中山大学数据科学与计算机学院本科生实验报告

(2020学年秋季学期)

课程名称：数据库系统　　任课教师：刘玉葆

年级&班级	18级 计科一班	专业(方向)	计科（大数据与人工智能方向）
学号	18340014	姓名	陈嘉宁
电话	18475934419	Email	734311072@qq.com
开始日期	2020/12/1	完成日期	2020/12/1

一、实验题目

SQL参照完整性

二、实验目的

1. 学习建立外键，以及利用**FOREIGN KEY...REFERENCES子句**以及各种约束保证参照完整性。

三、实验内容

1. 不违反参照完整性的插入数据示例；
2. 违反参照完整性的插入数据示例；
3. 级联删除；
4. 两张表的互相参照问题。

四、实验过程

1. **按照实验实例完成准备工作：**
    - 建立表Stu_Union：

SQLQuery1.sql - (l...KLS675B\dell (55))*

```
create table Stu_Union(
    sno char(5) not null unique,
    sname char(8),
    ssex char(1),
    sage int,
    sdept char(20),
    constraint PK_Stu_Union primary key (sno)
);
insert into Stu_Union values ('10001','李勇','0',24,'EE');
insert into Stu_Union values ('95002','王敏','1',23,'CS');
insert into Stu_Union values ('95003','王浩','0',25,'EE');
insert into Stu_Union values ('95005','王杰','0',25,'EE');
insert into Stu_Union values ('95009','李勇','0',25,'EE');
select * from Stu_Union;
```

结果 消息

	sno	sname	ssex	sage	sdept
1	10001	李勇	0	24	EE
2	95002	王敏	1	23	CS
3	95003	王浩	0	25	EE
4	95005	王杰	0	25	EE
5	95009	李勇	0	25	EE

- 建立表Course：

SQLQuery1.sql - (l...KLS675B\dell (55))*

```
create table Course(
    cno char(4) not null unique,
    cname varchar(50) not null,
    cpoints int,
    constraint PK primary key (cno)
);
insert Course values ('0001','ComputerNetworks',2);
insert Course values ('0002','Database',3);
select * from Course
```

结果 消息

	cno	cname	cpoints
1	0001	ComputerNetworks	2
2	0002	Database	3

- 建立表SC：

SQLQuery1.sql - (l...KLS675B\dell (55))*

```
create table SC(
sno char(5) references Stu_Union(sno) on delete cascade,
cno char(4) references Course(cno) on delete cascade,
grade INT,
constraint PK_SC primary key (sno,cno)
);
insert into SC values ('95002','0001',2);
insert into SC values ('95002','0002',2);
insert into SC values ('10001','0001',2);
insert into SC values ('10001','0002',2);
select * from SC;
```

结果 消息

	sno	cno	grade
1	10001	0001	2
2	10001	0002	2
3	95002	0001	2
4	95002	0002	2

- 尝试插入位数不符要求的主键：

```
SQLQuery1.sql - (l...KLS675B\dell (55))*
insert into SC values ('99','0001',2);
```

消息

```
消息 547，级别 16，状态 0，第 1 行
INSERT 语句与 FOREIGN KEY 约束"FK__SC__sno__48CFD27E"冲突。该冲突发生于数据库"School"
语句已终止。
```

- 尝试删除：

```
SQLQuery1.sql - (l...KLS675B\dell (55))*
delete from Stu_Union where sno='10001';
select * from SC;
```

结果 消息

	sno	cno	grade
1	95002	0001	2
2	95002	0002	2

- 建立表Stu_Card：

```
SQLQuery1.sql - (l...KLS675B\dell (55))*
create table Stu_Card(
    card_id char(14),
    stu_id char(10) references students(sid) on delete cascade,
    remained_money decimal(10,2),
    constraint PK_stu_card primary key (card_id)
);
insert into Stu_Card values ('05212567','800001216',100.25);
insert into Stu_Card values ('05212222','800005753',200.50);
select * from Stu_Card
```

结果 消息

	card_id	stu_id	remained_money
1	05212222	800005753	200.50
2	05212567	800001216	100.25

- 建立表ICBC_Card：

```
SQLQuery1.sql - (l...KLS675B\dell (55))*
create table ICBC_Card(
    bank_id char(20),
    stu_chrd_id char(14) references stu_card(card_id) on delete cascade,
    restored_money decimal(10,2),
    constraint PK_Icbc_card primary key (bank_id)
);
insert into ICBC_Card values ('9558844022312','05212567',15000.1);
insert into ICBC_Card values ('9558844023645','05212222',50000.3);
select * from ICBC_Card;
```

结果 消息

	bank_id	stu_chrd_id	restored_money
1	9558844022312	05212567	15000.10
2	9558844023645	05212222	50000.30

- 尝试删除students表中数据，但该项作为外键（on delete no action）不可被删除：

```
SQLQuery1.sql - (l...KLS675B\dell (55))*
delete from students where sid='800001216';
```

消息

```
消息 547，级别 16，状态 0，第 1 行
DELETE 语句与 REFERENCE 约束"FK_CHOICES_STUDENTS"冲突。该冲突发生于数据库"School"，
语句已终止。
```

- 将上述约束更改为on delete cascade：

```
SQLQuery1.sql - (l...KLS675B\dell (55))*
alter table choices drop [FK_CHOICES_STUDENTS];
alter table choices add
constraint [FK_CHOICES_STUDENTS] foreign key(
    [sid])
    references [dbo].[STUDENTS](
        [sid])
        on delete cascade;
delete from students where sid='800001216';
select * from stu_card;
select * from icbc_card;
```

结果 消息

	card_id	stu_id	remained_money
1	05212222	800005753	200.50

	bank_id	stu_chrd_id	restored_money
1	9558844023645	05212222	50000.30

- 将约束改为on delete no action：

```
SQLQuery1.sql - (l...KLS675B\dell (55))*
alter table ICBC_Card drop constraint FK__ICBC_Card__stu_c__4F7CD00D;
alter table ICBC_Card
    add constraint FK__ICBC_Card foreign key (stu_chrd_id)
        references Stu_card(card_id) on delete no action;
```

消息

命令已成功完成。

- 尝试删除students表中数据，但该项作为外键（on delete no action）不可被删除：

```
SQLQuery1.sql - (l...KLS675B\dell (55))*
begin transaction del
delete from students where sid='800005753';
select * from stu_card;
select * from icbc_card;
commit transaction del
```

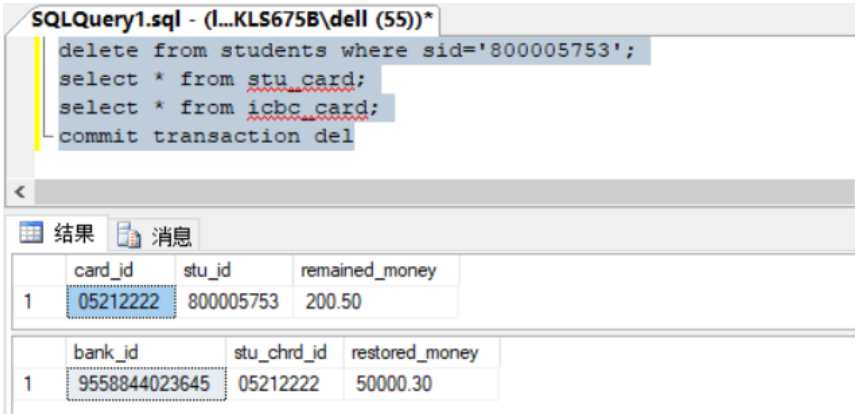
结果 消息

消息 547，级别 16，状态 0，第 2 行
DELETE 语句与 REFERENCE 约束"FK__ICBC_Card"冲突。该冲突发生于数据库"School"
语句已终止。

(1 行受影响)

(1 行受影响)

- 而尝试删除on delete cascade 的约束则可以实现级联删除：

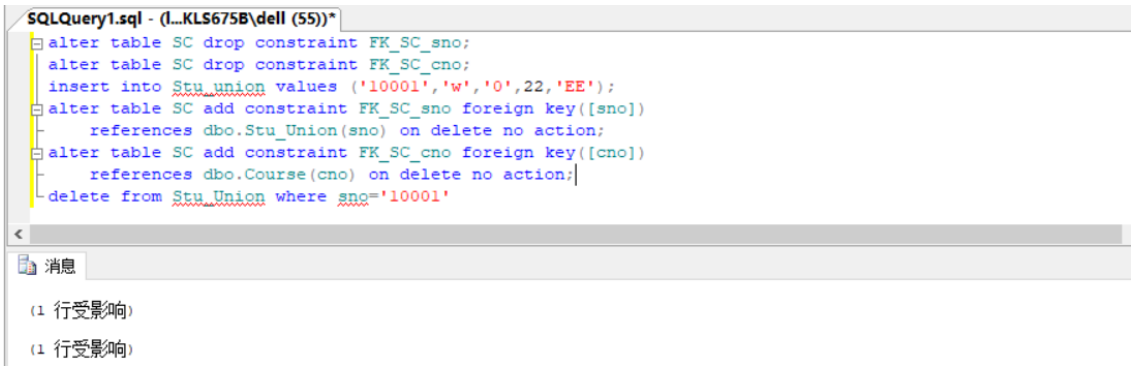


2. **用alter table语句将SC表中的on delete cascade改为on delete no action,重新插入SC的数据（按照实验一）。再删除Stu_Union中sno为'10001'的数据。观察结果，并分析原因：**

   执行的语句为：

```
alter table SC drop constraint FK_SC_sno;
alter table SC drop constraint FK_SC_cno;
insert into Stu_union values ('10001','w','0',22,'EE');
alter table SC add constraint FK_SC_sno foreign key([sno])
    references dbo.Stu_Union(sno) on delete no action;
alter table SC add constraint FK_SC_cno foreign key([cno])
    references dbo.Course(cno) on delete no action;
delete from Stu_Union where sno='10001'
```

   按照SQL语言的规则来说，代码块中最后面的delete语句不应该成功执行，因为它试图删除有on delete no action约束的外键，但在实验过程中，这条语句成功地执行了：



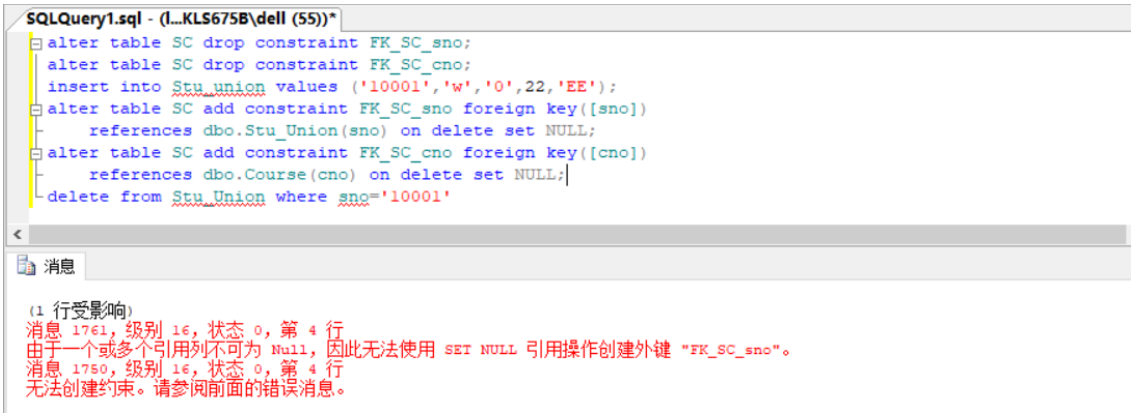
   具体原因尚不明确。

3. **用alter table语句将SC表中的on delete no action改为on delete set NULL,重新插入SC的数据（按照实验一）。再删除Stu_Union中sno为'10001'的数据。观察结果，并分析原因：**

   执行的语句为：

```
alter table SC drop constraint FK_SC_sno;
alter table SC drop constraint FK_SC_cno;
insert into Stu_union values ('10001','w','0',22,'EE');
alter table SC add constraint FK_SC_sno foreign key([sno])
    references dbo.Stu_Union(sno) on delete set NULL;
alter table SC add constraint FK_SC_cno foreign key([cno])
    references dbo.Course(cno) on delete set NULL;
delete from Stu_Union where sno='10001'
```

执行结果为：



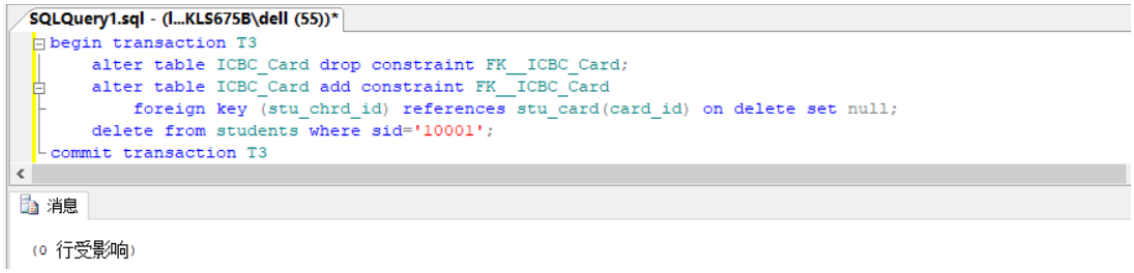
由于sno和cno均定义为not null的变量，因此on delete set null不能执行。

4. **建立事务T3，修改ICBC_Card表的外键属性，使其变为on delete set NULL,尝试删除students表中一条记录。观察结果，并分析原因：**

   执行的语句为：

```
begin transaction T3
    alter table ICBC_Card drop constraint FK__ICBC_Card;
    alter table ICBC_Card add constraint FK__ICBC_Card
        foreign key (stu_chrd_id) references stu_card(card_id) on delete
set null;
    delete from students where sid='10001';
commit transaction T3
```

   按照SQL语言的规则来说，这一条delete语句也不应该成功执行的。但在实验中又出现了令人意想不到的情况：



5. **创建一个班里的学生互助表，规定：包括学生编号，学生姓名，学生的帮助对象，每个学生有且仅有一个帮助对象，帮助对象也必须是班里的学生：**

   执行的语句为：

```
create table helper(
    hsid char(10),
    hsname char(10),
    h2sid char(10),
    constraint PK_hsid primary key (hsid)
);
create table behelper(
    bsid char(10),
    bsname char(10),
    bfsid char(10),
    constraint PK_bsid primary key (bsid)
```

```
);
alter table helper add constraint FK_h2sid foreign key (h2sid)
    references behelper(bsid) on delete no action;
alter table behelper add constraint FK_bfsid foreign key (bfsid)
    references helper(hsid) on delete no action;
```

6. **学校学生会的每个部门都有一个部长，每个部长领导多个部员，每个部只有一个部员有评测部长的权利，请给出体现这两种关系（领导和评测）的两张互参照的表的定义：**

   执行的语句为：

```
create table leader(
    lid char(10),
    lname char(10),
    f2id char(10),
    dept char(20),
    constraint PK_lid primary key (lid,dept)
);
create table fellow(
    fid char(10),
    fname char(10),
    l1id char(10),
    dept char(20),
    constraint PK_fid primary key (fid,dept)
);
alter table leader add constraint FK_f2id foreign key (f2id,dept)
    references fellow(fid,dept) on delete no action;
alter table fellow add constraint FK_l1id foreign key (l1id,dept)
    references leader(lid,dept) on delete no action;
```

   但是在这个相互参照问题中，我没能表现出“**每个部长领导多个部员**”这一性质。