

### 3 SQL 数据完整性

实验目的：

1. 熟悉通过 SQL 进行数据完整性控制的方法。

实验平台：

1. 数据库管理系统：mySQL 或其它。

实验内容和要求：

1. 定义若干表，其中包括 primary key, foreign key 和 check 的定义。
2. 让表中插入数据，考察 primary key 如何控制实体完整性。
3. 删除被引用表中的行，考察 foreign key 中 on delete 子句如何控制参照完整性。
4. 修改被引用表中的行的 primary key，考察 foreign key 中 on update 子句如何控制参照完整性。
5. 修改或插入表中数据，考察 check 子句如何控制校验完整性。
6. 定义一个 trigger，并通过修改表中数据考察触发器如何起作用。

实验三过程和实验结果：

(由学生补充完成，并加上每步实验的执行的 sql 语句和执行结果的截图)

1. 创建 library 数据库，在数据库中创建数据表

```
book (bno char(8) ,
      category char(10),
      title varchar(40),
      press varchar(30),
      year int,
      author varchar(20),
      price decimal(7,2),
      total int ,
      stock int)
```

主键：bno



```
MINGW64:/c:/Users/Lenovo
+-----+
| Tables_in_library |
+-----+
| book               |
+-----+
1 row in set (0.04 sec)

mysql> select *
-> from book;
Empty set (0.01 sec)

mysql> drop table book;
Query OK, 0 rows affected (0.06 sec)

mysql> show tables
-> ;
Empty set (0.00 sec)

mysql> create table book (bno char(8) ,
->                        category char(10),
->                        title varchar(40),
->                        press varchar(30),
->                        year int,
->                        author varchar(20),
->                        price decimal(7,2),
->                        total int ,
->                        stock int,
->                        primary key(bno));
Query OK, 0 rows affected (0.06 sec)

mysql> |
```

2. 编写创建数据表

card(

```
cno char(7),
name varchar(10),
department varchar(40),
type char(1))
```

主键: cno

type 字段 in ('T','G','U','O');

提示: type 字段可以定义为 enum 类型

```
mysql> show tables
-> ;
Empty set (0.00 sec)

mysql> create table book (bno char(8) ,
-> category char(10),
-> title varchar(40),
-> press varchar(30),
-> year int,
-> author varchar(20),
-> price decimal(7,2),
-> total int ,
-> stock int,
-> primary key(bno));
Query OK, 0 rows affected (0.06 sec)

mysql> create table card(
-> cno char(7),
-> name varchar(10),
-> department varchar(40),
-> type char(1),
-> primary key(cno),
-> check(type in ('T', 'G', 'U', 'O'))
-> )
-> ;
Query OK, 0 rows affected (0.06 sec)
```

### 3. 编写创建数据表

```
borrow(
cno char(7),
bno char(8),
borrow_date datetime,
return_date datetime,)
```

外键: bno, cno。 分别对应 book 和 card 的 bno 和 cno, 同时外键 bno 级联删除, cno 级联更新

```
mysql> create table borrow(
-> cno char(7),
-> bno char(8),
-> borrow_date datetime,
-> return_date datetime,
-> foreign key (bno)
-> references book(bno)
-> on delete cascade,
-> foreign key (cno)
-> references card(cno)
-> on update cascade
-> ) ;
Query OK, 0 rows affected (0.07 sec)

mysql> show tables
-> ;
+-----+
| Tables_in_library |
+-----+
| book              |
| borrow            |
| card              |
+-----+
3 rows in set (0.00 sec)

mysql>
```

### 4. 插入基本数据

Book 表('10', '心理学','新的世界','浙江大学',2002,'高云鹏',48.00,20,4)

Card 表('zju1001','雷锋','计算机','T');

Borrow 表('zju1001','10','2011-09-10','2012-2-23');

```
mysql> insert into book
-> values('10', '心理学','新的世界','浙江大学',2002,'高云鹏',48.00,20,4);
Query OK, 1 row affected (0.01 sec)

mysql>
mysql> insert into card
-> values('zju1001','雷锋','计算机','T');
Query OK, 1 row affected (0.03 sec)

mysql>
```

```
mysql> insert into borrow
-> values ('zju1001','10','2011-09-10','2012-2-23');
Query OK, 1 row affected (0.03 sec)

mysql> select *
-> from borrow
-> ;
+-----+-----+-----+-----+
| cno   | bno   | borrow_date      | return_date      |
+-----+-----+-----+-----+
| zju1001 | 10    | 2011-09-10 00:00:00 | 2012-02-23 00:00:00 |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select *
-> from book;
+-----+-----+-----+-----+-----+-----+-----+-----+
| bno | category | title   | press | year | author | price | total | stock |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 10  | 心理学   | 新的世界 | 浙江大学 | 2002 | 高云鹏 | 48.00 | 20    | 4      |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select *
-> from card;
+-----+-----+-----+-----+
| cno   | name | department | type |
+-----+-----+-----+-----+
| zju1001 | 雷锋 | 计算机     | T    |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

5. 向 book 表, 再次插入一条记录, 看看数据库的提示。

('10', '生物学','物种起源','哈佛大学',2002,'Darwin',48.00,20,4)

```
mysql> insert into book
-> values('10', '生物学','物种起源','哈佛大学',2002,'Darwin',48.00,20,4);
ERROR 1062 (23000): Duplicate entry '10' for key 'book.PRIMARY'
mysql>
```

6. 向 card 表添加一条记录, 看看数据库的提示。

('zju1002','凤姐','心理学','K')

```
mysql> insert into card
-> values ('zju1002','凤姐','心理学','K');
ERROR 3819 (HY000): Check constraint 'card_chk_1' is violated.
mysql>
```

7. 删除 card 表中的记录, 观察数据库的提示。

```
mysql> delete from card;
mysql>
```

8. 更新 Book 表中‘新的世界’ bno 为‘100’ 看看 borrow 表中的记录

```
mysql> update book
-> set bno = '100'
-> where title = '新的世界';
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`library`.`borrow`, CONSTRAINT
`borrow_ibfk_1` FOREIGN KEY (`bno`) REFERENCES `book` (`bno`) ON DELETE CASCADE)
mysql> select *
-> from book;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| bno | category | title | press | year | author | price | total | stock |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 10 | 心理学 | 新的世界 | 浙江大学 | 2002 | 高云鹏 | 48.00 | 20 | 4 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>
```

## 9. 编写触发器

作用是一个借书证不能同时借三本书（简化一点，认为借书记录中有记录即为借了书）。大家尝试编写下先验（before）触发器，作为对比，下面是后验触发器的例子（例子为 sql server 语法，同学们的答案可以是 mysql）。

```
create trigger st1 on borrow
after insert
as
Begin
    Declare @num int
    Declare @cno char(10)
    select @cno = cno from inserted
    select @num=count(*) from borrow where cno=@cno
    If(@num>2)
    begin
        Print('一位同学一个学期不可以借三本书! ')
        Rollback
    End
End
```

```
mysql> delimiter $
mysql> create trigger before_borrow_insert
-> before insert on borrow for each row
-> begin
-> declare num int;
-> declare c char(10);
-> select cno from inserted into c;
-> select count(*) from borrow where cno=c into num;
-> if (num = 2) then signal sqlstate '45000'
-> set message_text = '一位同学一个学期不可以借三本书! ';
-> end if;
-> end$
Query OK, 0 rows affected (0.02 sec)

mysql> delimiter ;
mysql>
```

## 10. 实验总结及思考

1. 触发器的语法规则比较复杂，需要进一步的学习。
2. 确定主键之后，主键的值不能重复