《数据库系统实验》

实验报告

|  |  |
| --- | --- |
| **题目** | （实验12） |
| **姓名** | 陈欣宇 |
| **学号** | 21307347 |
| **班级** | 人工智能与大数据 |

1. 实验环境：

1、操作系统：Windows 10

2、DBMS：mysql 8.0.31

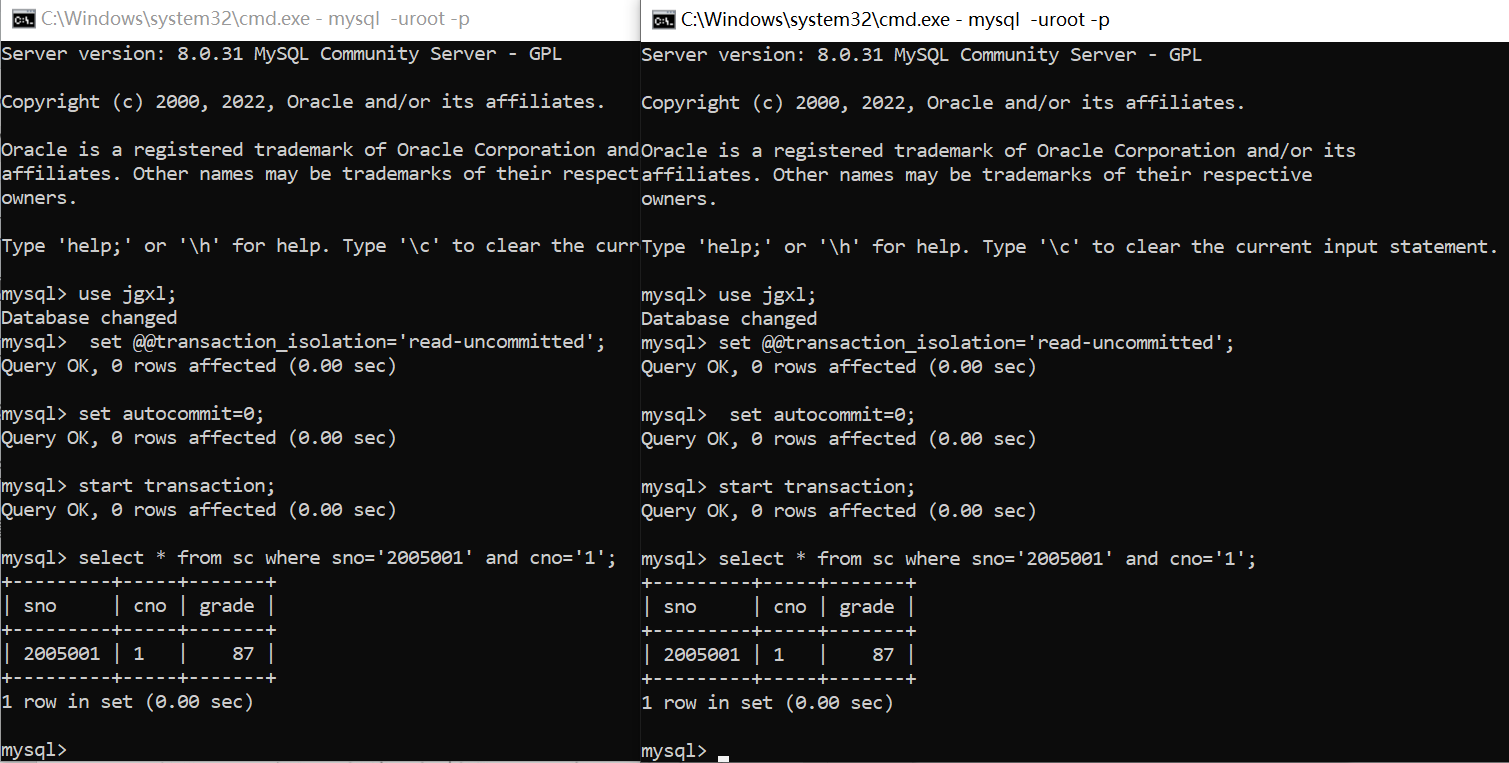
1. 实验内容与完成情况：

**2.1 重复表12-5**

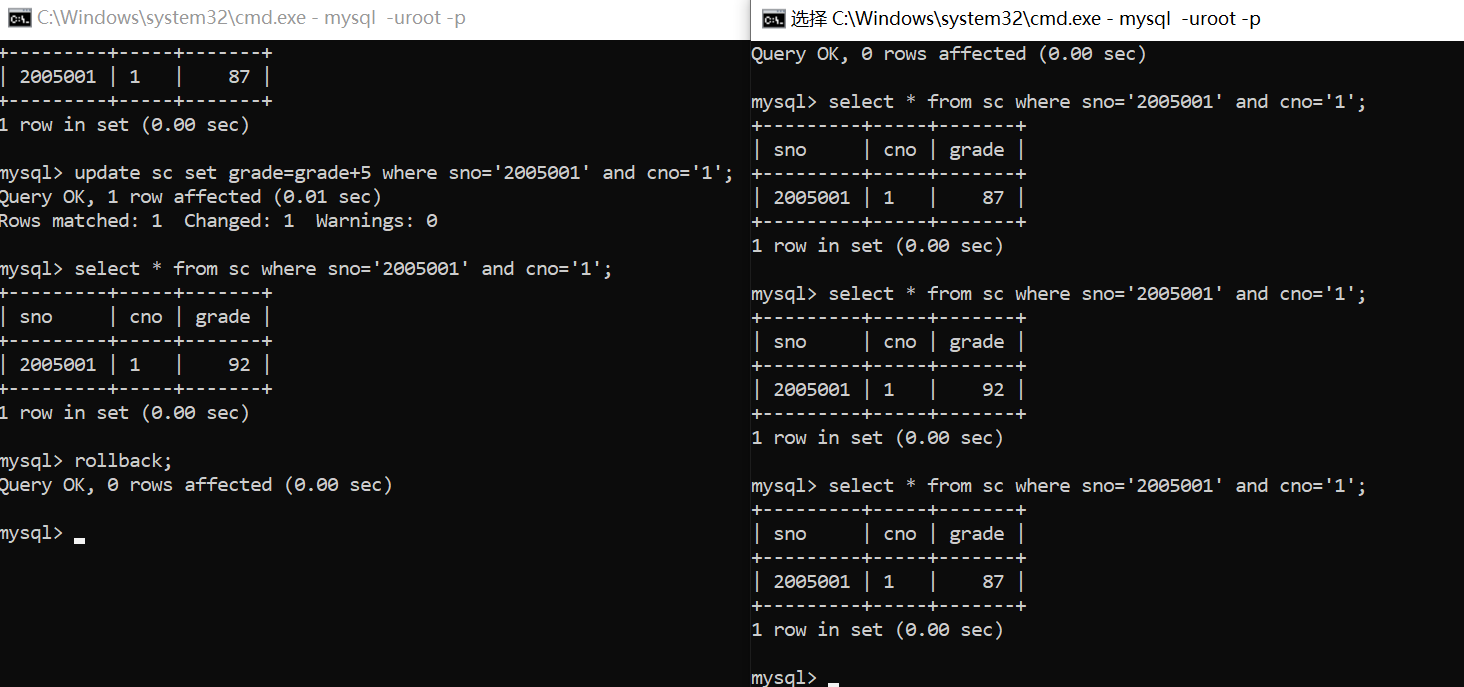
打开两个cmd窗口连接mysql，use jxgl调用jgxl数据库，分别重复表12-5命令

注意：mysql数据库的更新，在旧版本中tx\_isolation是作为transaction\_isolation的别名被应用

执行到update之前，select显示如下：



执行update，进行select显示都为grade=92，在session\_1执行rollback后session\_2也被rollback，可看到select结果倒退回grade=87

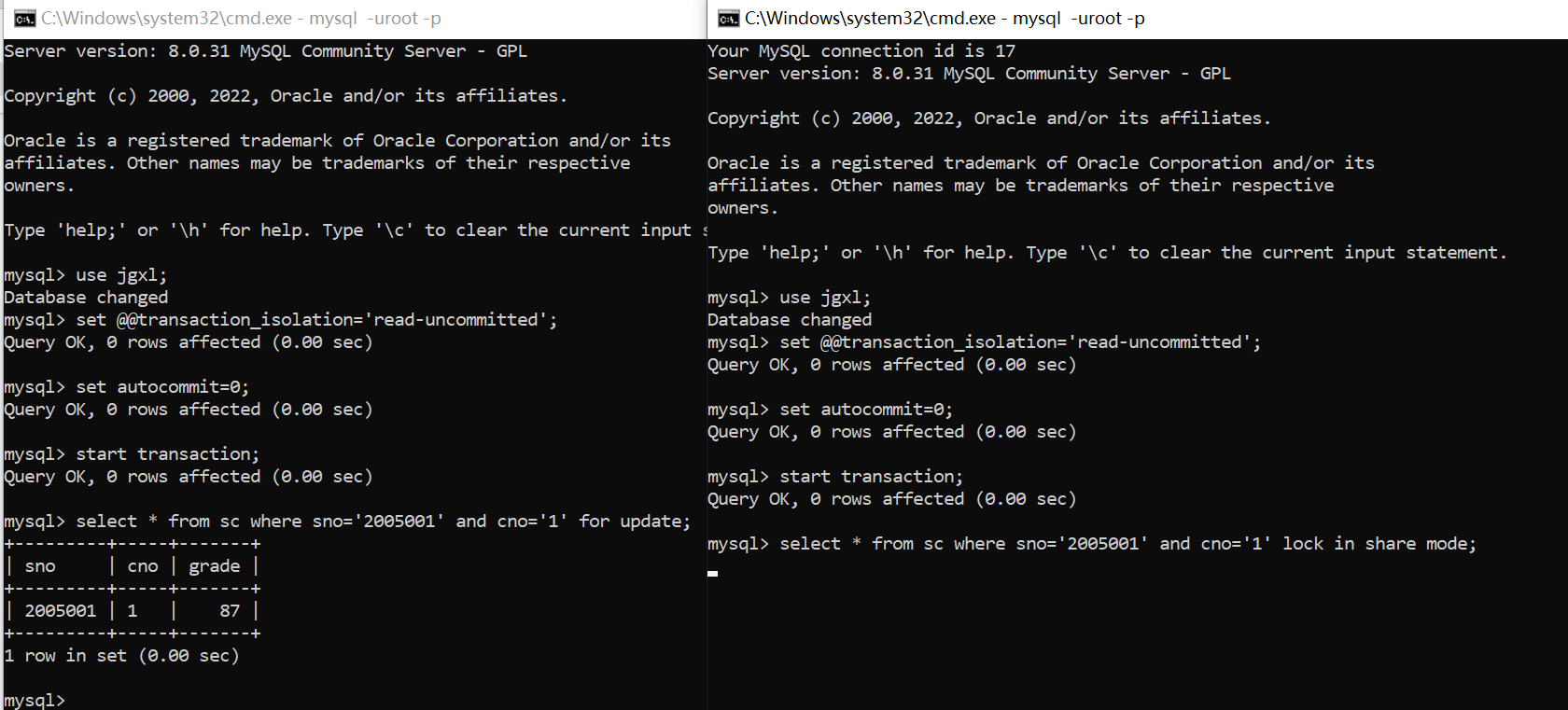


隔离级别为未提交读（read-uncommitted）时会产生脏读，即在不同事务下，可以督导另一个事务未提交的数据，违反了隔离性。表12-5中操作session的隔离剂相同，且未设置锁，导致session\_2一直能够督导session\_1修改或rollback但未提交的数据。

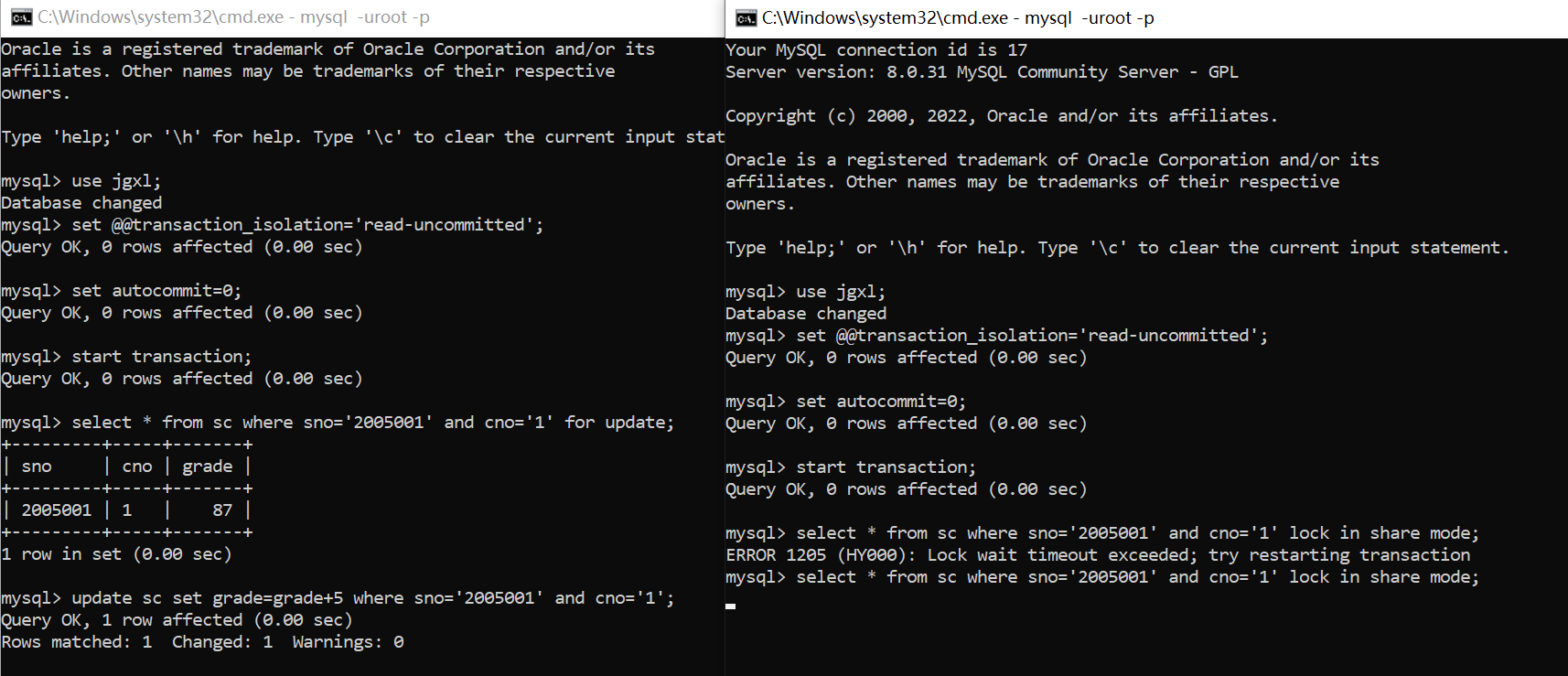
**2.2 重复表12-6**

表12-6解决了脏读，通过使用更高级别的事务隔离等级，如READ COMMITTED，或对查询添加共享锁或排他锁。

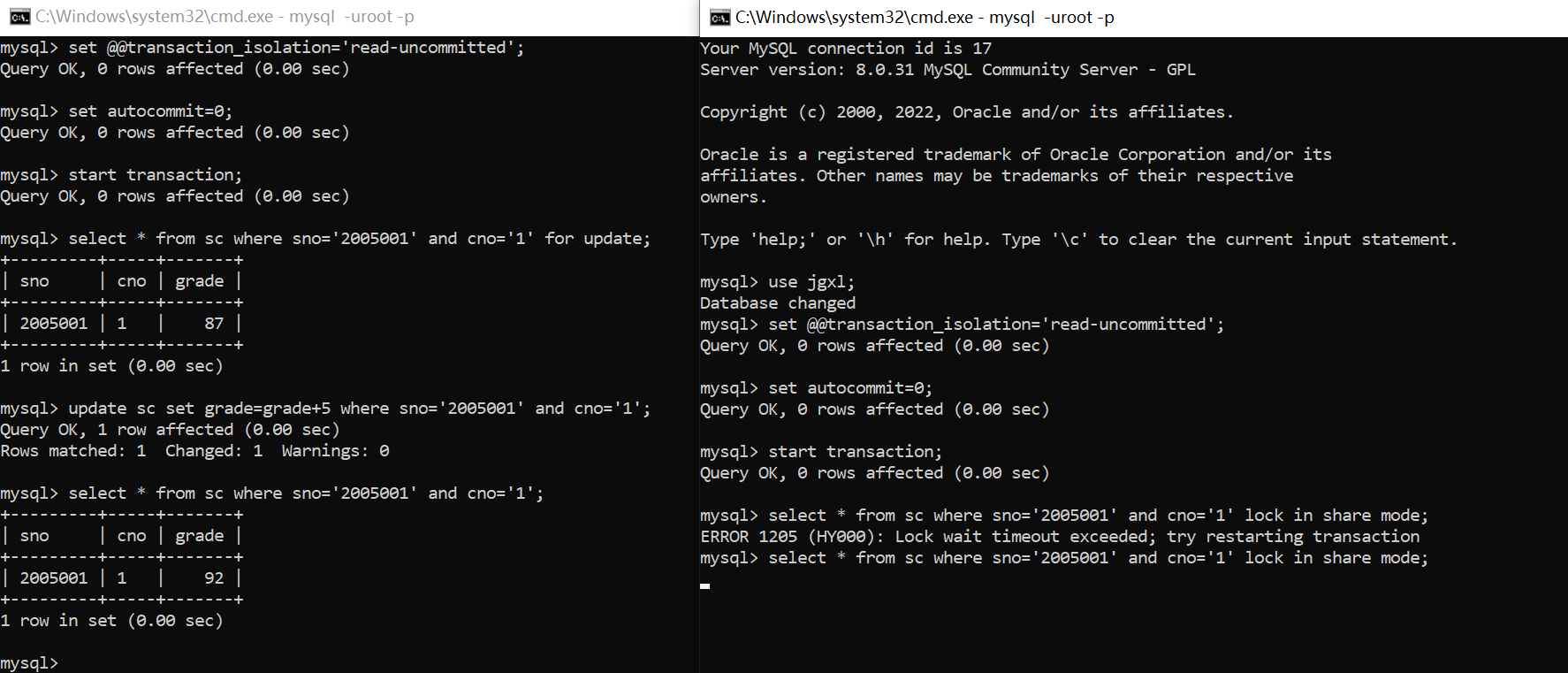
按表12-6，两个session都执行到第一个select语句，分别加上排他锁和共享锁。可看到session\_2进入“等待锁”状态



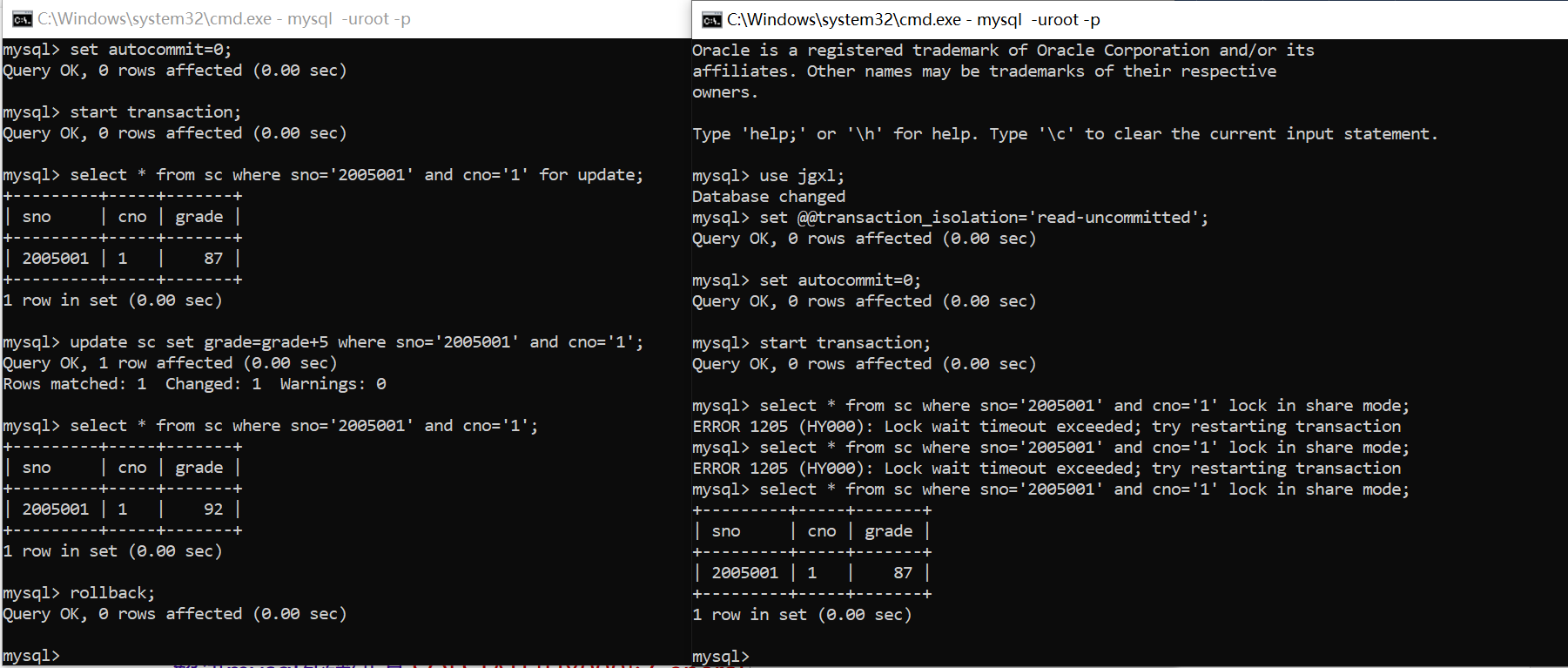
session\_1执行完grade操作，session\_2仍处于等待状态

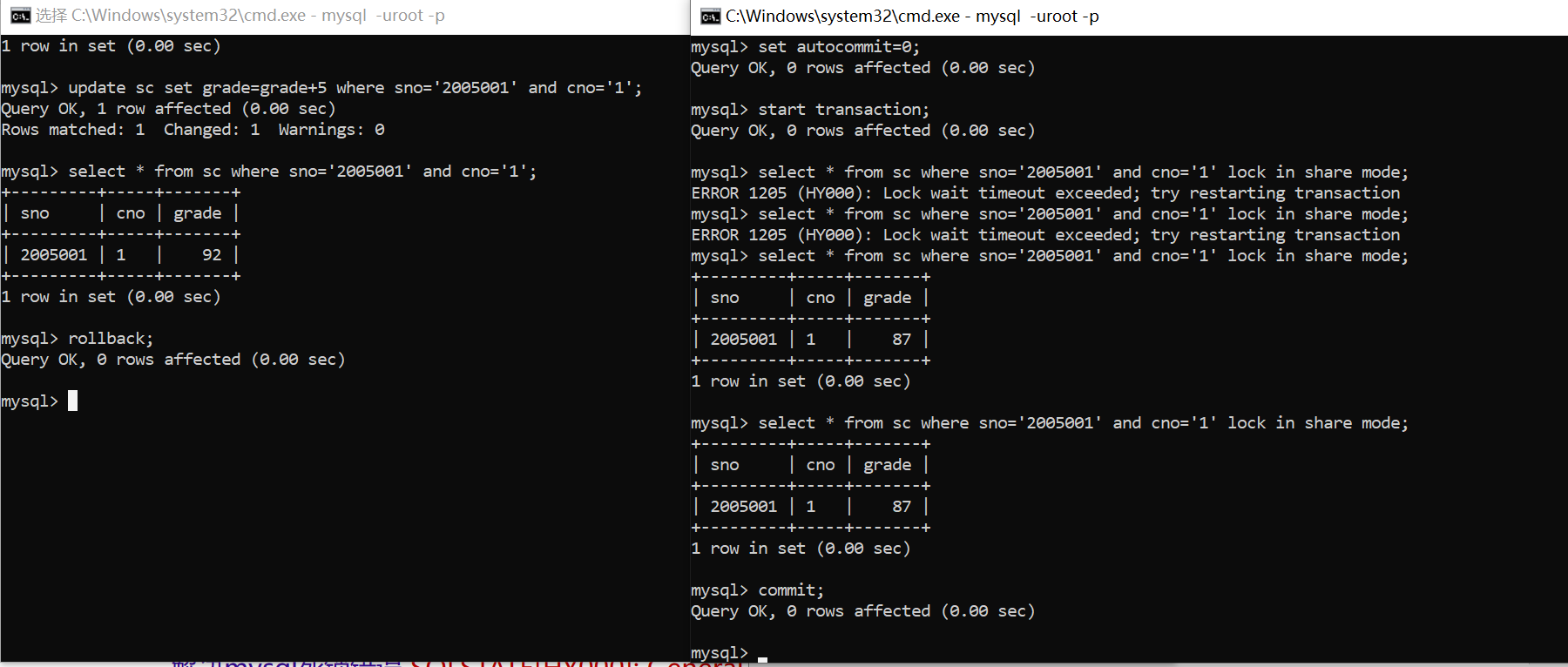


session\_1执行select操作重新查询，数据更新正常，session\_2仍处于等待状态



session\_1执行rollback操作后，session\_2成功获得锁，执行select语句



session\_2执行select操作再次查询，查询正常执行，最后commit

排他锁for update在上锁后，另一个线程不可进行读和修改，共享锁lock in share mode在上锁后，另一个线程只可以读，不可修改。因为先执行力session\_1的指令，在session\_1添加排他锁for update后，session\_2共享锁lock in share mode一直处于等待状态，所以无法得到查询结果，只有等待session\_1执行完rollback之后才能获得锁，得到查询结果。