一.事务的安全隐患

1.读的安全隐患

(1).脏读: 一个事务读到另外一个事务还未提交的数据。事务A读取了事务B更新的数据, 然后B回滚操作, 那么A读取到的数据是脏数据。

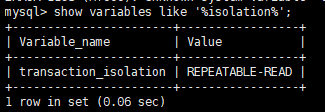
(2).不可重复读: 一个事务读到了另外一个事务提交的数据, 造成了前后两次查询结果不一致。事务A多次读取同一数据, 事务B在事务A多次读取的过程中,对数据作了更新并提交, 导致事务A多次读取同一数据时, 结果不一致。

(3).幻读: 一个事务读到了另一个事务insert的数据, 造成前后查询结果不一致。系统管理员A将数据库中所有学生的成绩从具体分数改为ABCDE等级, 但是系统管理员B就在这个时候插入了一条具体分数的记录, 当系统管理员A改结束后发现多出了一条记录, 就好像发生了幻觉一样, 这就叫幻读。

(4).不可重复读的和幻读很容易混淆, 不可重复读侧重于修改, 幻读侧重于新增或删除。解决不可重复读的问题只需锁住满足条件的行, 解决幻读需要锁表。

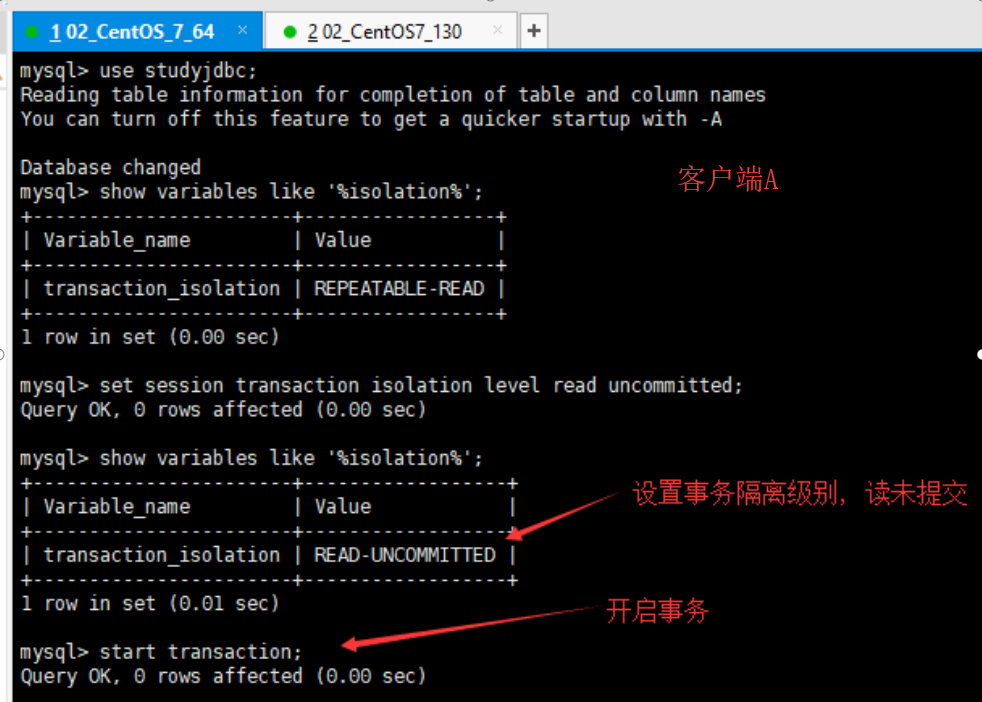
二.事务的隔离级别

1.查询当前的mysql8的默认隔离级别

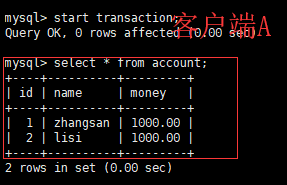


2.隔离级别第一级别:读未提交(read-uncommitted)

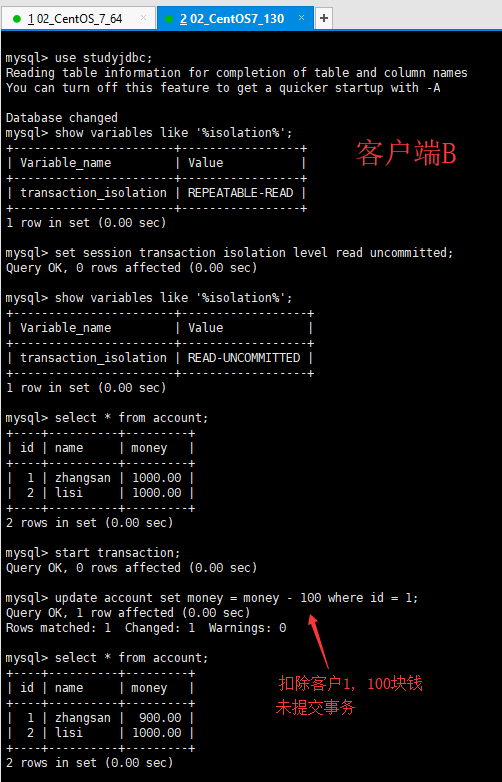
(1).打开客户端A, 先将事务的隔离级别设定为读未提交, 并开启事务



(2).客户端A, 查询用户信息



(3).客户端A未提交事务, 打开客户端B, 先将事务的隔离级别设定为读未提交, 并开启事务, 扣除客户1的100块钱, 不提交事务。



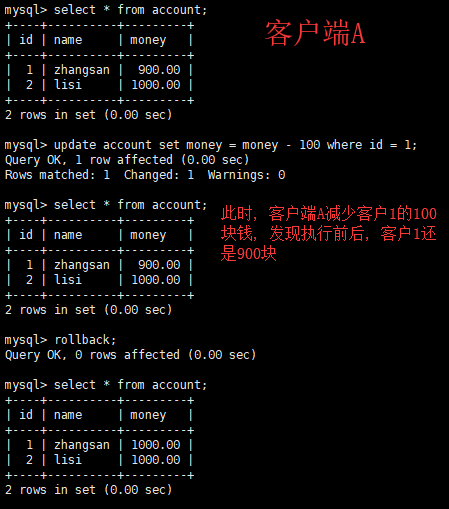
(4).客户端B未提交事务, 客户端A查看数据, 查到了客户端B更新的数据



(5).客户端B执行了回滚操作, 那么客户端A之前读到的数据就是脏数据



(6).此时, 客户端A减少客户1的100块钱, 发现执行前后, 客户1都是900块, 并没有变成800块。因为, 客户端A读到了客户端B更新的数据, 后来客户端B又回滚了数据, 客户端A读到了一条脏数据。

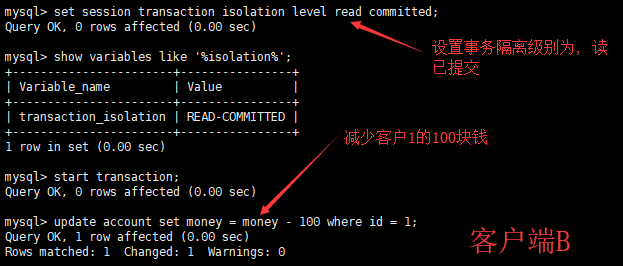


3.读已提交(第二级别 read-committed)解决了脏读问题, 但是出现了不可重复读问题, 存在幻读问题。

(1).打开客户端A, 设置当前的事务隔离级别为read committed(未提交读), 开启事务, 查询表account的所有记录



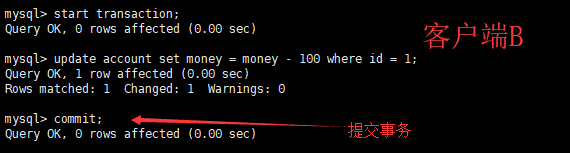
(2).打开客户端B, 设置当前的事务隔离级别为read committed(未提交读), 开启事务, 减少客户1的100块钱, 未提交事务。



(3).客户端A, 在客户端B更新数据前后, 查询的数据一样, 解决了脏的问题



(4).客户端B提交事务

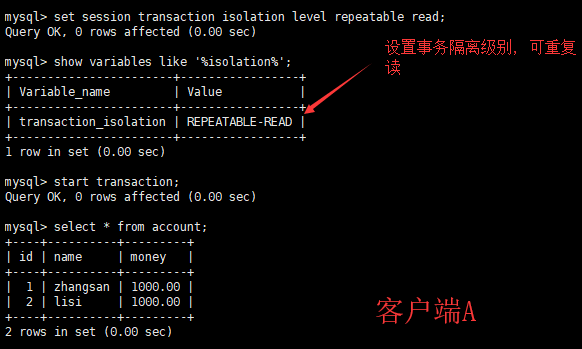


(5).客户端A查询到了客户端B提交的数据。产生了不可重复读的问题, 客户端A进行同样的2次查询, 2次查询结果不一致。



4.可重复读(第三级别 repeatable read)

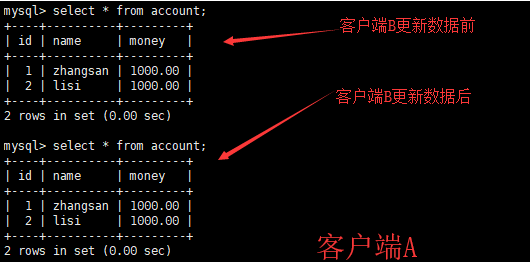
(1).打开客户端A, 设置当前的事务隔离级别为repeatable read (可重复读), 开启事务, 查询表account的所有记录



(2).打开客户端B, 设置当前的事务隔离级别为repeatable read (可重复读), 开启事务, 扣除客户1的100块钱, 未提交事务



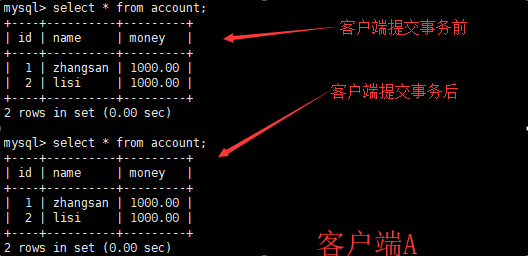
(3).客户端B更新数据, 未提交事务, 查询客户端A, 客户端B更新数据前后, 客户端的数据不变, 解决了脏读问题



(4).客户端B提交事务



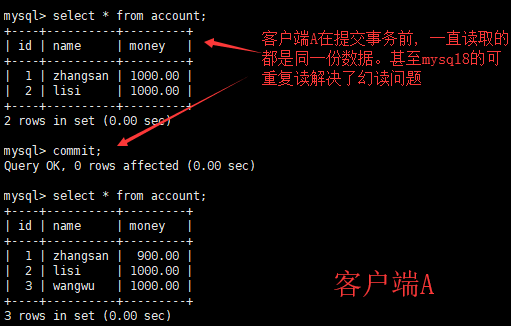
(5).客户端B提交事务后, 查询客户端A, 数据依然不变



(6).客户端B插入一条数据, 并且提交事务

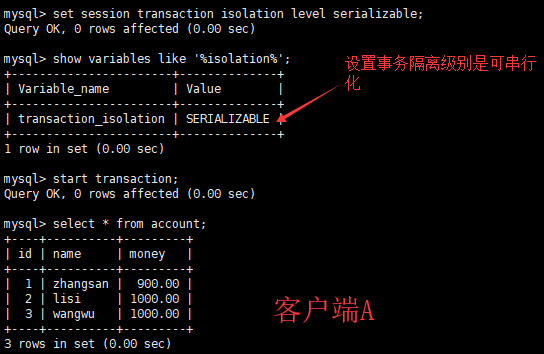


(7).客户端B插入一条数据, 并且提交事务后, 客户端A查询数据, 依然是以前的数据。客户端A提交事务后, 查询出了最新数据。



5.可串行化(第四级别 serializable)

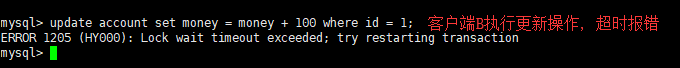
(1).打开客户端A, 设置事务隔离级别为可串行化, 开启事务, 查询account表



(2).打开客户端B, 设置事务隔离级别为可串行化, 开启事务, 查询account表, 客户端B可以进行查询操作



(3).在客户端B, 执行一个更新操作, 客户端B卡住了, 如果此时客户端A提交 或回滚事务, 客户端B会更新成功; 如果客户端A长时间没有提交或回滚事务, 客户端B的更新操作就超时报错。



(4).如果有一个连接的隔离级别设置为了串行化， 那么谁先打开了事务, 就会在整张表上加了一个锁, 其它事务不能进行写操作。这种事务隔离级别解决了所有的安全问题。但是这种隔离级别一般比较少用, 容易造成性能上的问题, 效率比较低。

6.事务的隔离级别, 按效率划分, 从高到低: 读未提交-->读已提交-->可重复读-->可串行化。

7.事务的隔离级别, 按拦截程度, 从高到底: 可串行化-->可重复读-->读已提交-->读未提交。

8.MySQL和Oracle默认的事务隔离级别

(1).MySQL默认的事务隔离级别是:可重复读。

(2).Oracle默认的事务隔离级别是:读已提交。

9.隔离级别

(1).读未提交, 引发问题脏读。

(2).读已提交, 解决脏读, 引发不可重复读。

(3).可重复读, 解决脏读、不可重复读, 引发幻读。

(4).可串行化, 解决脏读、不可重复读、幻读。