

《数据库系统实验》

实验报告

题目	实验 13
姓名	郝裕玮
学号	18329015
班级	计科 1 班

一、实验环境

MySQL 命令行 + MySQL Workbench 8.0 CE

二、实验内容与完成情况

针对教学管理系统 (jxgl) 做如下操作：

12.1 对表 sc, course 和 student 分别做表备份和表恢复。

(1) 表备份

在 MySQL Workbench 中执行以下语句：

```
use jxgl;

lock tables student read;
select * into outfile 'student.bak' from student;
unlock tables;

lock tables sc read;
select * into outfile 'sc.bak' from sc;
unlock tables;

lock tables course read;
select * into outfile 'course.bak' from course;
unlock tables;
```

然而在执行该行 SQL 语句时报错：

Error Code: 1290. The MySQL server is running with the --secure-file-priv option so it cannot execute this statement

✓	2	21:05:43	use jxgl	0 row(s) affected	0.000 sec
✓	3	21:05:45	lock tables student read	0 row(s) affected	0.000 sec
✗	4	21:05:47	select * into outfile 'student.bak' from student	Error Code: 1290. The MySQL server is running with the --secure-file-priv option so it...	0.000 sec

查询得知需要修改 my.ini 文件。

修改步骤如下：

1. 文件夹中输入 %ProgramData% 并定位到 C:\ProgramData\MySQL\MySQL Server

5.7

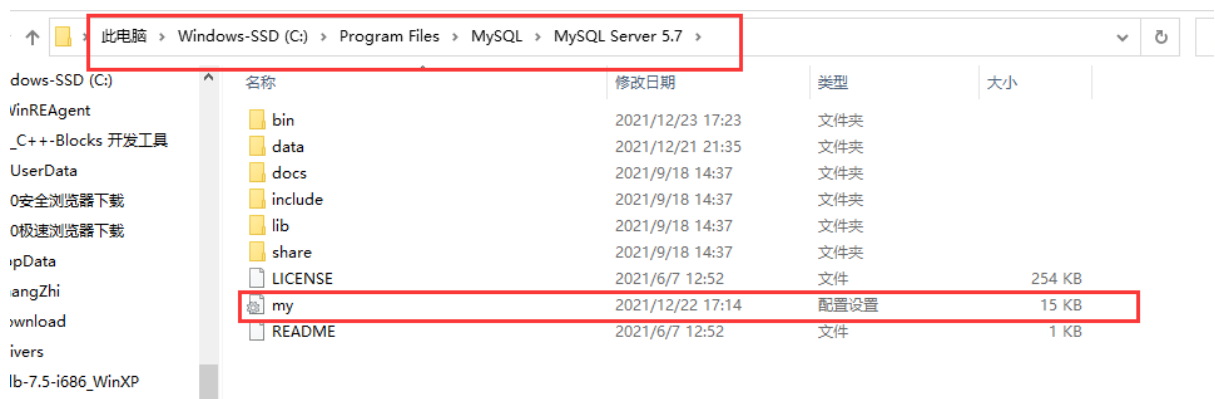
2. 修改 my.ini 文件，将 secure-file-priv 的定位目录修改为''

```
# ***** Group Replication Related *****  
# Defines the algorithm used to hash the writes extracted during a transaction. If you  
# are using Group Replication, this variable must be set to XXHASH64 because the process  
# of extracting the writes from a transaction is required for conflict detection on all  
# group members.  
# transaction_write_set_extraction=0.0  
lower_case_table_names=1
```

```
# Secure File Priv.  
secure-file-priv=""
```

并根据下图，将 my.ini 文件复制到指定位置

```
# To run the server from the command line, execute this in a  
# command line shell, e.g.  
# mysqld --defaults-file="C:\Program Files\MySQL\MySQL Server X.Y\my.ini"  
#  
# To install the server as a Windows service manually, execute this in a  
# command line shell, e.g.  
# mysqld --install MySQLXY --defaults-file="C:\Program Files\MySQL\MySQL Server X.Y\my.ini"  
#  
# And then execute this in a command line shell to start the server, e.g.  
# net start MySQLXY  
..
```



复制完成后，打开 cmd 输入 services.msc，找到 MySQL 服务，右键重启，并执行语

句：SHOW VARIABLES LIKE "secure_file_priv"来查看 secure-file-priv 是否修改成功。

若修改成功，则重新执行上述语句即可。

	Variable_name	Value
▶	secure_file_priv	

✓	6	21:35:57	use jxgl	0 row(s) affected	0.000 sec
✓	7	21:36:01	lock tables student read	0 row(s) affected	0.016 sec
✓	8	21:36:04	SHOW VARIABLES LIKE "secure_file_priv"	1 row(s) returned	0.000 sec / 0.000 sec
✓	9	21:37:00	select * into outfile 'student.bak' from student	9 row(s) affected	0.000 sec
✓	10	21:37:59	unlock tables	0 row(s) affected	0.000 sec
✓	11	21:38:01	lock tables sc read	0 row(s) affected	0.015 sec
✓	12	21:38:03	select * into outfile 'sc.bak' from sc	18 row(s) affected	0.000 sec
✓	13	21:38:04	unlock tables	0 row(s) affected	0.000 sec
✓	14	21:38:09	lock tables course read	0 row(s) affected	0.000 sec
✓	15	21:38:11	select * into outfile 'course.bak' from course	7 row(s) affected	0.000 sec
✓	16	21:38:13	unlock tables	0 row(s) affected	0.000 sec

至此，备份成功！

(2) 表恢复

在 MySQL Workbench 中执行以下语句：

```
set FOREIGN_KEY_CHECKS = 0;
lock tables student write;
load data infile 'student.bak' replace into table student;
unlock tables;
set FOREIGN_KEY_CHECKS = 1;

set FOREIGN_KEY_CHECKS = 0;
lock tables sc write;
load data infile 'sc.bak' replace into table sc;
unlock tables;
set FOREIGN_KEY_CHECKS = 1;

set FOREIGN_KEY_CHECKS = 0;
lock tables course write;
load data infile 'course.bak' replace into table course;
unlock tables;
set FOREIGN_KEY_CHECKS = 1;
```

一开始未加 set FOREIGN_KEY_CHECKS = 0，直接执行第 3 行语句时会报错：

Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails
 (`jxgl`.`sc`, CONSTRAINT `sc_ibfk_1` FOREIGN KEY (`sno`) REFERENCES `student`
 (`sno`))

18	21:46:37	lock tables student write	0 row(s) affected	0.000 sec
19	21:46:46	load data infile 'student.bak' replace into table student	Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fail...	0.000 sec
20	21:47:55	SET FOREIGN_KEY_CHECKS = 0	0 row(s) affected	0.000 sec
21	21:47:58	lock tables student write	0 row(s) affected	0.000 sec
22	21:48:01	load data infile 'student.bak' replace into table student	18 row(s) affected Records: 9 Deleted: 9 Skipped: 0 Warnings: 0	0.015 sec
23	21:48:33	unlock tables	0 row(s) affected	0.000 sec

查询得知，该条记录的某个字段是作为 innodb 的另外一个表的外键。而在进行删除操作时会自动检查外键。

所以设置不在执行语句时不检查外键即可：

```
set FOREIGN_KEY_CHECKS = 0;
```

并在执行结束后将其重置为 1

```
set FOREIGN_KEY_CHECKS = 1;
```

12.2 对教学管理系统采用不同方法进行完整备份和恢复。

完整备份：

(1) MySQL 脚本

以下 3 行语句均可备份：

```
mysqldump -u root -p jxgl > jxgl_version1.sql
mysqldump -u root -p --databases jxgl > jxgl_version2.sql
mysqldump -u root -p --all-databases > jxgl_version3.sql
```

第 1 种：导出单个数据库

第 2 种：导出多个数据库

第 3 种：导出所有数据库

```
管理员: C:\WINDOWS\System32\cmd.exe
Microsoft Windows [版本 10.0.19044.1415]
(c) Microsoft Corporation。保留所有权利。

C:\WINDOWS\system32>cd C:\Program Files\MySQL\MySQL Server 5.7\bin

C:\Program Files\MySQL\MySQL Server 5.7\bin>mysqldump -u root -p jxgl > jxgl_version1.sql
Enter password: *****

C:\Program Files\MySQL\MySQL Server 5.7\bin>mysqldump -u root -p --databases jxgl > jxgl_version2.sql
Enter password: *****

C:\Program Files\MySQL\MySQL Server 5.7\bin>mysqldump -u root -p --all-databases > jxgl_version3.sql
Enter password: *****

C:\Program Files\MySQL\MySQL Server 5.7\bin>
```

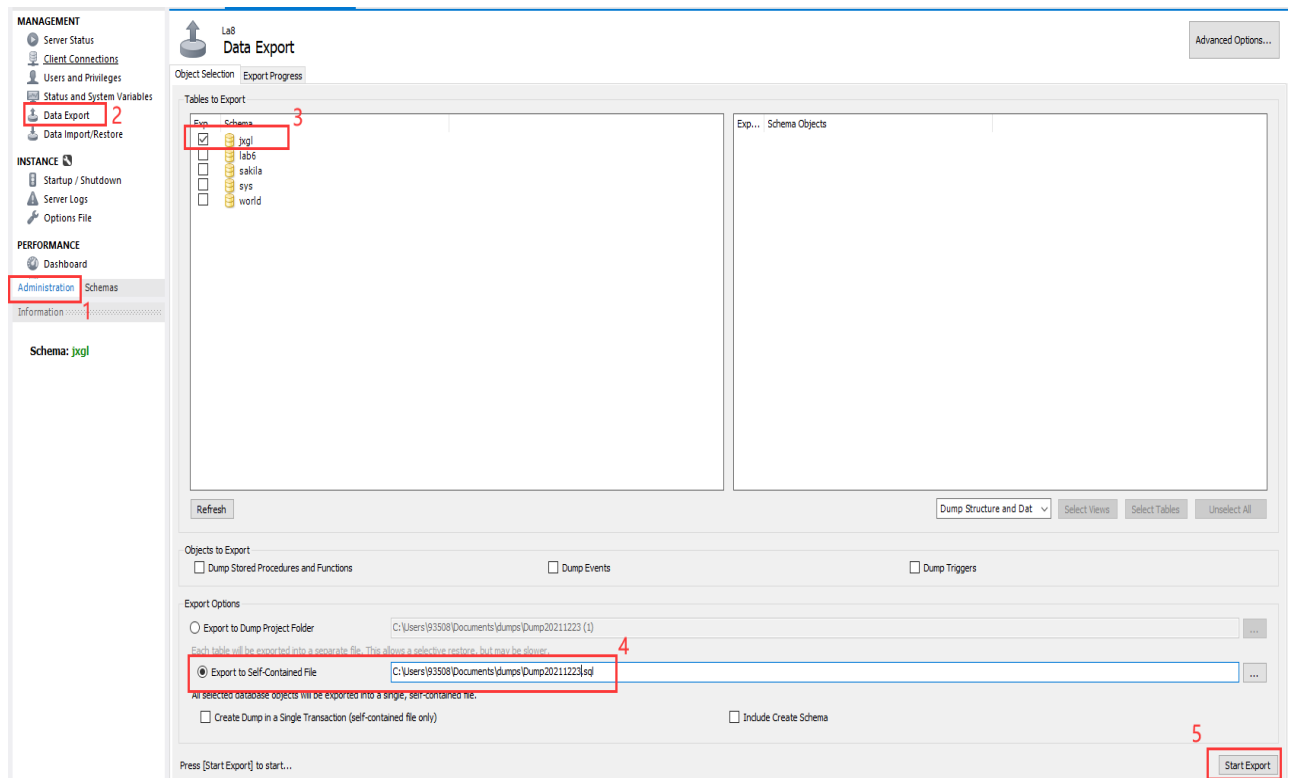
注意，以上语句需在 cmd 管理员模式下执行。

执行后可在对应文件夹下发现备份文件。

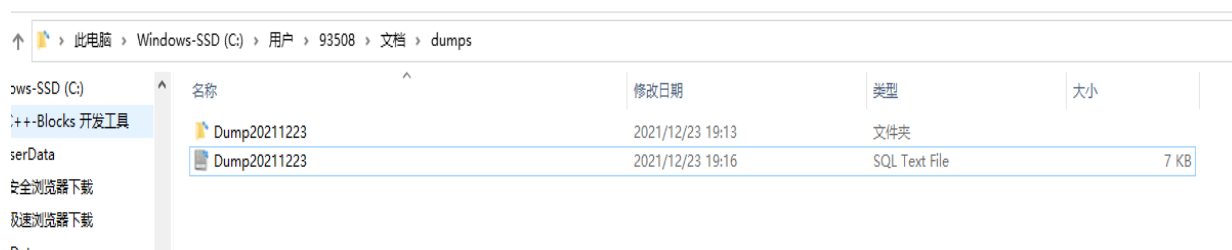
↑ > 此电脑 > Windows-SSD (C:) > Program Files > MySQL > MySQL Server 5.7 > bin

名称	修改日期	类型	大小
innochecksum	2021/6/7 9:32	应用程序	5,387 KB
jxgl_version1	2021/12/21 22:13	SQL Text File	9 KB
jxgl_version2	2021/12/21 22:13	SQL Text File	9 KB
jxgl_version3	2021/12/21 22:14	SQL Text File	4,439 KB
libmecab.dll	2021/4/21 11:55	应用程序扩展	1,797 KB
lz4_decompress	2021/6/7 9:32	应用程序	189 KB
my	2021/12/21 21:34	配置设置	14 KB

(2) MySQL Workbench 图形化界面



执行后同样可在对应文件夹下发现备份文件。



完整恢复：

(1) MySQL 脚本：

恢复语句为：

```
mysql -u root -p jxgl < jxgl_version1.sql

mysql -u root -p jxgl < jxgl_version2.sql

mysql -u root -p --default-character-set=utf8 jxgl < jxgl_version3.sql
```

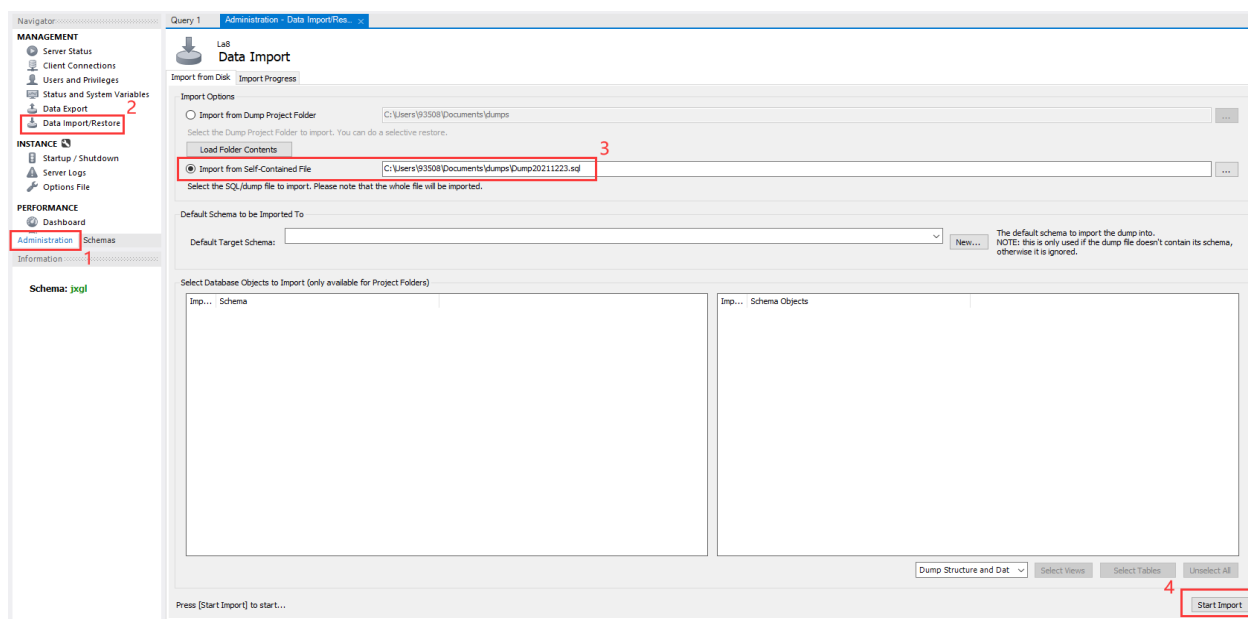
```
C:\WINDOWS\system32>cd C:\Program Files\MySQL\MySQL Server 5.7\bin
C:\Program Files\MySQL\MySQL Server 5.7\bin>mysql -u root -p jxgl < jxgl_version1.sql
Enter password: *****
C:\Program Files\MySQL\MySQL Server 5.7\bin>mysql -u root -p jxgl < jxgl_version2.sql
Enter password: *****
C:\Program Files\MySQL\MySQL Server 5.7\bin>mysql -u root -p jxgl < jxgl_version3.sql
Enter password: *****
ERROR at line 1249: Unknown command '?'.
C:\Program Files\MySQL\MySQL Server 5.7\bin>mysql -u root -p --default-character-set=utf8 jxgl < jxgl_version3.sql
Enter password: *****
C:\Program Files\MySQL\MySQL Server 5.7\bin>_
```

注意，以上语句需在 cmd 管理员模式下执行。

其中对于 jxgl_version3 的备份恢复，由于字符集的问题，需要在导入命令中添加：

--default-character-set=utf8

(2) MySQL Workbench 图形化界面



12.3 对教学管理系统与 2011 年 6 月 29 日上午 9: 00: 00 进行了差异备份；上午 9: 40 数据库发生故障，根据其差异备份和日志文件进行位置恢复。

首先需要再次修改 my.ini 文件，修改步骤如下：

1. 文件夹中输入 %ProgramData% 并定位到 C:\ProgramData\MySQL\MySQL Server 5.7

2. 修改 my.ini 文件，设置 log-bin 的保存目录

```
# ***** Group Replication Related *****  
# Specifies the base name to use for binary log files. With binary logging  
# enabled, the server logs all statements that change data to the binary  
# log, which is used for backup and replication.  
# log-bin  
log-bin="C:/Program Files/MySQL/MySQL Server 5.7/bin/jxgl.log"
```

3. 将 my.ini 文件复制到之前的位置 C:\Program Files\MySQL\MySQL Server 5.7

4. 复制完成后，打开 cmd 输入 services.msc，找到 MySQL 服务，右键重启（重启后在 log-bin 对应的文件夹 C:\Program Files\MySQL\MySQL Server 5.7\bin 中会出现 jx

gl.index 和 jxgl.000001 两个日志文件，可将其删除，便于后续操作)

之后的操作步骤如下：

(1) 使用 mysqldump -F (flush-logs) 备份数据库

```
mysqldump -u root -p jxgl -F > jxglf.sql
```

```
C:\WINDOWS\system32>cd C:\Program Files\MySQL\MySQL Server 5.7\bin
C:\Program Files\MySQL\MySQL Server 5.7\bin>mysqldump -u root -p jxgl -F > jxglf.sql
Enter password: *****
```

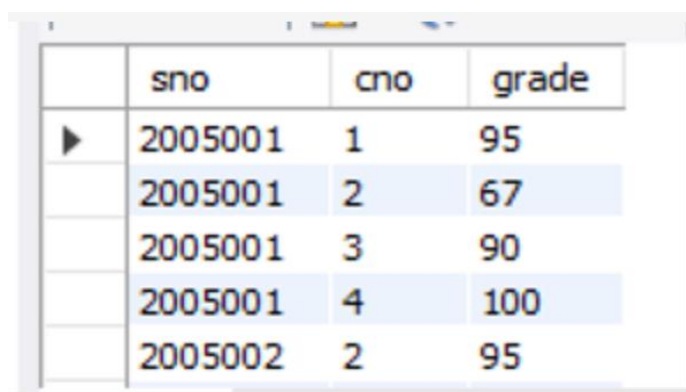
注意，以上语句需在 cmd 管理员模式下执行。

执行后发现 log-bin 对应的文件夹 C:\Program Files\MySQL\MySQL Server 5.7\bin 中再次出现 jxgl.index 和 jxgl.000001 两个日志文件。

(2) 在 MySQL Workbench 中执行增删改 (insert, delete, update) 的操作，本次实验中执行操作如下：

```
insert into sc values('2005001','4',100);
update sc set grade = 95 where sno = '2005001' and cno = '1';
```

执行后 sc 表变化如下：



	sno	cno	grade
▶	2005001	1	95
	2005001	2	67
	2005001	3	90
	2005001	4	100
	2005002	2	95

(3) 生成新的日志文件

```
mysqladmin -u root -p flush-logs
```

```
C:\WINDOWS\system32>cd C:\Program Files\MySQL\MySQL Server 5.7\bin
C:\Program Files\MySQL\MySQL Server 5.7\bin>mysqladmin -u root -p flush-logs
Enter password: *****
```


上述命令在 cmd 管理员模式下执行。

执行后会在 log-bin 对应的文件夹 C:\Program Files\MySQL\MySQL Server 5.7\bin 中发现有新的日志文件 jxgl.000002

(4) 执行 mysql 导入备份文件

```
mysql -uroot -p jxgl < jxglf.sql
```

```
C:\Program Files\MySQL\MySQL Server 5.7\bin>mysql -uroot -p jxgl < jxglf.sql
Enter password: *****
```

上述命令在 cmd 管理员模式下执行。

(5) 查看日志文件

```
mysqlbinlog -v jxgl.000001
```

```
C:\Program Files\MySQL\MySQL Server 5.7\bin>mysqlbinlog -v jxgl.000001
/*!50530 SET @@SESSION.PSEUDO_SLAVE_MODE=1*/;
/*!50003 SET @OLD_COMPLETION_TYPE=@@COMPLETION_TYPE, COMPLETION_TYPE=0*/;
DELIMITER /*!*/;
# at 4
#211222 18:01:50 server id 1  end_log_pos 123 CRC32 0x44c1d7bb  Start: binlog v 4, server v 5.7.35-log created 211222 18:01:50
BINLOG '
jvfCYQ8BAAAAAdwAAAHsAAAAAAQANS43LjM1LWxyZwAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAEzgNAAGAgAEBAQEAgAAAXwAEGggAAAAICAgCAAAACgoKKioAEJQA
AbvXwUQ=
/*!*/;
# at 123
#211222 18:01:50 server id 1  end_log_pos 154 CRC32 0x2485f2a0  Previous-GTIDs
# [empty]
# at 154
#211222 18:02:08 server id 1  end_log_pos 219 CRC32 0xa249a897  Anonymous_GTID  last_committed=0  sequence_number=
#
```

上述命令在 cmd 管理员模式下执行。

执行后在日志中寻找两个 update 语句执行前后的位置号，如下图所示：

对于：

```
insert into sc values('2005001','4',100);
```

```
BEGIN
/*!*/;
# at 291
#211222 18:02:08 server id 1  end_log_pos 342 CRC32 0x75ab9fe3  Table_map: `jxgl`.`sc` mapped to number 144
# at 342
#211222 18:02:08 server id 1  end_log_pos 392 CRC32 0x0515162f  Write_rows: table id 144 flags: STMT_END_F
BINLOG '
oPfCYRMBAAAAmWAAAFYBAAAAAJAAAAAAAAEABGp4Z2wAAAnNjAAMPDwMEFAAUAAATjn6t1
oPfCYR4BAAAAmGAAAIgBAAAAAJAAAAAAAAEAAgAD//gHMjAwNTAwMQEOZAAAACSWFQU=
/*!*/;
### INSERT INTO `jxgl`.`sc`
### SET
###   @1='2005001'
###   @2='4'
###   @3=100
# at 392
#211222 18:02:08 server id 1  end_log_pos 423 CRC32 0x5c50e3c9  Xid = 632
COMMIT/*!*/;
# at 423
```

对于:

```
update sc set grade = 95 where sno = '2005001' and cno = '1';

BEGIN
/*!*/;
# at 560
#211222 18:02:19 server id 1 end_log_pos 611 CRC32 0xb04c5bf8 Table_map: `jxgl`.`sc` mapped to number 144
# at 611
#211222 18:02:19 server id 1 end_log_pos 677 CRC32 0x4c29e899 Update_rows: table id 144 flags: STMT_END_F

BINLOG '
q/fCYRMBAAAAmWAAAGMCAAAAAJAAAAAAAAEABGp4Z2wAAAnJAAMPDwMEFAAUAT4W0yw
q/fCYR8BAAAAQgAAAKUCAAAAAJAAAAAAAAEAAGAD//4BzIwMDUwMDEBMVcAAAD4BzIwMDUwMDEB
MVSAAACZ6C1M
'/*!*/;
### UPDATE `jxgl`.`sc`
### WHERE
###   @1='2005001'
###   @2='1'
###   @3=87
### SET
###   @1='2005001'
###   @2='1'
###   @3=95
# at 677
#211222 18:02:19 server id 1 end_log_pos 708 CRC32 0x2215d9c1 Xid = 634
COMMIT/*!*/;
# at 708
```

在这里我们选择跳过 insert 语句，但仍然执行 update 语句来进行位置恢复。

```
mysqlbinlog --no-defaults --stop-position=291 jxgl.000001|mysql -u root -h localhost -p
```

```
mysqlbinlog --no-defaults --start-position=423 jxgl.000001|mysql -u root -h localhost -p
```

```
C:\Program Files\MySQL\MySQL Server 5.7\bin>mysqlbinlog --no-defaults --stop-position=291 jxgl.000001|mysql -u root -h localhost -p
Enter password: *****

C:\Program Files\MySQL\MySQL Server 5.7\bin>mysqlbinlog --no-defaults --start-position=423 jxgl.000001|mysql -u root -h localhost -p
Enter password: *****
```

上述命令在 cmd 管理员模式下执行。

执行后，在 MySQL Workbench 中查询 sc:

	sno	cno	grade
▶	2005001	1	95
	2005001	2	67
	2005001	3	90
	2005002	2	95
	2005003	3	88
	2005010	1	87

发现成功跳过了 insert 语句，并继续执行 update 语句，位置恢复成功！