**java的原子性**

　　原子性：即一个操作或者多个操作 要么全部执行并且执行的过程不会被任何因素打断，要么就都不执行。

原子性就是同步的问题：比如：

一个很经典的例子就是银行账户转账问题：

　　比如从账户A向账户B转1000元，那么必然包括2个操作：从账户A减去1000元，往账户B加上1000元。

　　试想一下，如果这2个操作不具备原子性，会造成什么样的后果。假如从账户A减去1000元之后，操作突然中止。然后又从B取出了500元，取出500元之后，再执行 往账户B加上1000元 的操作。这样就会导致账户A虽然减去了1000元，但是账户B没有收到这个转过来的1000元。

所以这2个操作必须要具备原子性才能保证不出现一些意外的问题。

java实现原子性的方法有通过synchronized和Lock，他保证了任何时刻都只能有一个线程去执行这个代码块。 这样的不好处是影响性能，因为这样会阻塞其他线程运行。

Java:

保证原子性：synchronized和Lock

可见性：volatile synchronized和Lock

阻塞同步和非阻塞同步