

1. AQS：AbstractQueuedSynchronizer抽象同步队列

作用：实现Lock、ReentrantLock、Semaphore、CountDownLatch等的基础

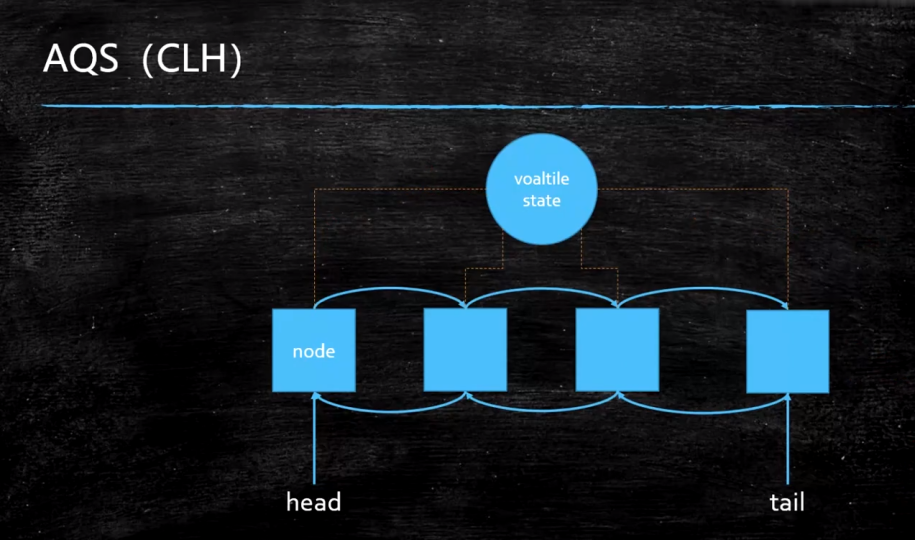
1. AQS原理：CAS（改变state、线程入队）+volatile（修饰state）

Volatile（state）+线程队列（CLH，双向链表）

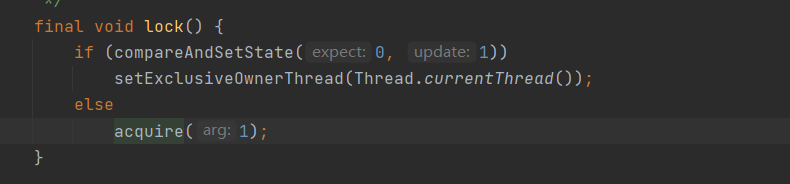
有个线程去访问state（1为持有锁、0为未持有锁）

非公平锁：竞争得到，就直接使用；竞争不到，就在队列里面等待

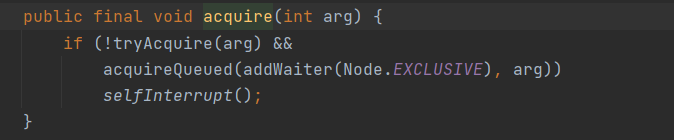
公平锁：到等待队列等待



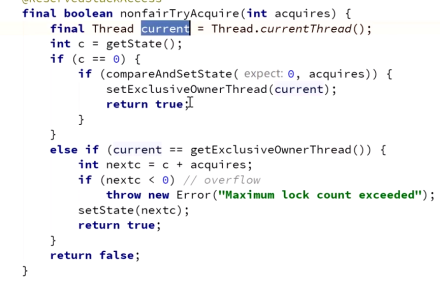
lock：获取锁



acquire:如果获取得到直接用（tryAcquire），否则需要加入线程队列（acquireQueued）



非公平锁：



只有一个线程能把state从0改为1（CAS操作），就代表这个线程拥有这把锁

1. 等待所有线程结束

