

数据库系统实验报告

课程名称： 数据库系统原理 实验类型： 上机

实验项目名称： 通过 SQL 进行数据完整性控制

学生姓名： 王子腾 专业： 软件工程 学号： 3180102173

同组学生姓名： 李想、潘凯航、求昊泽、杨锐 指导老师： 陈岭

实验地点： 个人电脑 实验日期： 2020 年 3 月 27 日

一、 实验目的和要求

熟悉通过 SQL 进行数据完整性控制的方法。

二、 实验内容和原理

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。

三、 主要仪器设备

1. 操作系统： windows 10
2. DBMS： MySQL 8.0.18
3. 图形化界面： Navicat Premium 12

四、 操作方法与实验步骤

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

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')

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

```

1  create database library;
2  use library;
3  create table book (
4  bno char(8),
5  category char(10),
6  title varchar(40),
7  press varchar(30),
8  year int,
9  author varchar(20),
10 price decimal(7,2),
11 total int ,
12 stock int,
13 primary key(bno));
14
15 create table card(
16 cno char(7),
17 name varchar(10),
18 department varchar(40),
19 type char(1),
20 primary key (cno),
21 check (type in ('T','G','U','O')) );
22
23 create table borrow(
24 cno char(7),
25 bno char(8),
26 borrow_date datetime,
27 return_date datetime,
28 foreign key(bno) references book(bno) on delete cascade,
29 foreign key(cno) references card(cno) on update cascade);
30

```

信息

状态

```

create database library
. 1007 - Can't create database 'library'; database exists
. 时间: 0.075s

```

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');

```

1  insert into book
2  values('10', '心理学', '新的世界', '浙江大学', 2002, '高云鹏',
3  ,48.00, 20, 4);
4
5  insert into card
6  values('zju1001', '雷锋', '计算机', 'T');
7
8  insert into borrow
9  values('zju1001', '10', '2011-09-10', '2012-2-23');

```

```
insert into book
values('10', '心理学', '新的世界', '浙江大学', 2002, '高云鹏', 48.00, 20, 4)
> Affected rows: 1
> 时间: 0.067s
```

```
insert into card
values('zju1001', '雷锋', '计算机', 'T')
> Affected rows: 1
> 时间: 0.059s
```

```
insert into borrow
values('zju1001', '10', '2011-09-10', '2012-2-23')
> Affected rows: 1
> 时间: 0.087s
```

- 5) 向 **book** 表，再次插入一条记录，看看数据库的提示，提出可行的修改方案。

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

```
10 insert into book
11 values('10', '生物学', '物种起源', '哈佛大学', 2002, 'Darwin'
,48.00,20,4);
```

```
insert into book
values('10', '生物学', '物种起源', '哈佛大学', 2002, 'Darwin', 48.00, 20, 4)
> 1062 - Duplicate entry '10' for key 'PRIMARY'
> 时间: 0.001s
```

修改方案:

由于 primary key 重复，因此无法插入，可将第一个元素'10'修改为其他值，或使用 update 更新 bno=10的这一行其他属性为新值。

- 6) 向 **card** 表添加一条记录，看看数据库的提示，并提出可行的修改方案。

```
('zju1002', '东野圭吾', '心理学', 'K'),
```

```
13 insert into card
14 values('zju1002', '东野圭吾', '心理学', 'K');
15
```

```
insert into card
values('zju1002','东野圭吾','心理学','K')
> 3819 - Check constraint 'card_chk_1' is violated.
> 时间: 0s
```

修改方案:

由于 type 是限定在 ('T' , 'G' , 'U' , 'O') 中取值, 由于给定的值 'K' 不属于该范围, check 语句会报错, 因此无法插入。可将 'K' 值更换为限定值之一, 完成插入。

7) 删除 card 表中的某条记录, 观察数据库的提示。

```
16 delete from card;
```

```
delete from card
> 1451 - Cannot delete or update a parent row: a foreign key
constraint fails (`library`.`borrow`, CONSTRAINT `borrow_ibfk_2`
FOREIGN KEY (`cno`) REFERENCES `card` (`cno`) ON UPDATE CASCADE)
> 时间: 0.454s
```

8) 更新 Book 表中 '新的世界' bno 为 '100', 观察数据库的提示。

```
18 update book
19 set bno = '100'
20 where title='新的世界';
```

```
update book
set bno = '100'
where title='新的世界'
> 1451 - 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)
> 时间: 0.289s
```

9) 编写触发器

检查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)。

```
22 delimiter $
23 create trigger borrow_limit
24 after insert on borrow
25 for each row
26 begin
27 declare cnt int;
28 select count(tt.cno) into cnt from(select cno from
   borrow where borrow.cno = inserted.cno) tt;
29 if cnt>2 then delete from borrow where borrow.bno =
   inserted.bno;
30 end if;
31 end;
32 $
33 delimiter;
```

五、 总结与思考

本次实验中，加强了对外键，主键相关约束的考察，在最后一题中，我对于触发器的编写仍然有不熟练的情况，如不同类型的触发器写法，存在逻辑不清晰，语法错误等问题，其中 row 或者 inserted (deleted) 含义，if-else 语句，case-when 语句等用法，在日后的学习当中，还应当继续加强对 sql 编程思想的理解，同时多了解相关函数有助于在编程中事半功倍。