

北京邮电大学 计算机学院
《数据库系统原理》实验报告

姓名 王睿嘉

学号 2015211906

班级 2015211307

实验七 事务创建与运行

一、实验内容 and 环境描述

1. 实验目的

通过实验，了解 MySQL 中各类事务的定义和基于锁的并发控制机制，掌握 MySQL 数据库系统的事务控制机制。

2. 实验内容

- 1) 定义三种模式的数据库事务；
- 2) 查看事务的隔离级别。

3. 实验环境

数据库系统：5.7.21 MySQL Community Server (GPL)

二、实验结论和心得

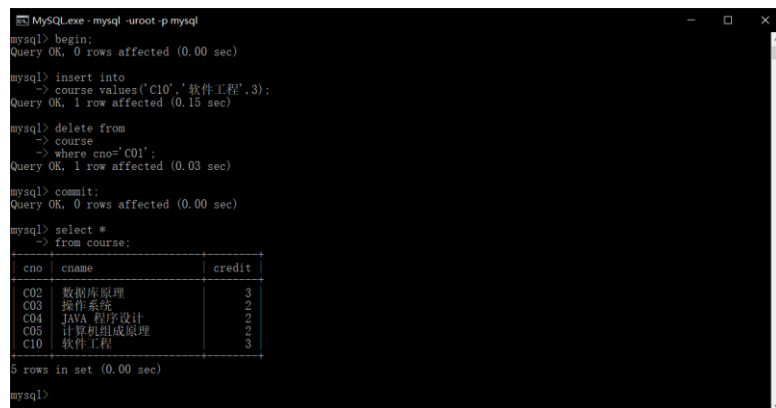
1. 实验结论

1.1 定义三种模式的数据库事务

1) 显式事务

显式事务是一种由用户自己指定的事务，允许自己决定哪一批工作必须成功完成，否则所有工作都不完成。操作包括 begin, rollback 和 commit 等。

例如，建立一个显式事务，删除和插入记录同时进行：



```
MySQL.exe - mysql -uroot -p mysql
mysql> begin;
Query OK, 0 rows affected (0.00 sec)

mysql> insert into
  -> course values('C10','软件工程',3);
Query OK, 1 row affected (0.13 sec)

mysql> delete from
  -> course
  -> where cno='C01';
Query OK, 1 row affected (0.03 sec)

mysql> commit;
Query OK, 0 rows affected (0.00 sec)

mysql> select *
  -> from course;
+----+-----+-----+
| cno | cname | credit |
+----+-----+-----+
| C02 | 数据库原理 | 3 |
| C03 | 操作系统 | 4 |
| C04 | JAVA 程序设计 | 4 |
| C05 | 计算机组成原理 | 4 |
| C10 | 软件工程 | 3 |
+----+-----+-----+
6 rows in set (0.00 sec)

mysql>
```

删除和插入要么同时成功，要么同时失败，这正是显式事务的特点。

2) 自动提交事务

MySQL 的默认模式，无需加 begin、rollback 和 commit 等。

例如：

```
MySQL.exe - mysql -uroot -p mysql
mysql> insert into
  -> course values('C10','软件工程',3);
Query OK, 1 row affected (0.02 sec)

mysql> select *
  -> from course;
+----+-----+-----+
| cno | cname | credit |
+----+-----+-----+
| C01 | 编译原理 | 3 |
| C02 | 数据库原理 | 3 |
| C03 | 操作系统 | 3 |
| C04 | JAVA 程序设计 | 3 |
| C05 | 计算机组成原理 | 3 |
| C10 | 软件工程 | 3 |
+----+-----+-----+
6 rows in set (0.00 sec)

mysql>
```

将每条单独的 SQL 语句视为一个事务。如果成功执行，则自动提交，否则回滚。

3) 隐式事务

打开隐式事务：set autocommit=OFF。

在前一个事务完成时，新事务隐式启动，但每个事务仍以 commit 或 rollback 语句显式完成。

例如：

```
MySQL.exe - mysql -uroot -p mysql
mysql> set autocommit=OFF;
Query OK, 0 rows affected (0.00 sec)

mysql> insert into
  -> course values('C10','软件工程',3);
Query OK, 1 row affected (0.00 sec)

mysql> commit;
Query OK, 0 rows affected (0.00 sec)

mysql> select *
  -> from course;
+----+-----+-----+
| cno | cname | credit |
+----+-----+-----+
| C01 | 编译原理 | 3 |
| C02 | 数据库原理 | 3 |
| C03 | 操作系统 | 3 |
| C04 | JAVA 程序设计 | 3 |
| C05 | 计算机组成原理 | 3 |
| C10 | 软件工程 | 3 |
+----+-----+-----+
6 rows in set (0.00 sec)

mysql>
```

在提交或回滚事务后自动启动新事务，无需描述事务的开始。

1.2 查看事务的隔离级别

```
MySQL.exe - mysql -uroot -p mysql
mysql> select @@global.tx_isolation;
+-----+
| @@global.tx_isolation |
+-----+
| REPEATABLE-READ |
+-----+
1 row in set, 1 warning (0.00 sec)

mysql>
```

REPEATABLE-READ 是 MySQL 默认的事务隔离级别，即重读读取，可解决脏读和不可重复读问题，但可能存在幻读。

上述语句中，部分术语定义如下。

脏读：事务 A 读取了事务 B 更新的数据，然后 B 回滚，那么 A 读取到的数据称为脏数据。

不可重复读：事务 A 多次读取同一数据，事务 B 在 A 读取的过程中，对数据做了更新并提交，导致 A

读取结果不一致。

幻读：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 A、B、C、D 和 E 等级，但系统管理员 B 在这时插入了一条具体分数的记录。从而，当 A 修改完成后发现还有一条记录没有更改，好像发生了幻觉一样。

2. 实验心得

在本次实验中，遇到的主要问题有以下两点：

- ① 隐式事务的开启。MySQL 默认自动提交事务。经查阅网上资料，使用 `set autocommit=OFF` 语句关闭默认模式，即可达到相应要求；
- ② 隔离级别的理解。MySQL 存在 4 种事务隔离级别：READ-UNCOMMITTED、READ-COMMITTED、REPEATABLE-READ 和 SERIALIZABLE。通过有关博客，清晰了每种级别的含义。隔离级别越高，越能保证数据的完整性和一致性，但对并发性能的影响也越大。对于多数应用程序，可以优先考虑把隔离级别设为 READ-COMMITTED。该模式可以避免脏读，且具有较好的并发性能。尽管会导致不可重复读等问题，但可由应用程序的相关锁进行控制。

实践出真知，本次事务创建与运行实验是对课堂和书本所学知识的补充。通过自己动手、亲力亲为编写事务语句，加深了对事务模式的理解和记忆，收获颇丰。