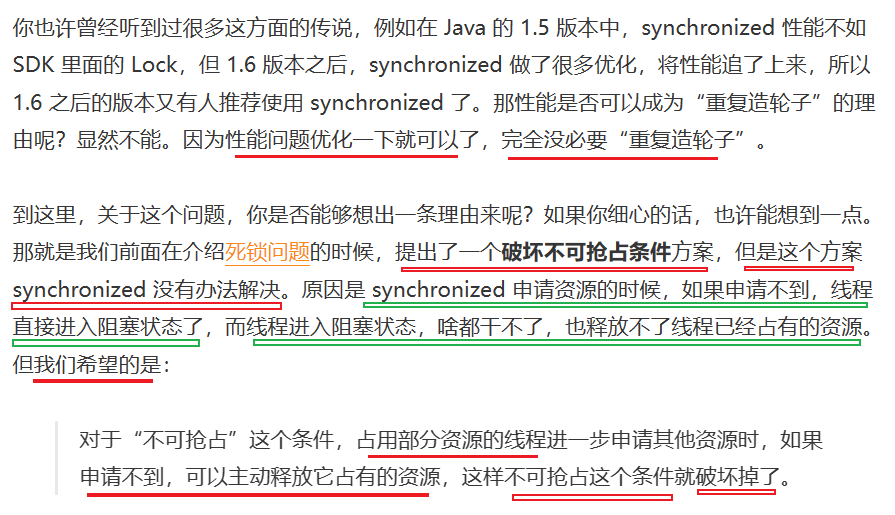
## Lock&Condition与synchronized的对比

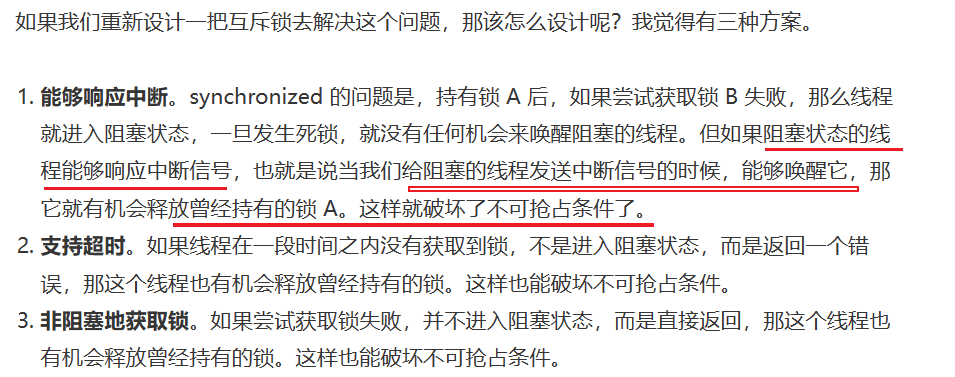
并发包其实就是管程的另一种实现方式





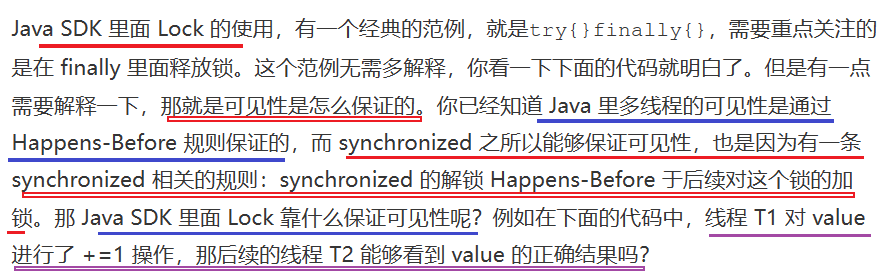
## 再造管程的理由

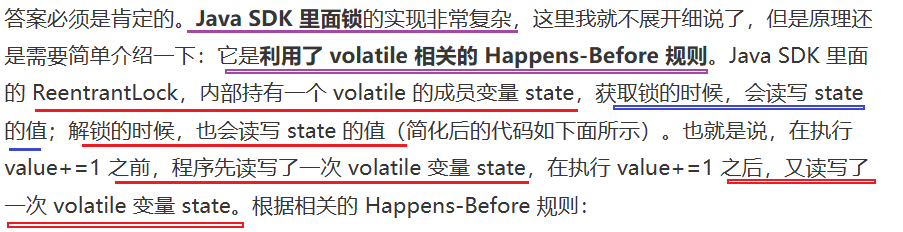


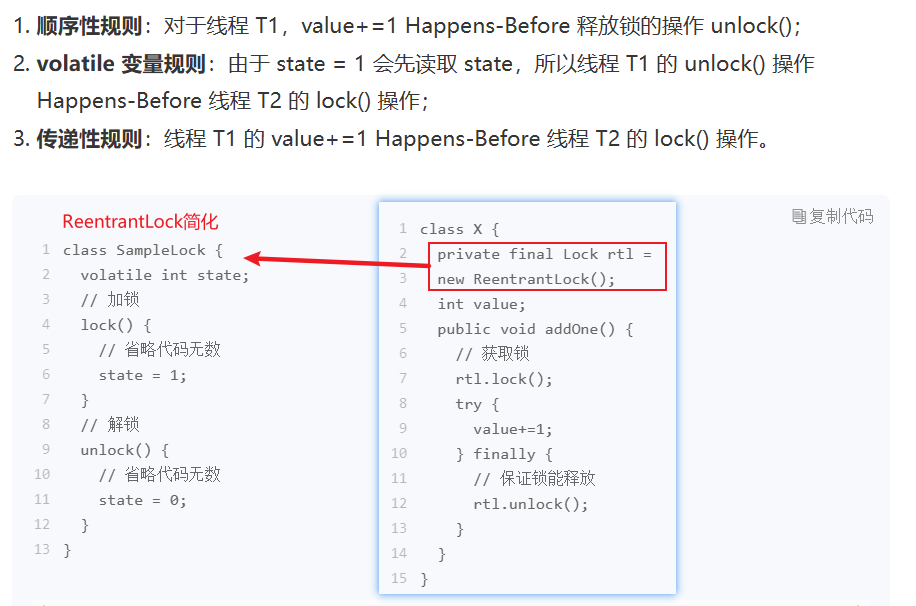




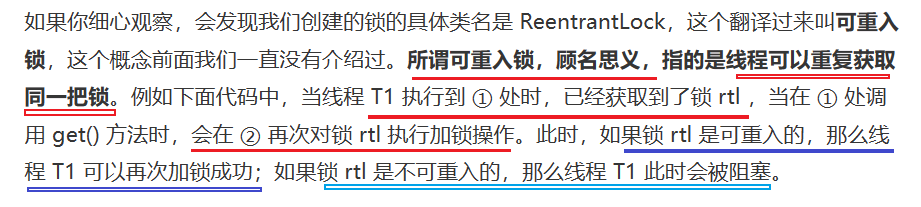
## 如何保证可见性

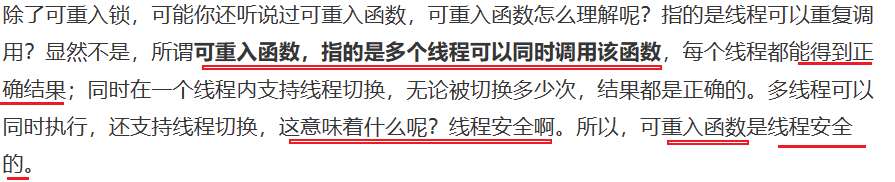
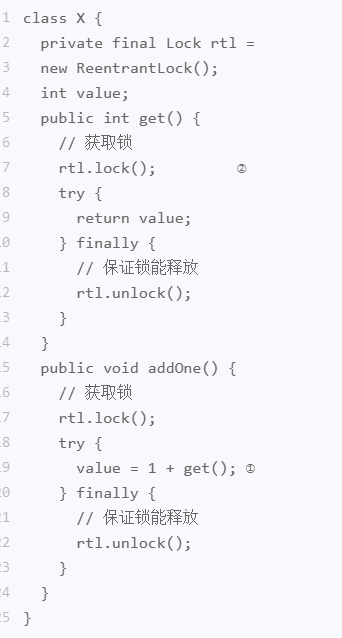




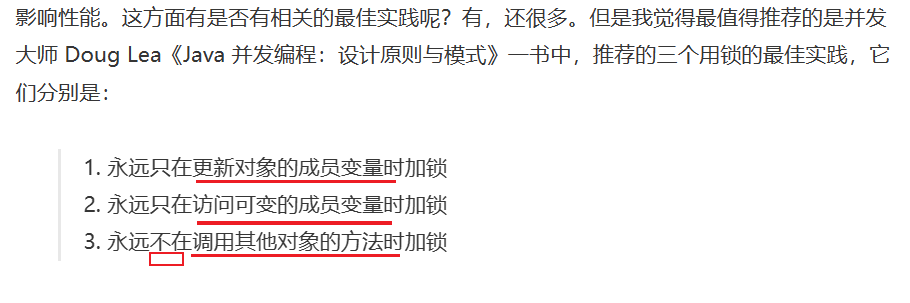


## 什么是可重入锁





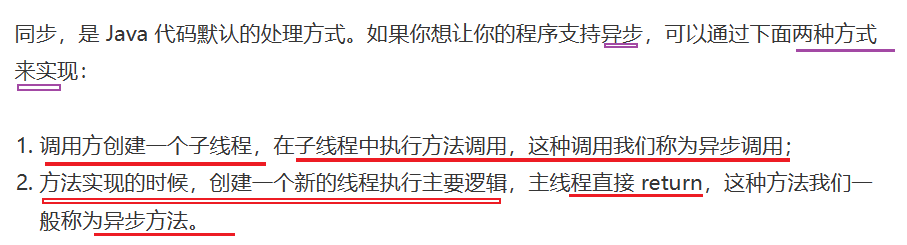
## 实践准则



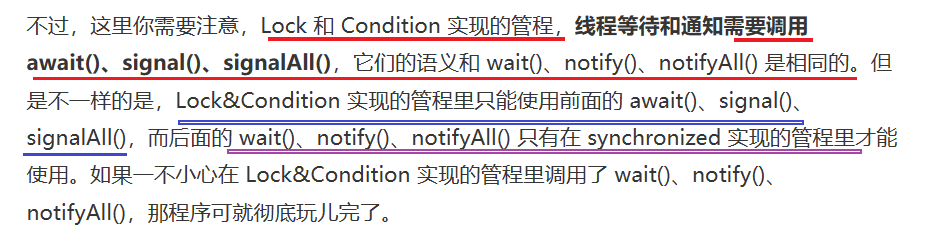
# Condition（下）

## 同步与异步





## Lock&Condition总结

 **所谓可重入锁，顾名思义，指的是线程可以重复获取同一把锁**。

Lock&Condition 是管程的一种实现，所以能否用好 Lock 和 Condition 要看你对管程模型理解得怎么样。

[《Java 并发编程的艺术》](time://mall?url=https%3A%2F%2Fh5.youzan.com%2Fv2%2Fgoods%2F35z7jjvd4r4oo)一书的第 5 章《Java 中的锁》，里面详细介绍了实现原理，我觉得写得非常好。

Lock&Condition 实现的管程相对于 synchronized 实现的管程来说更加灵活、功能也更丰富。