《数据库系统实验》

实验报告

|  |  |
| --- | --- |
| **题目** | 实验12 |
| **姓名** | 郝裕玮 |
| **学号** | 18329015 |
| **班级** | 计科1班 |

一、实验环境

MySQL 命令行

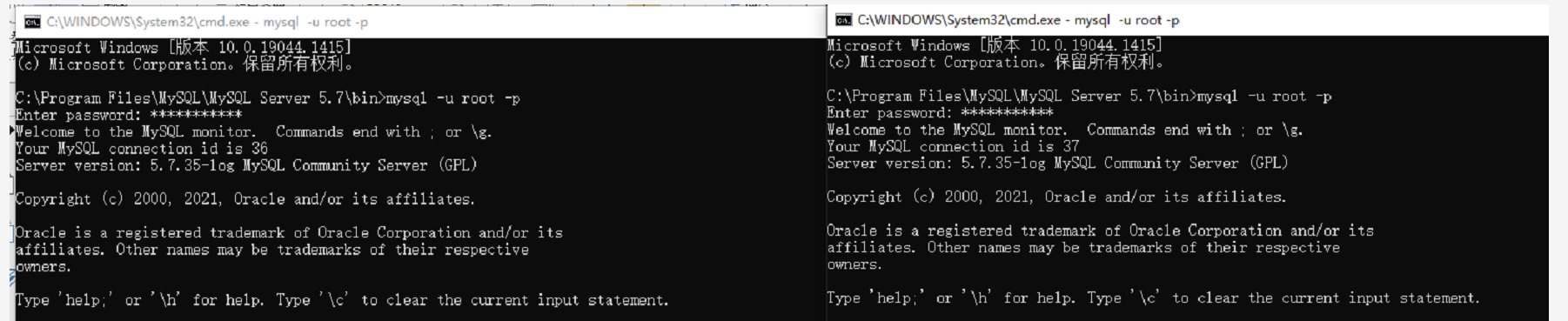
二、实验内容与完成情况

**12.1针对表12-5**



首先我们在该目录C:\Program Files\MySQL\MySQL Server 5.7\bin下打开两个cmd窗口。并连接到SQL。

连接到SQL的语句为：mysql -u root -p，执行后输入密码即可。



两个cmd窗口连接到SQL后，第一条语句均为：

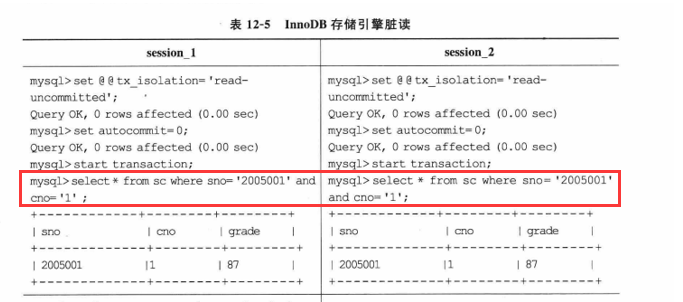
use jxgl

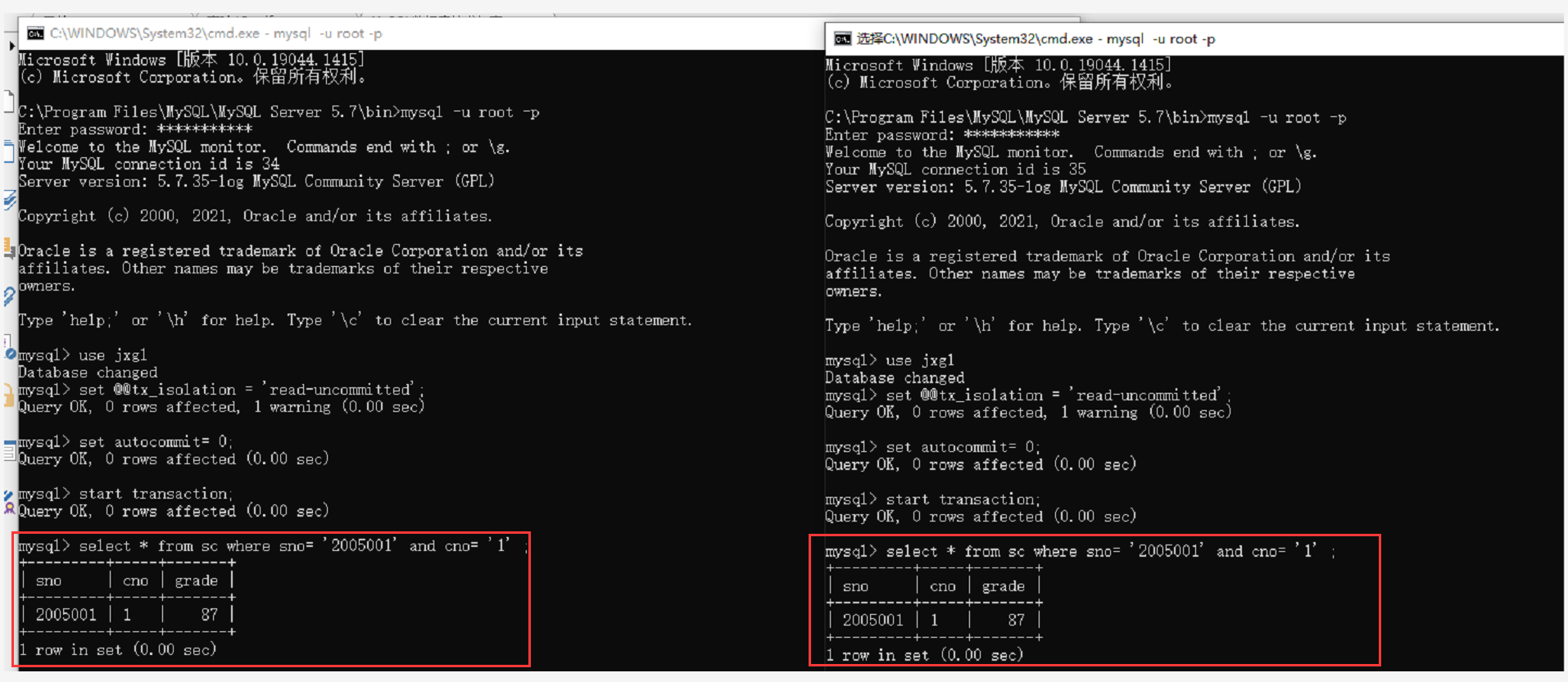
执行该语句以调用jxgl数据库。

执行后根据上图顺序输入语句即可。

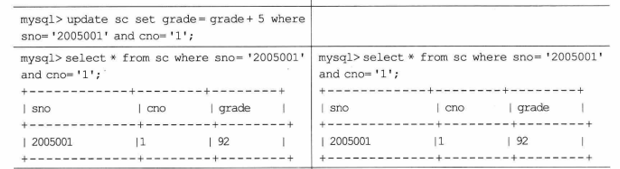
结果如下：

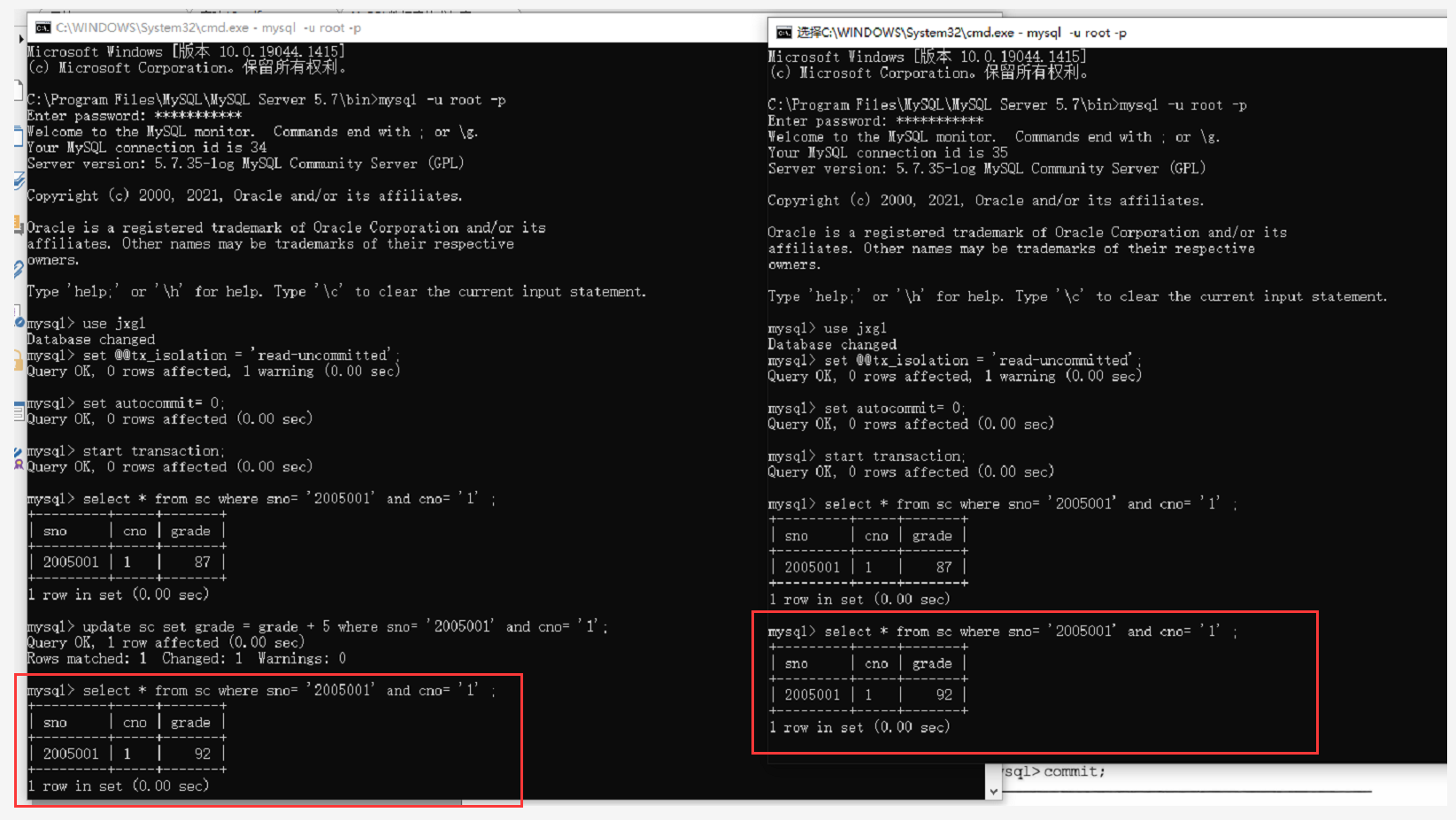
（1）两边均执行到这步时（结果图见下页）：



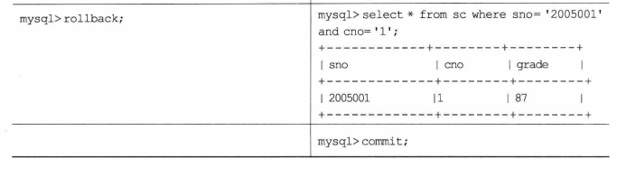


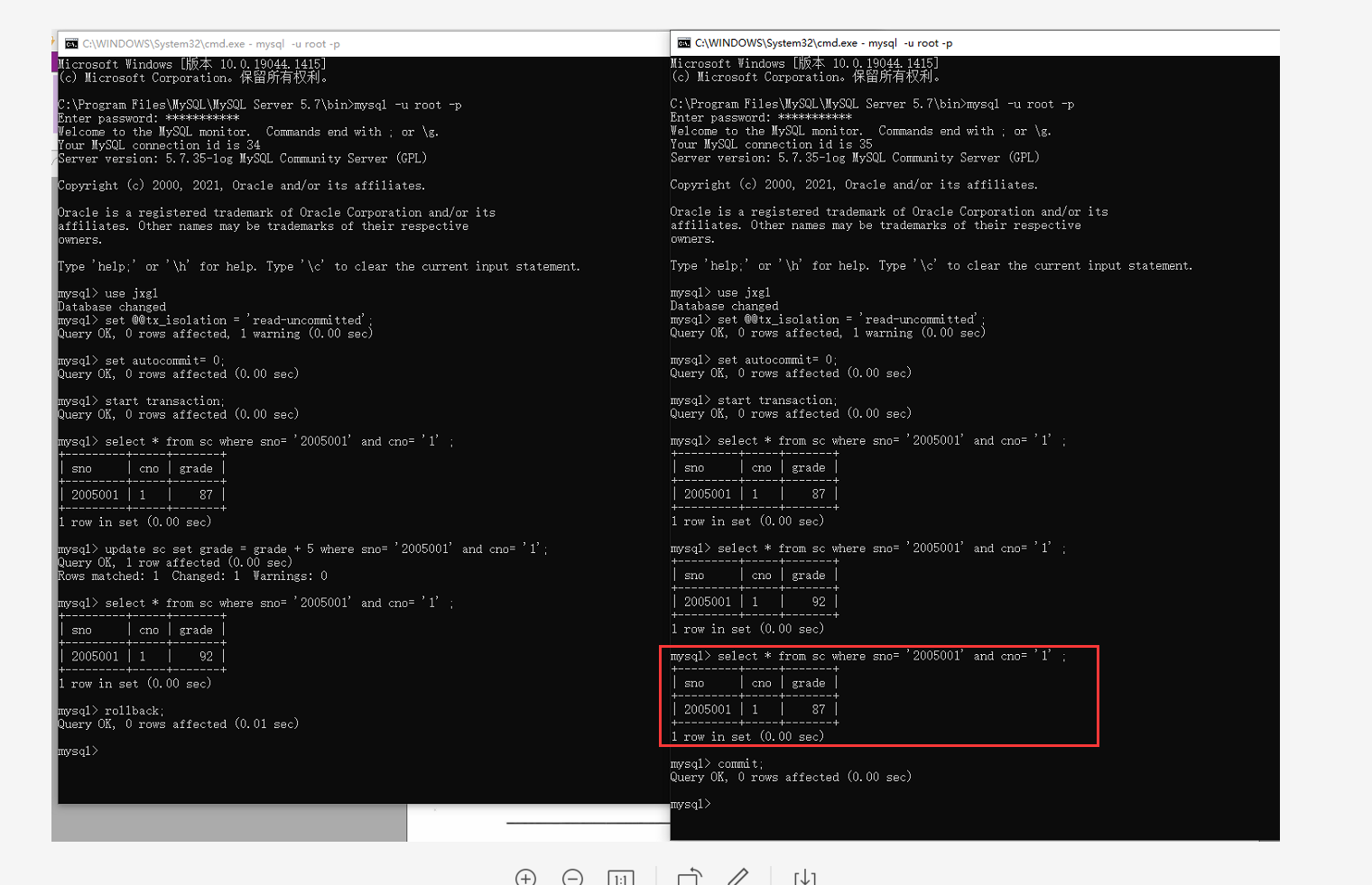
（2）session\_1更新完grade后：



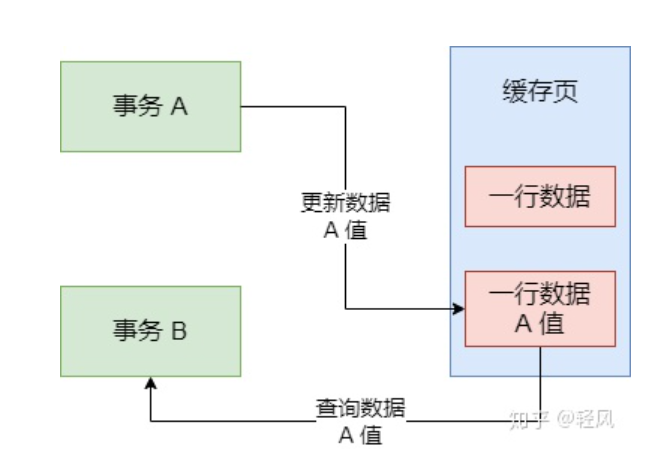


（3）session\_1执行rollback后：





由上述现象可知：隔离级别为未提交读（READ UNCOMMITTED）时会产生脏读。脏读指的是在不同的事务下，可以读到另外事务未提交的数据，则违反了数据库的隔离性。



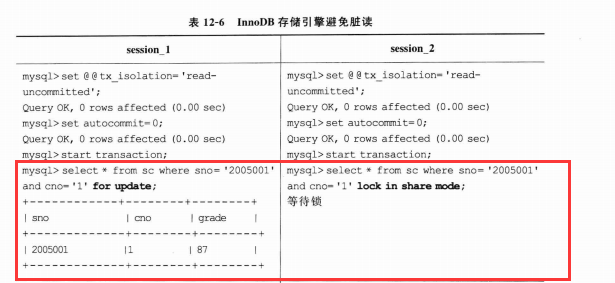
而本题中由于二者隔离级别相同，且未设置任何锁，所以会导致session\_2一直可以读到session\_1修改但未提交commit或回滚rollback的数据。

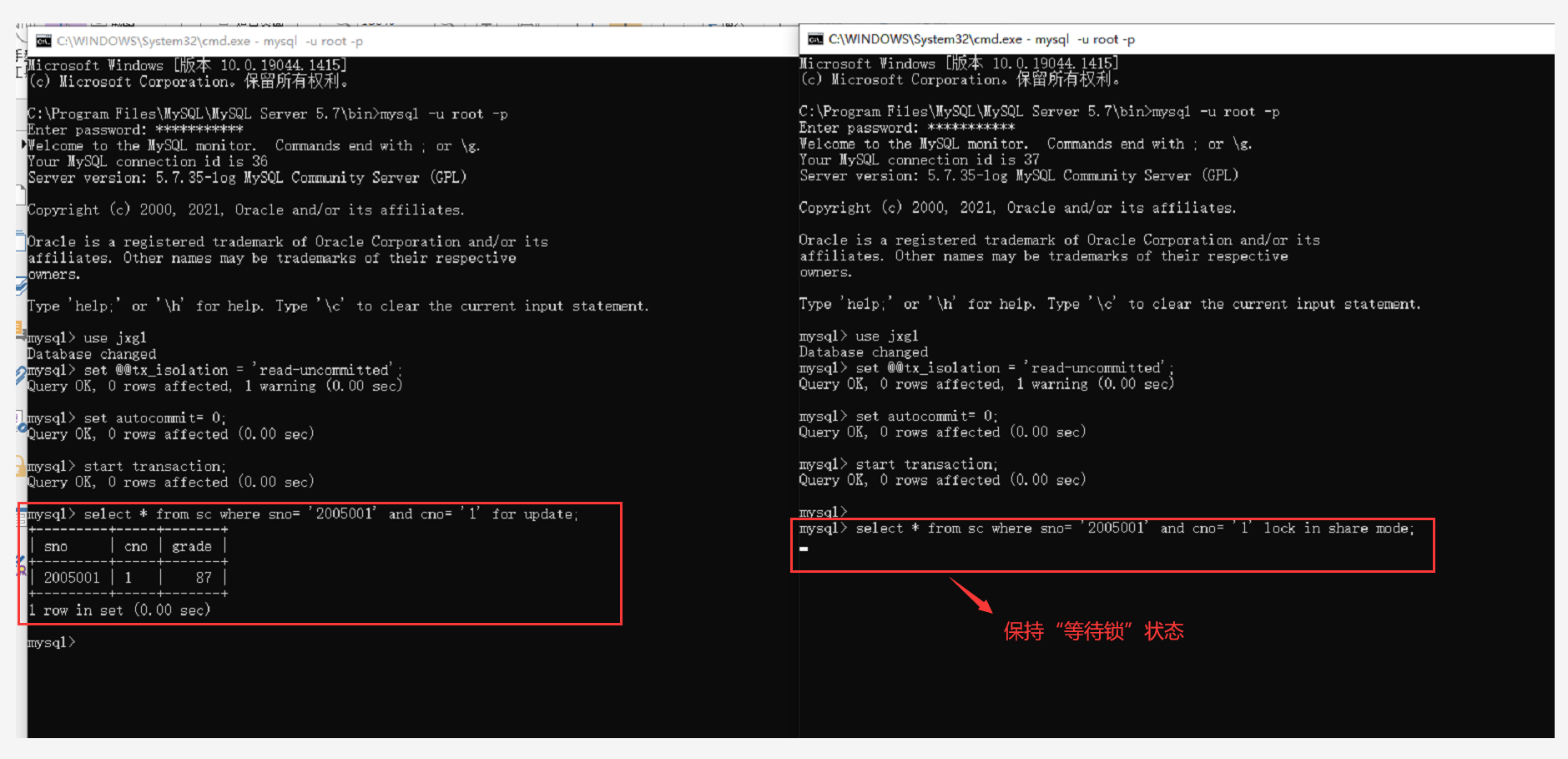
**12.2针对表12-6**



解决脏读的办法是指定更高级别的事务隔离级别，如READ COMMITTED，或者对查询添加共享锁或排他锁。

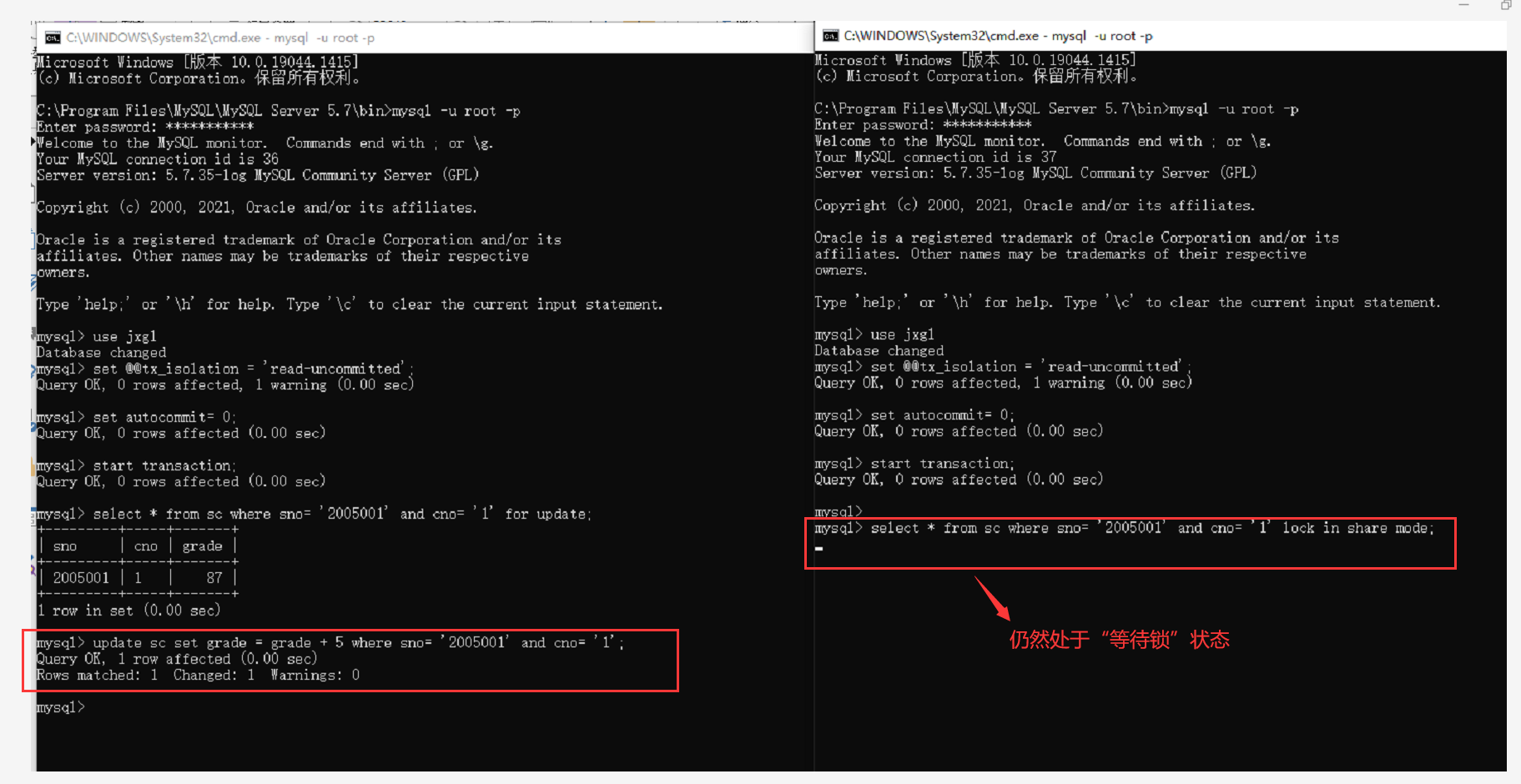
（1）两边执行到第一次select查询时：



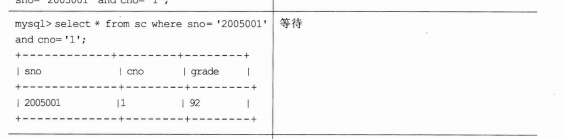


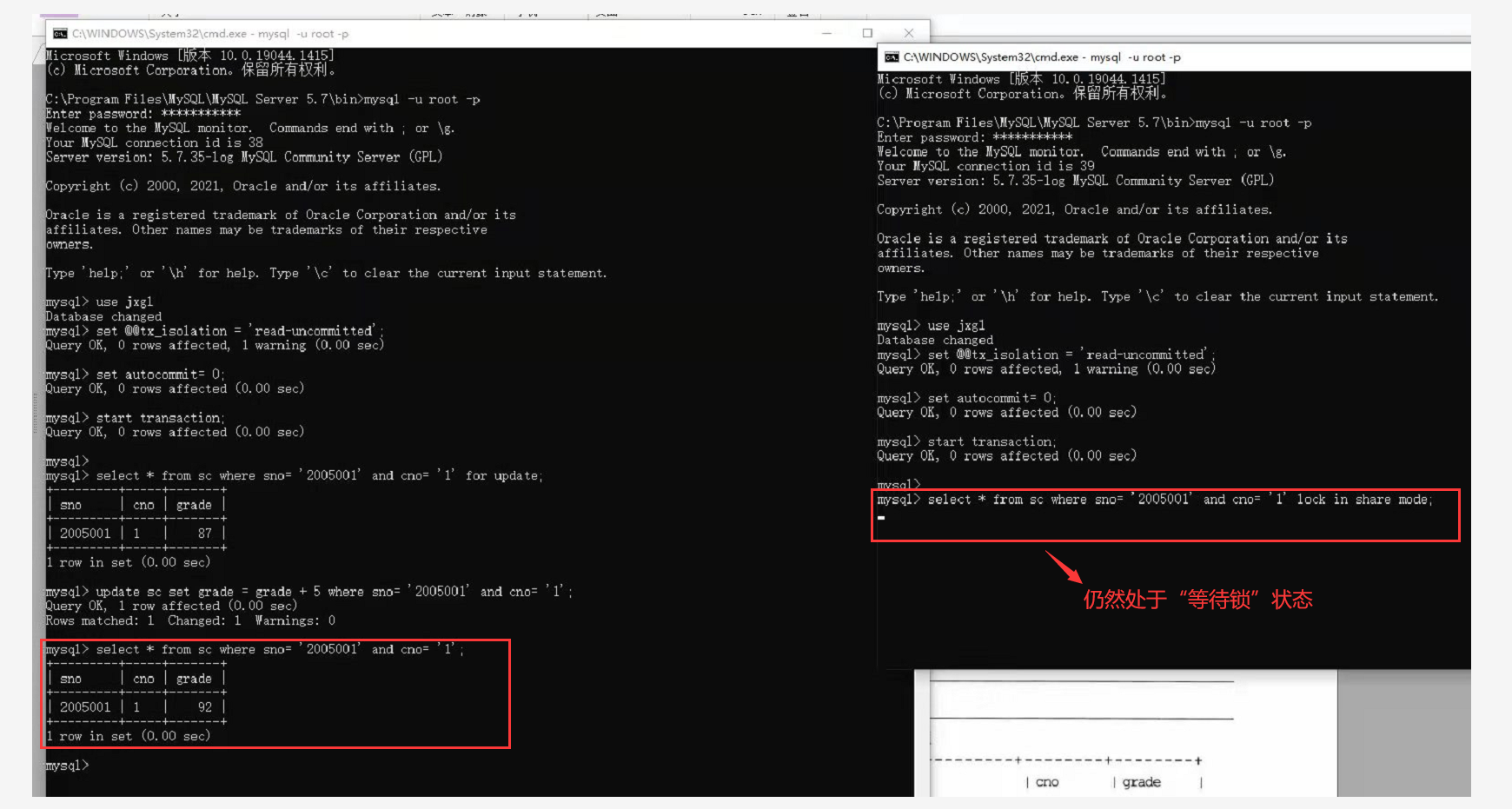
（2）session\_1更新完grade后（结果图见下页）：



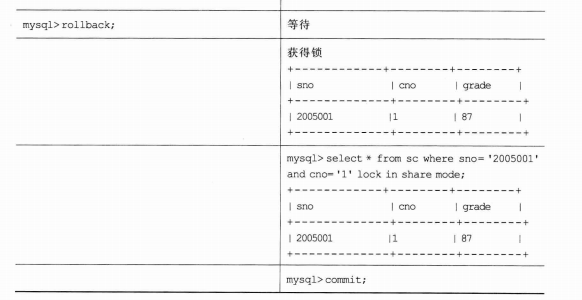


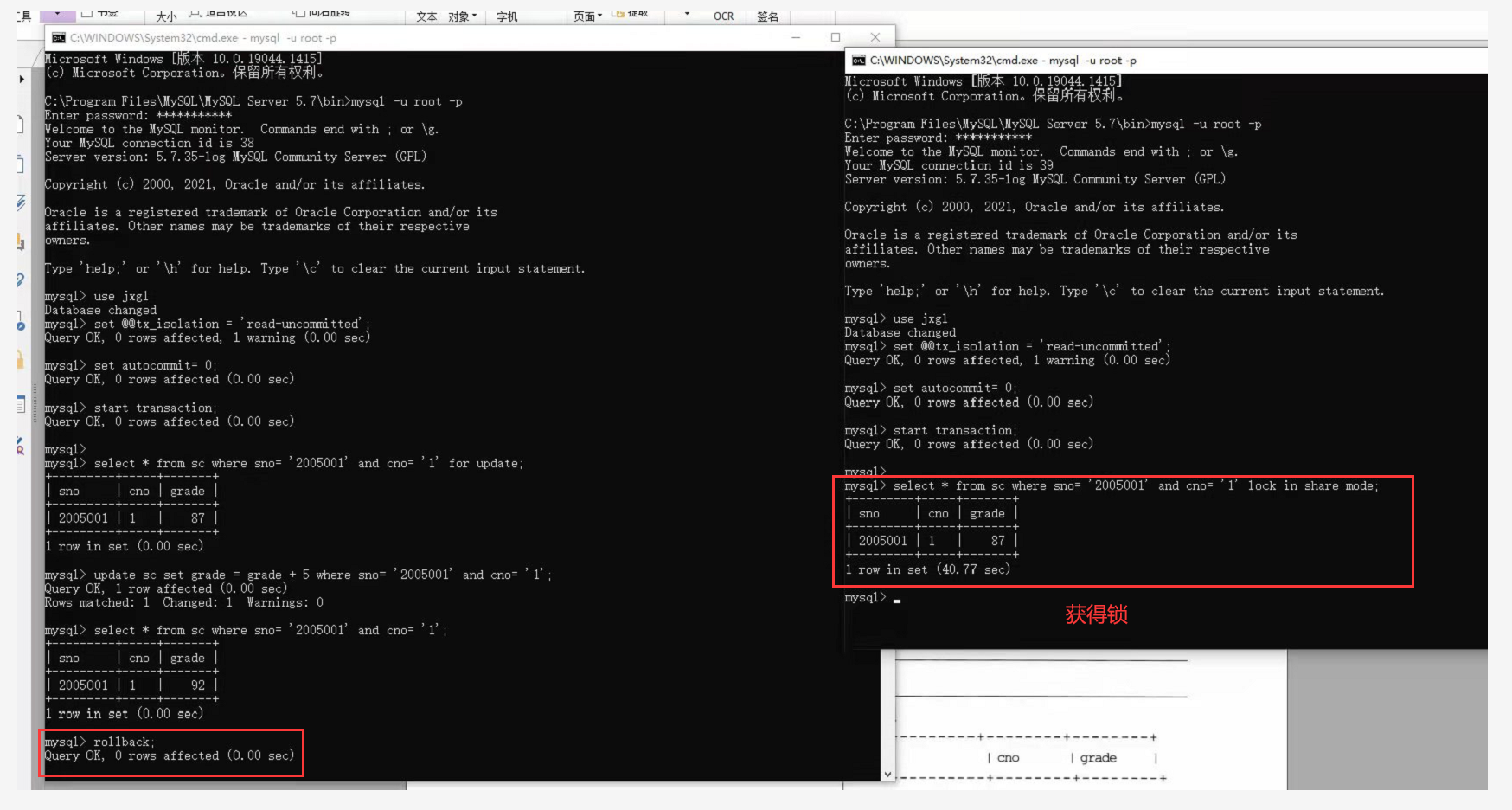
（3）session\_1更新完grade重新查询时（图见下页）：



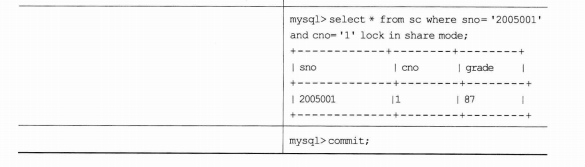


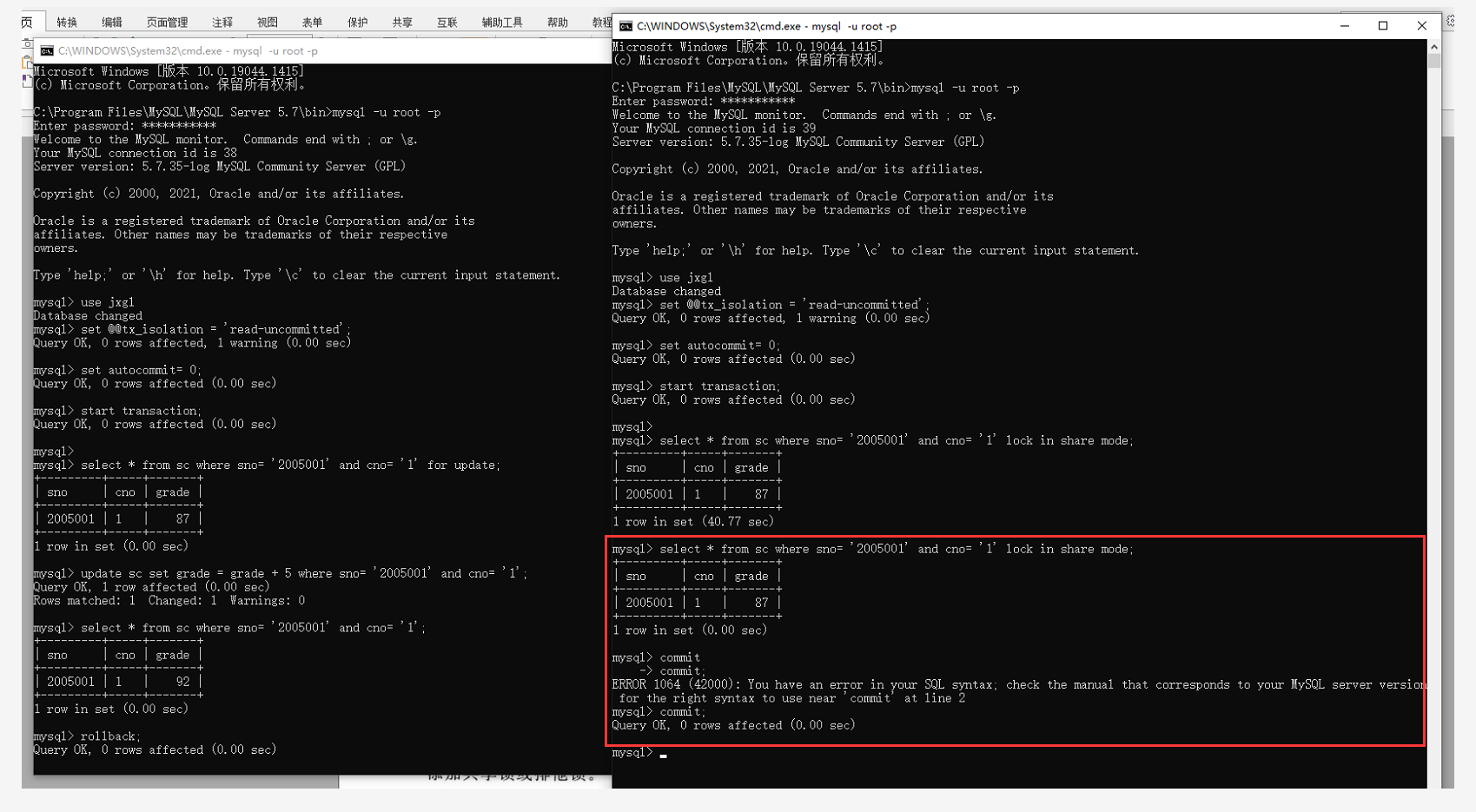
（4）session\_1执行rollback后（结果图见下页）：





（5）session\_2执行再次查询并commit（结果图见下页）：





接下来我们对该现象进行分析：

① 排他锁for update

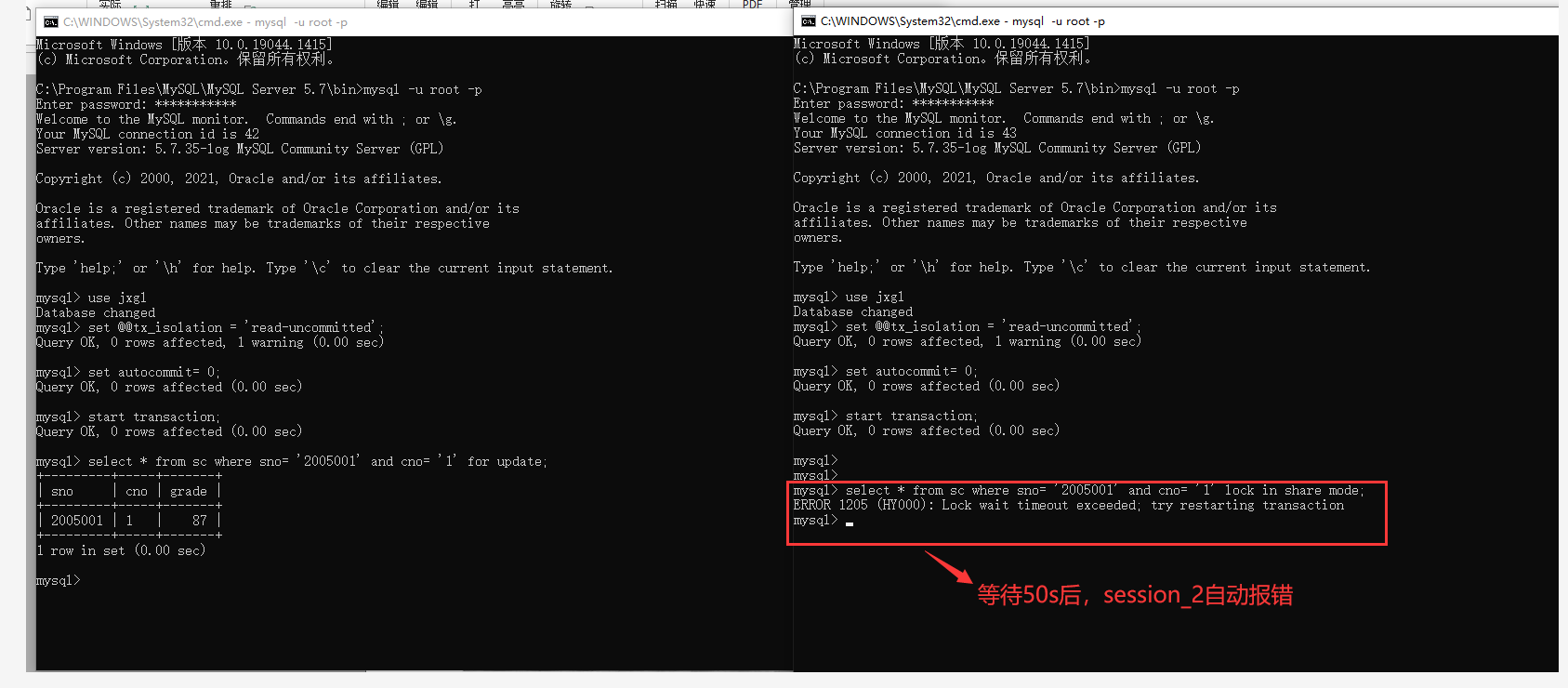
又称写锁（for update），例如update，insert，delete，上锁之后，另一个线程不可以读和修改。

② 共享锁lock in share mode

又称读锁（lock in share mode），例如select，当上锁之后，另一个线程只可以读，不可以修改。

所以在session\_1添加排他锁for update后，session\_2的共享锁lock in share mode就一直处于等待状态，无法获取到查询结果（等待另一个事务session\_1处理完）。直至session\_1执行rollback之后，session\_2才可以获得锁，得到select查询结果。

同时，对于共享锁lock in share mode，若在等待一定时间后未获取结果，则会直接报错。



最后再补充解释下2个session开头相同的3个SQL语句：

① set @@tx\_isolation = 'read-uncommitted';

用于修改事务隔离级别。

② set autocommit= 0;

当autocommit为0时，不管有没有START TRANSACTION，只有当commit数据才会生效，ROLLBACK后就会回滚。

③ start transaction;

这条命令开始一个新的事务。如果声明了隔离级别或者读写模式，那么新事务就使用这个特性。