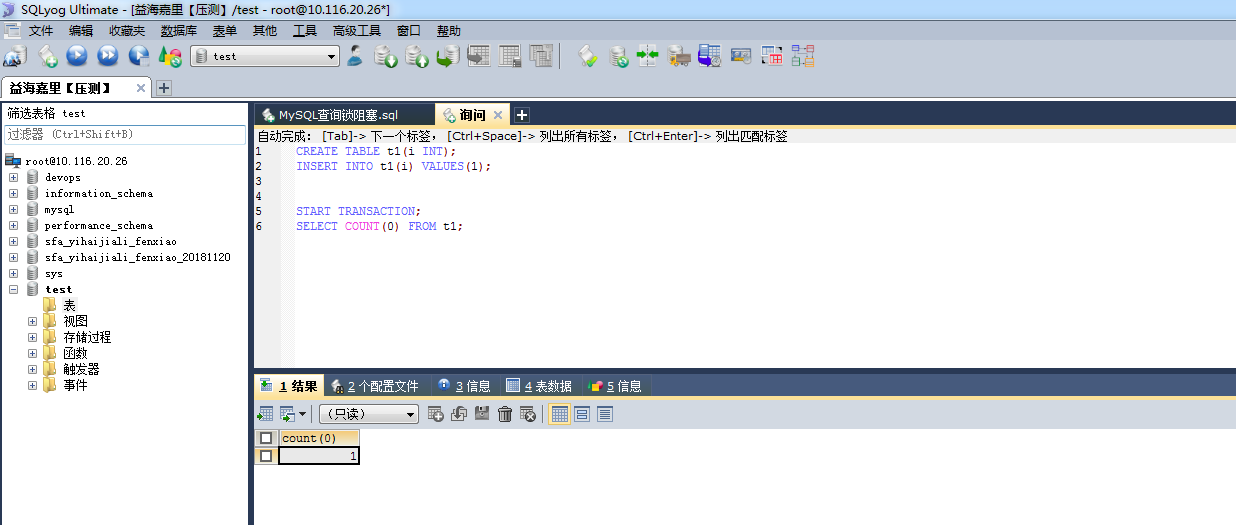
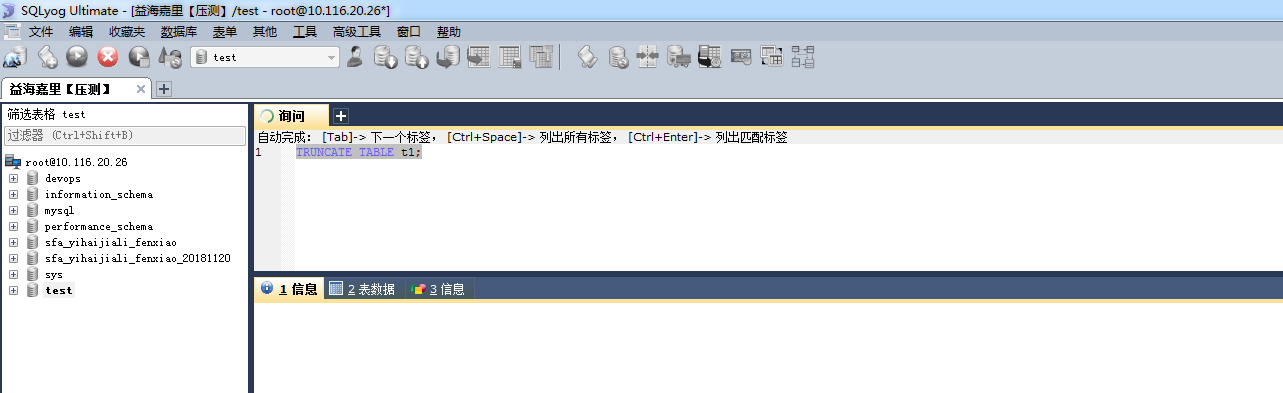
1. **案例一（阻塞）**

会话1：

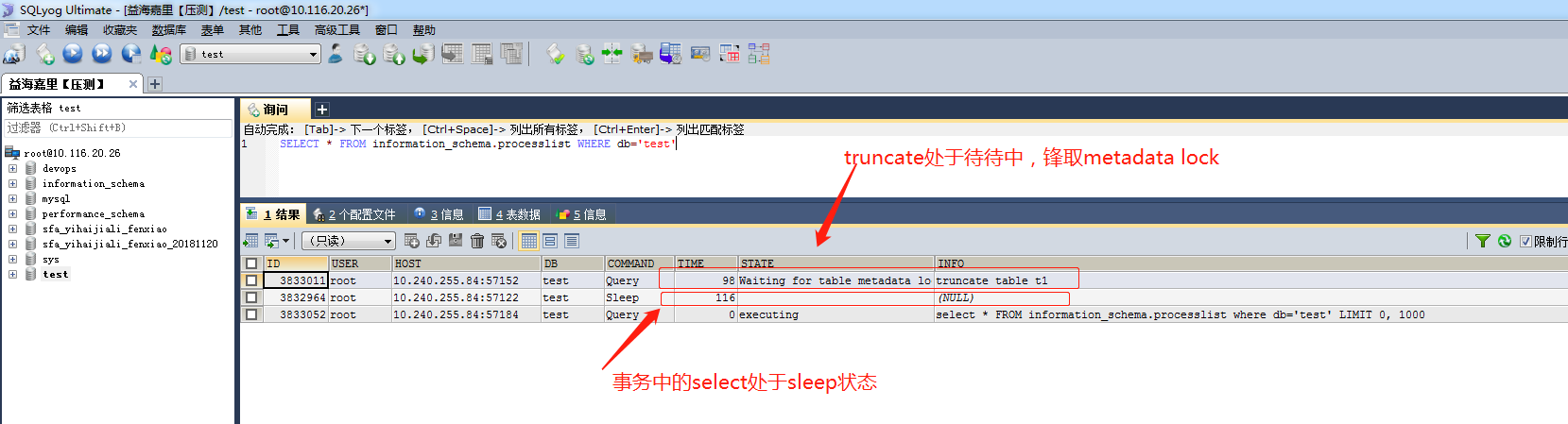


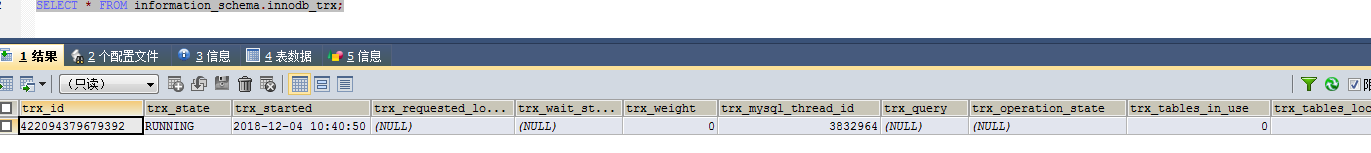
test库新建一张表，只有一行数。把一个select放在事务中

会话2：



另一个会话执行truncate操作，处于等待中





SELECT \* FROM information\_schema.processlist WHERE db='test'  
state=’Waiting for table metadata lock’

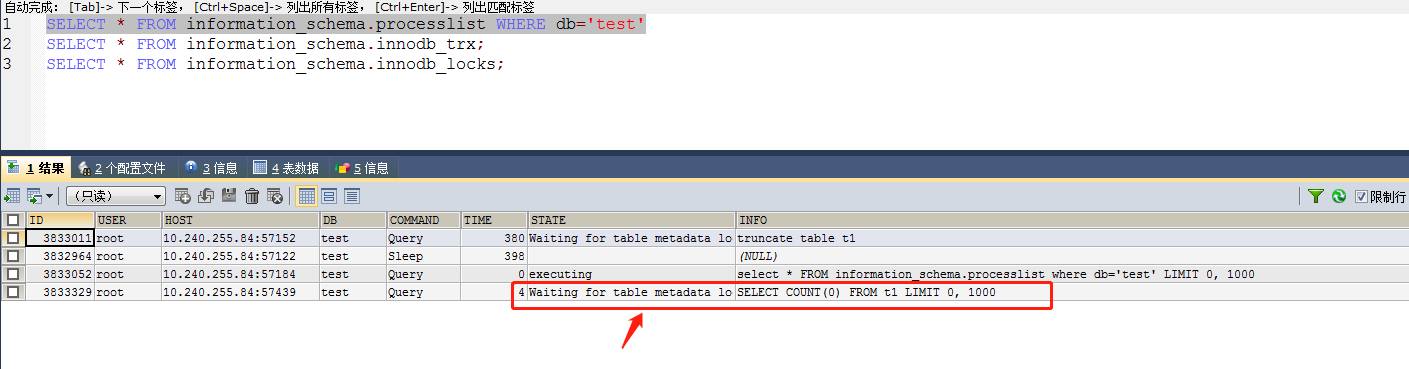
在锁定表中看不到元数据锁的信息

所以我的阻塞监控没有查询到数据

得想个办法监控这一类元数据锁

会话三：

此时再开一个会话进行查询操作（事务中）



新的查询也被阻塞了，因为事务中的查询也要获取元数据锁

最初的源头就是有查询执行完了，没有提交或回滚

应该结合两部分还监控这一类事件

造成这种问题的，一定在事务表中有数据，虽然看不到具体SQL

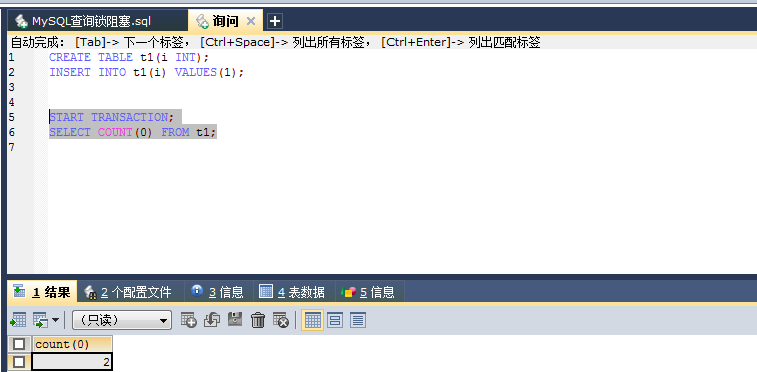
同时在processlist表中的state='Waiting for table metadata lock'这一类的状态

这一类的全部是处理等待中，而事务表中是处理锁定中

所以我觉得得加一个大事务的监控，发现长事务就得分析原因，只要没有长事务，这一类问题就能解决。

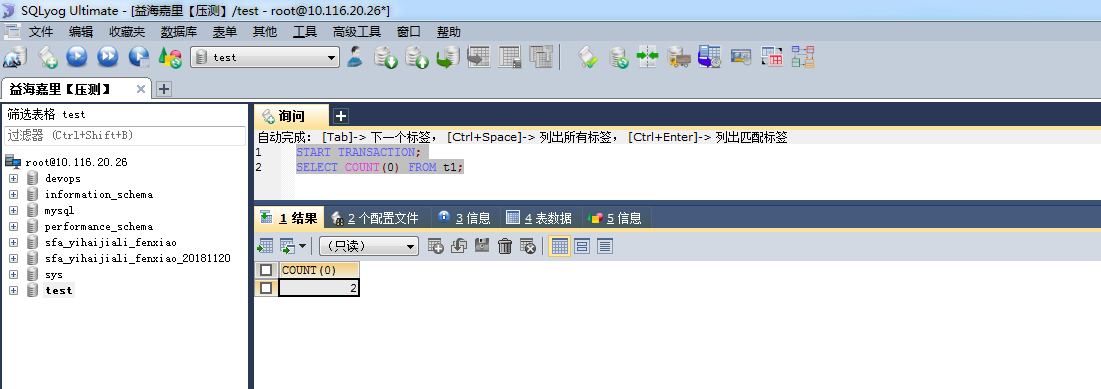
1. **案例二（无阻塞）**

刚才的例子是元数据锁有阻塞的情况，现在再看一个可以同时获取元数据锁的情况，没有阻塞的情况。

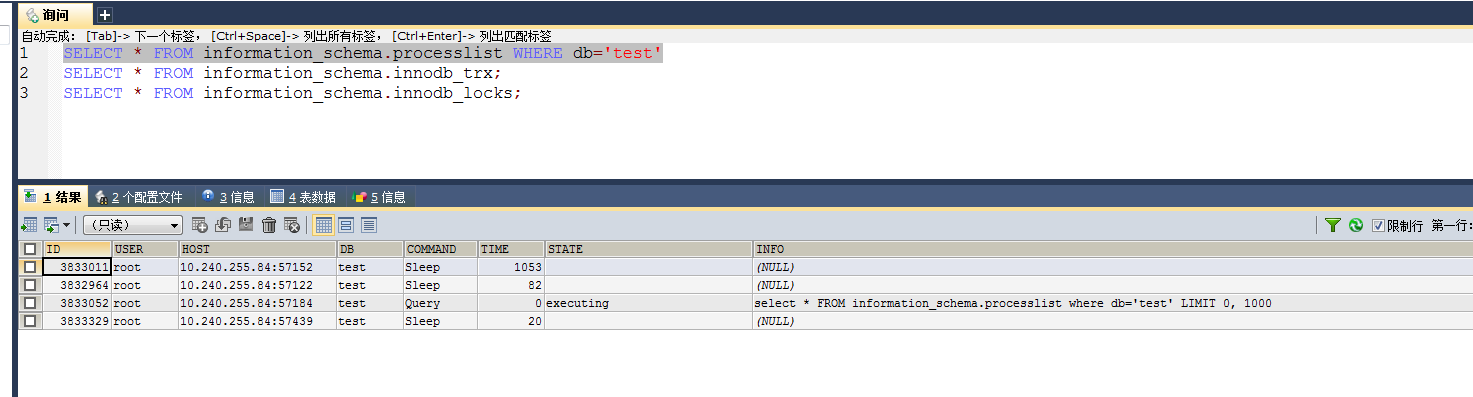


会话1，还是在事务中查询一张表数据

再开一个会话，也在事务中，也查询这张表



此时没有阻塞的情况

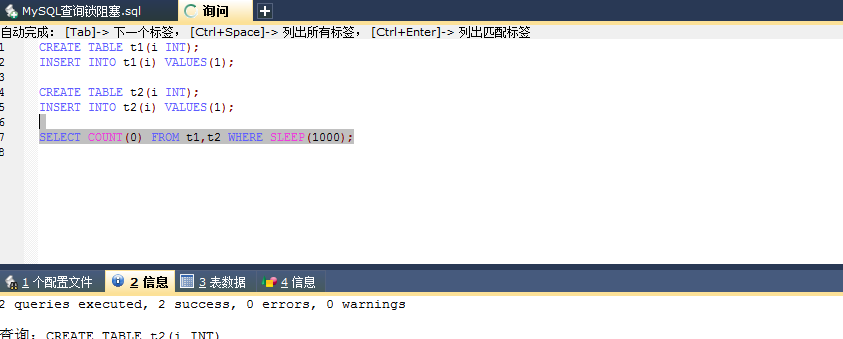


因为元数据锁有读元数据锁，写元数据锁

读元数据锁，可以共存，没有冲突，写元数据锁与读有冲突。

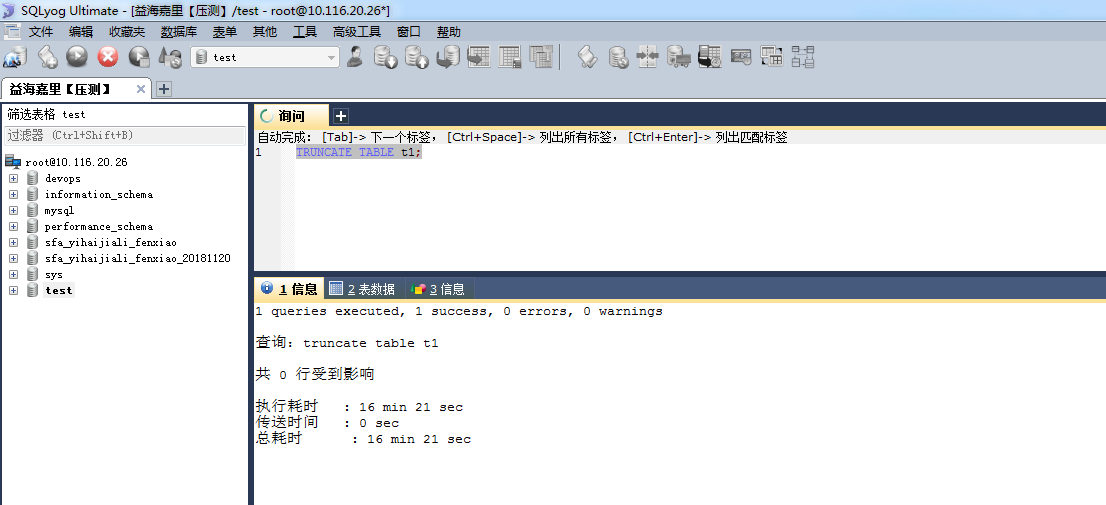
1. **案例三（阻塞DDL）**

非事务，正在进行中的查询，所有相关表上加的也是只读元数据锁

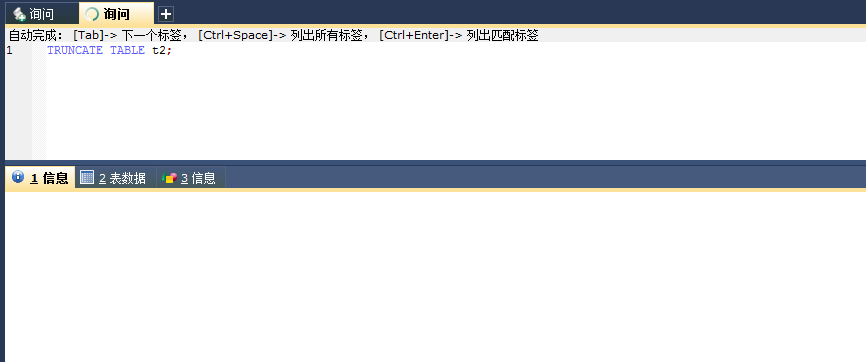


现在是两张表,各有一条数据,我模拟一个慢查询

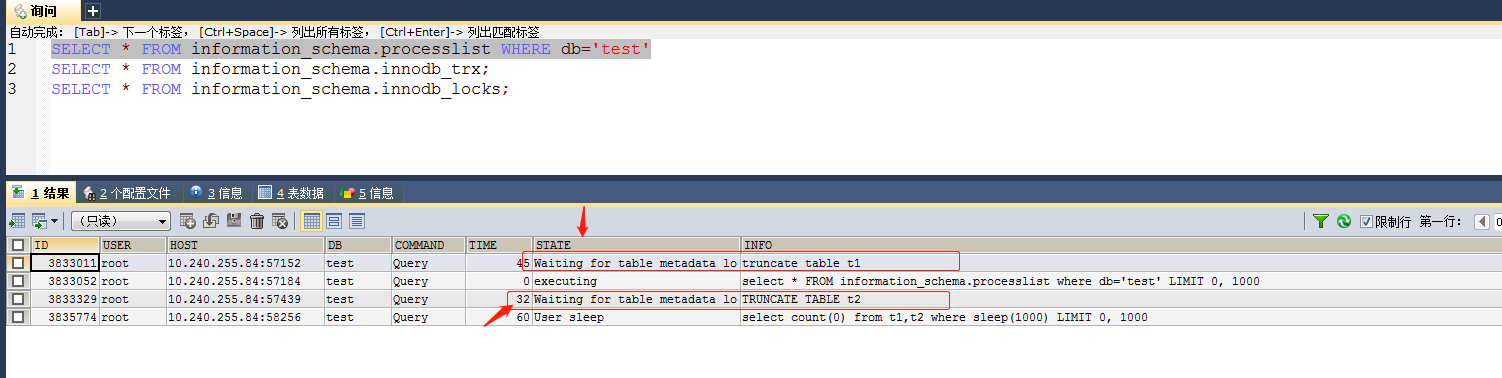
会话2：



会话3：

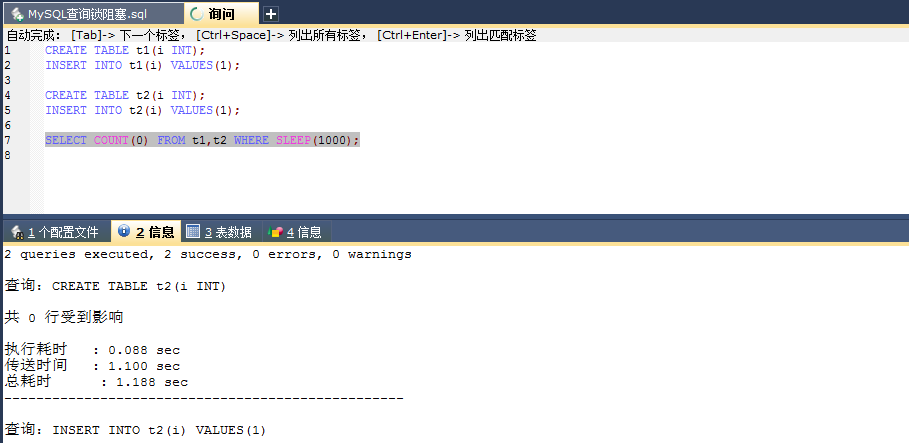


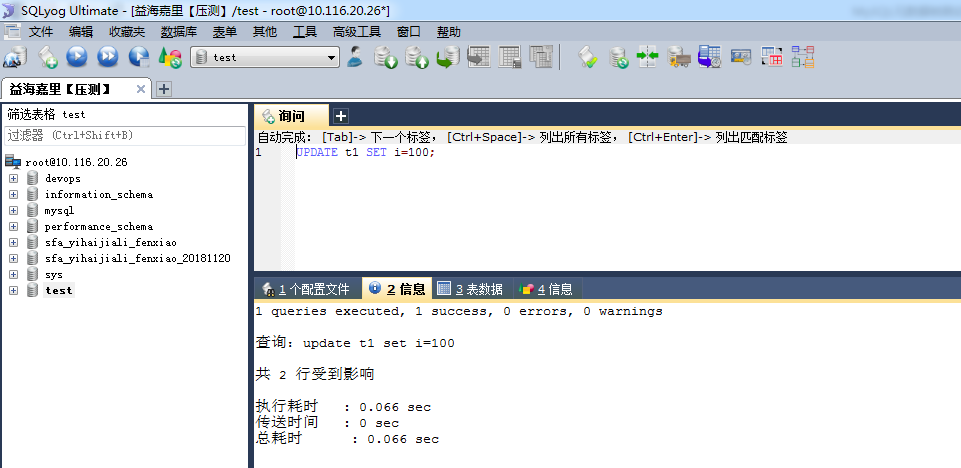
查询会话等待：

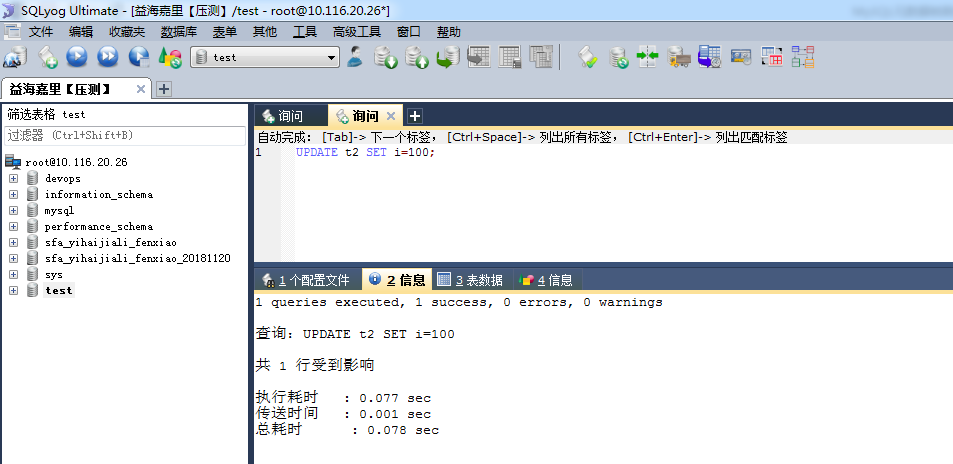


1. **案例三（无阻塞DML）**

非事务，正在进行DML操作，元论更新操作是否在事务中，所有相关表上可以进行DML操作均可正常进行。







1. **总结**

第一个案例，第一个会话首先获取的是读元数据锁，第二个会话试图获取写元数据锁时发生阻塞，后面的会话再获取读元数据锁也有阻塞，因为写与读是冲突的。

第二个案例，第一个会话首先获取的是读元数据锁，第二个会话也获取的是读元数据锁，因此没有冲突

第三个案例：正在进行中的查询，所有表上都会加只读元数据锁，不会影响这些表上的查询，但会阻塞DDL操作及DML操作。