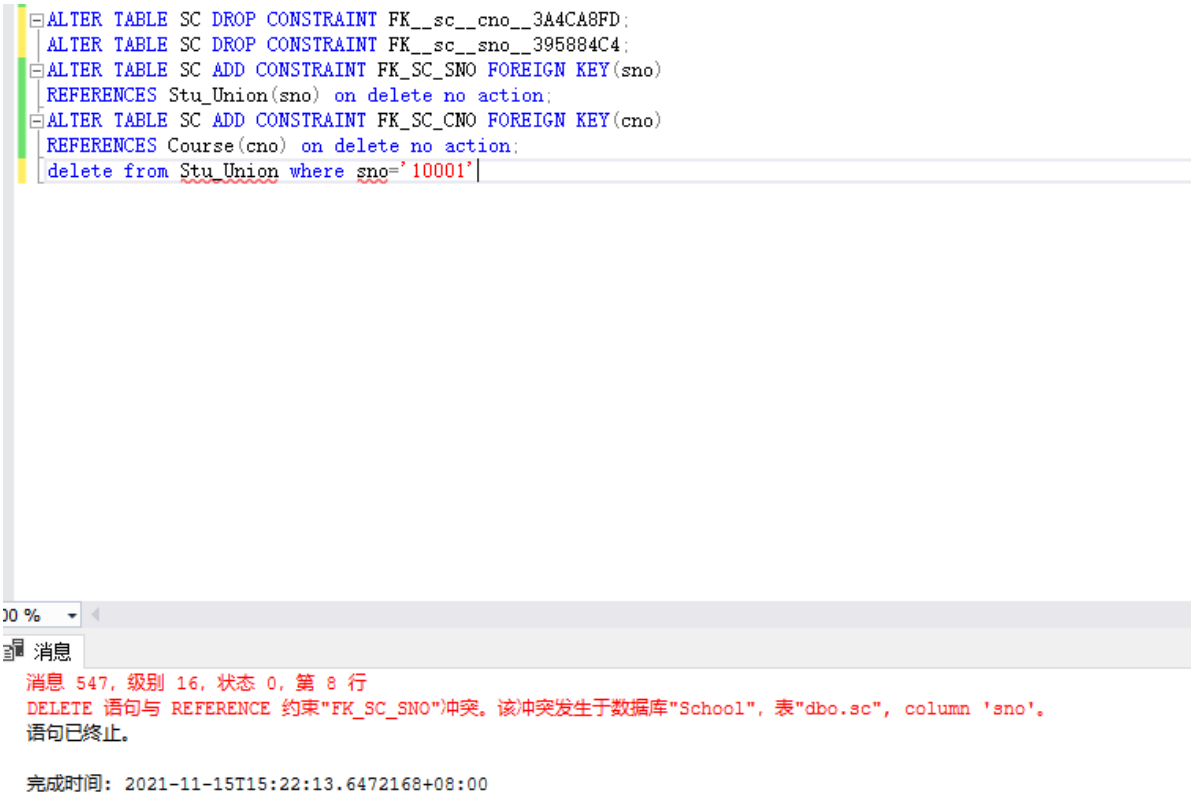
第十次实验

(1)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。

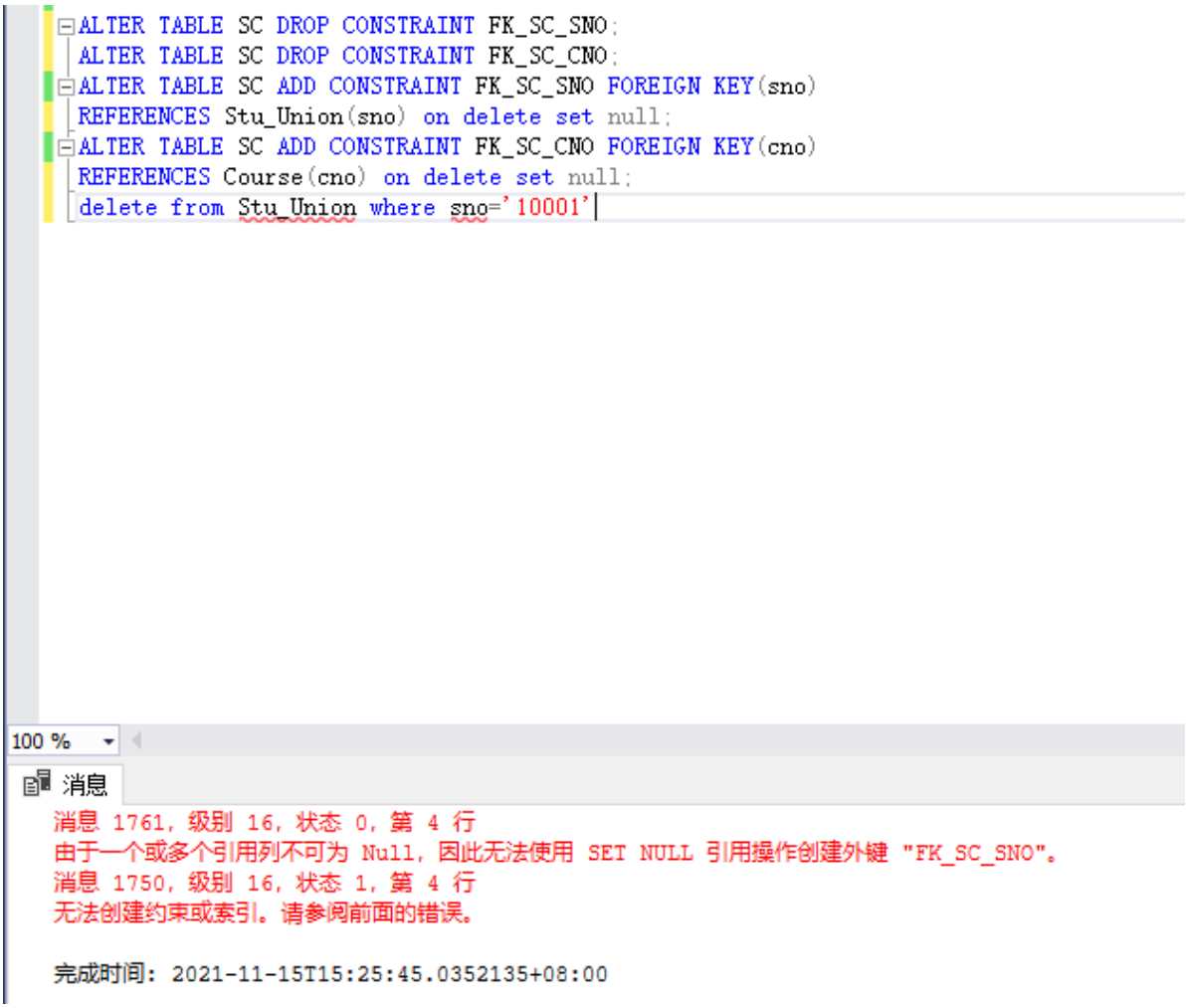
```
ALTER TABLE SC DROP CONSTRAINT FK__sc__cno__3A4CA8FD;
ALTER TABLE SC DROP CONSTRAINT FK__sc__sno__395884C4;
ALTER TABLE SC ADD CONSTRAINT FK_SC_SNO FOREIGN KEY(sno)
REFERENCES Stu_Union(sno) on delete no action;
ALTER TABLE SC ADD CONSTRAINT FK_SC_CNO FOREIGN KEY(cno)
REFERENCES Course(cno) on delete no action;
delete from Stu_Union where sno='10001'
```



on delete no action意味着当从表中有匹配的记录时，主表中相应的候选键不允许update/delete操作。此处SC表中的两个外键都改成了on delete no action，也就不能从主表Stu_Union中删除键值了。

(2)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。

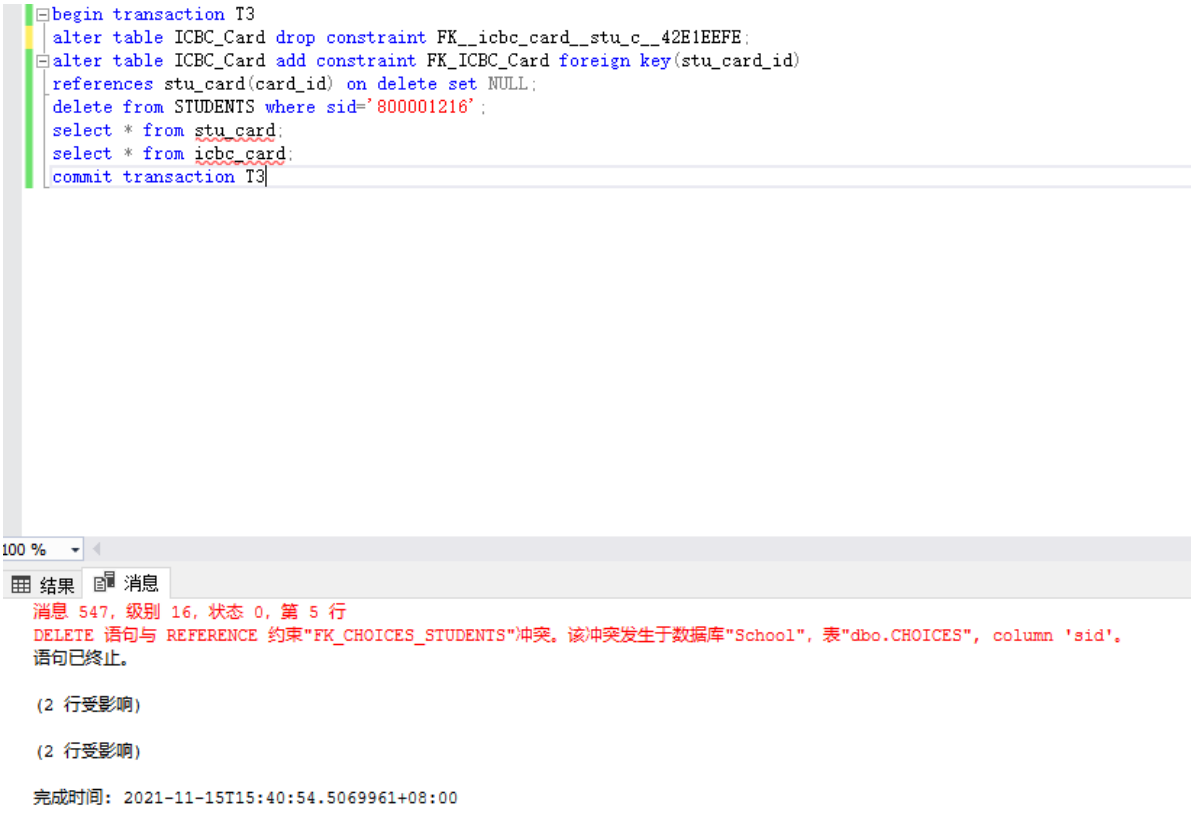
```
ALTER TABLE SC DROP CONSTRAINT FK_SC_SNO;
ALTER TABLE SC DROP CONSTRAINT FK_SC_CNO;
ALTER TABLE SC ADD CONSTRAINT FK_SC_SNO FOREIGN KEY(sno)
REFERENCES Stu_Union(sno) on delete set null;
ALTER TABLE SC ADD CONSTRAINT FK_SC_CNO FOREIGN KEY(cno)
REFERENCES Course(cno) on delete set null;
delete from Stu_Union where sno='10001'
```



因为Stu_Union表中的sno是非空NULL的，引用它的外键也就不能为NULL，不能设置为on delete set NULL。

(3)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。

```
begin transaction T3
alter table ICBC_Card drop constraint
FK__icbc_card__stu_c__42E1EEFE;
alter table ICBC_Card add constraint FK_ICBC_Card foreign
key(stu_card_id)
references stu_card(card_id) on delete set NULL;
delete from STUDENTS where sid='800001216';
select * from stu_card;
select * from icbc_card;
commit transaction T3
```



Choices使用了外键关联Students表，采用on delete no action。当从表中有匹配的记录时，主表中相应的候选键不允许update/delete操作。

(4)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。（表的自参照问题）

```
create table mutual_help(
    sid char(5),
    sname varchar(20),
    help_stu char(5),
    CONSTRAINT PK_help primary key(sid)
);
alter table mutual_help add constraint FK_help foreign key(help_stu)
references mutual_help(sid);
```

```
create table mutual_help(
    sid char(5),
    sname varchar(20),
    help_stu char(5),
    CONSTRAINT PK_help primary key(sid)
);
alter table mutual_help add constraint FK_help foreign key(help_stu)
references mutual_help(sid);
```

100 %

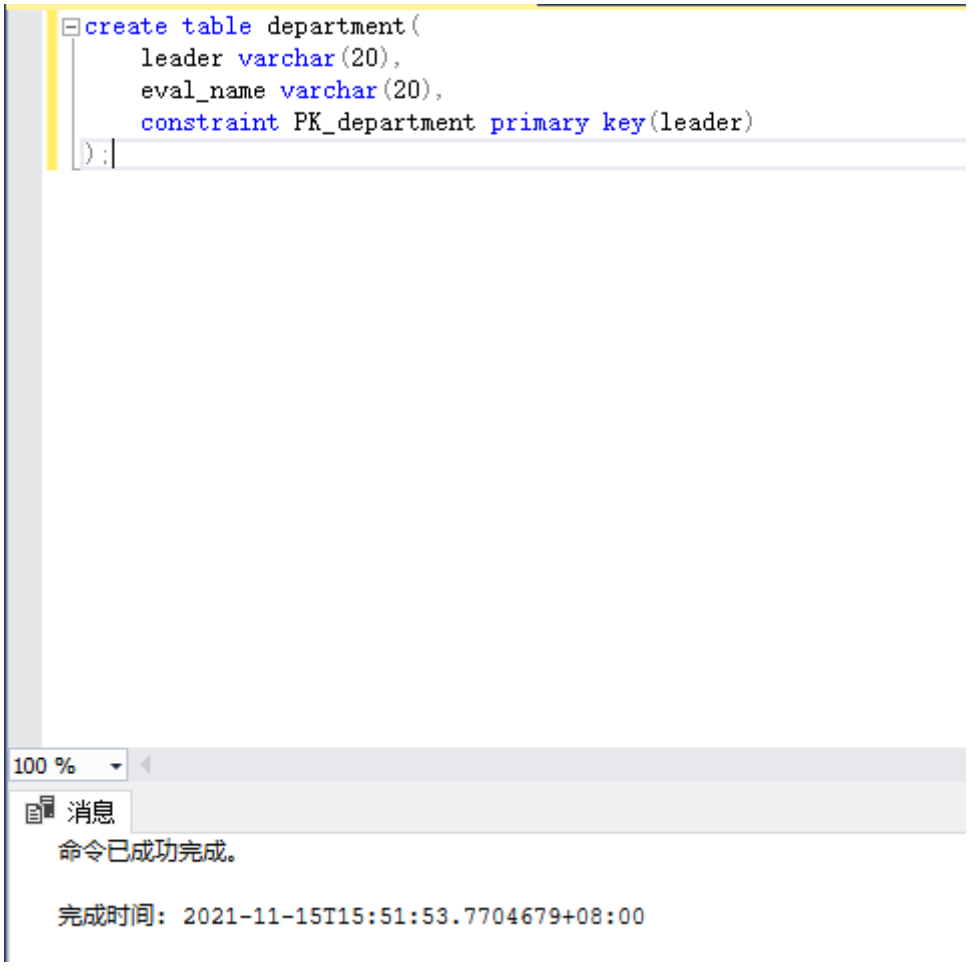
消息

命令已成功完成。

完成时间: 2021-11-15T15:48:00.6044261+08:00

(5)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。（两个表互相参照的问题）

```
create table department(
    leader varchar(20),
    eval_name varchar(20),
    constraint PK_department primary key(leader)
);
```




```
create table student(
    eval_name varchar(20),
    leader varchar(20),
    constraint PK_student primary key(eval_name),
    constraint FK_student foreign key(leader) references
department(leader)
);
alter table department add constraint FK_department foreign
key(eval_name)
references student(eval_name);
```

```sql
create table student(
    eval_name varchar(20),
    leader varchar(20),
    constraint PK_student primary key(eval_name),
    constraint FK_student foreign key(leader) references department(leader)
);
alter table department add constraint FK_department foreign key(eval_name)
references student(eval_name);
```

100 %

消息

命令已成功完成。

完成时间: 2021-11-15T15:54:15.2706271+08:00