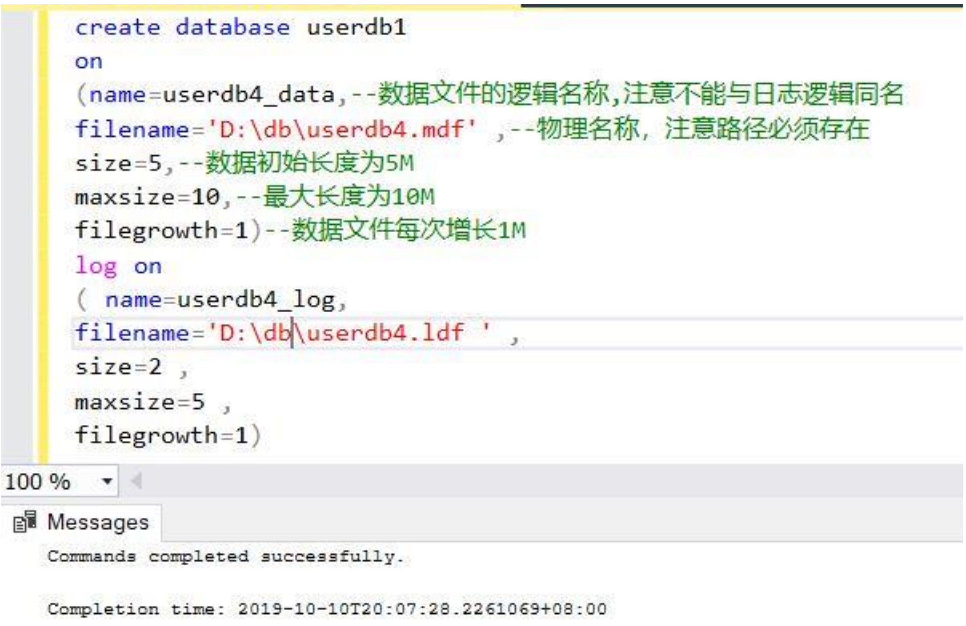
实验报告　第一次实验

Hollow Man

一、实验三

1. 问题 1

运行结果如下图所示:



```
create database userdb1
on
(name=userdb4_data,--数据文件的逻辑名称,注意不能与日志逻辑同名
filename='D:\db\userdb4.mdf' ,--物理名称, 注意路径必须存在
size=5,--数据初始长度为5M
maxsize=10,--最大长度为10M
filegrowth=1)--数据文件每次增长1M
log on
( name=userdb4_log,
filename='D:\db\userdb4.ldf ' ,
size=2 ,
maxsize=5 ,
filegrowth=1)
```

```
Commands completed successfully.

Completion time: 2019-10-10T20:07:28.2261069+08:00
```

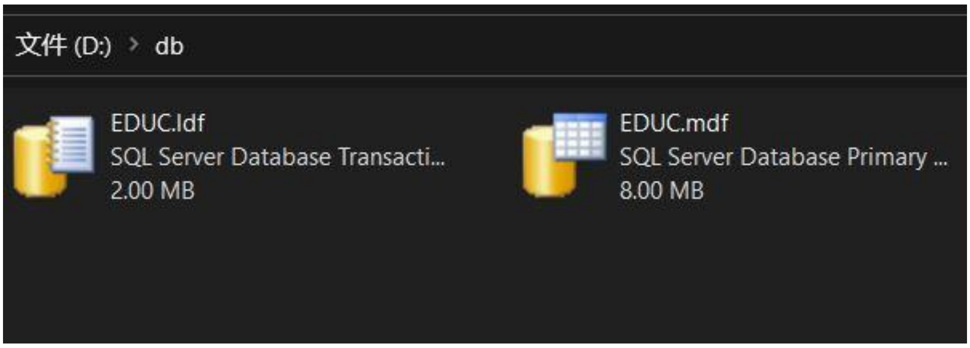
在这里，我修改了问题 1 提供的代码中的 filename，使其符合我的文件存贮路径。

2. 问题 2

SQL 语句如下所示:

```
create database EDUC on(
    name=EDUC_data,--数据文件的逻辑名称,注意不能与日志逻辑同名
    filename='D:\db\EDUC.mdf' ,--物理名称, 注意路径必须存在
    size=5,--数据初始长度为5M
    maxsize=10,--最大长度为10M
    filegrowth=1
    )--数据文件每次增长1M
log on(
    name=EDUC_log,
    filename='D:\db\EDUC.ldf ' ,
    size=2 ,
    maxsize=5 ,
    filegrowth=1
    )
```

运行成功后 D 盘 db 文件夹下生成的文件:



EDUC.ldf
SQL Server Database Transacti...
2.00 MB

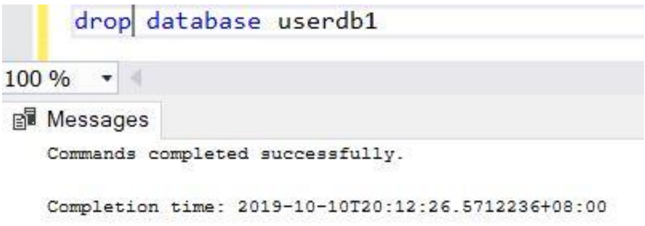
EDUC.mdf
SQL Server Database Primary ...
8.00 MB

3. 问题 3

使用

```
drop database userdb1
```

SQL 语句成功删除 1 中创建的数据库:



```
drop database userdb1
```

```
Commands completed successfully.

Completion time: 2019-10-10T20:12:26.5712236+08:00
```

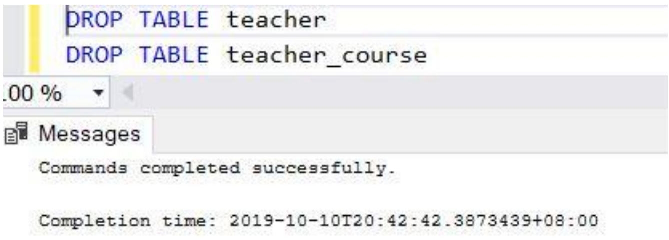
二、实验四

1. 问题 4

使用如下 SQL 命令:

```
DROP TABLE teacher
DROP TABLE teacher_course
```

结果如图:



```
DROP TABLE teacher
DROP TABLE teacher_course
```

```
Commands completed successfully.

Completion time: 2019-10-10T20:42:42.3873439+08:00
```

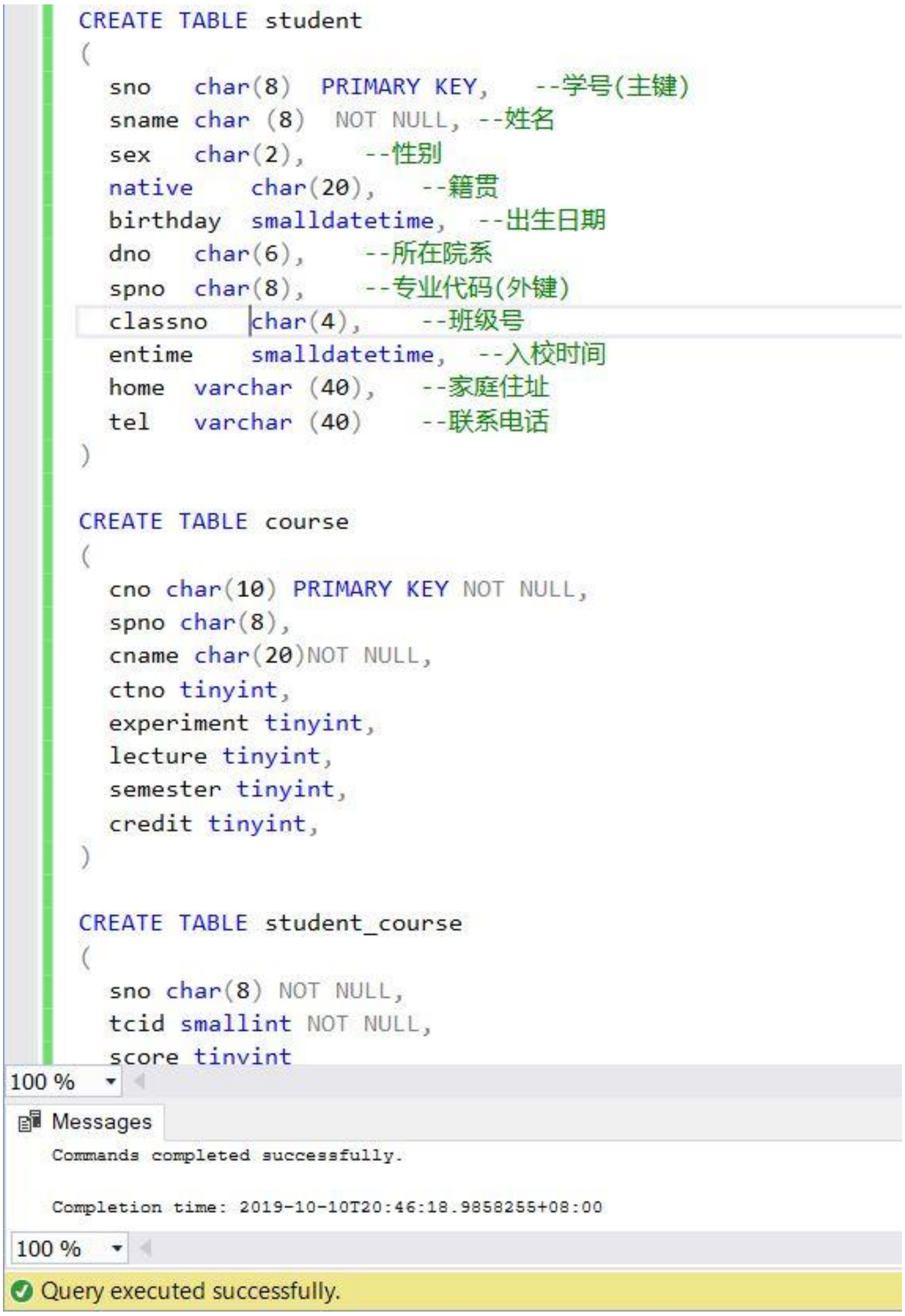
2. 问题 5

使用 SQL 语句创建各表命令如下:

```
Use EDUC
Go
```

```
CREATE TABLE student
(
  sno    char(8)  PRIMARY KEY,    --学号(主键)
  sname char (8)  NOT NULL, --姓名
  sex    char(2),--性别
  nativechar(20),    --籍贯
  birthday  smalldatetime,  --出生日期
  dno    char(6),--所在院系
  spno   char(8),--专业代码(外键)
  classno    char(4),--班级号
  entimesmalldatetime,  --入校时间
  home   varchar (40),   --家庭住址
  tel    varchar (40)--联系电话
)

CREATE TABLE course
(
  cno char(10) PRIMARY KEY NOT NULL,
  spno char(8),
  cname char(20)NOT NULL,
  ctno tinyint,
  experiment tinyint,
  lecture tinyint,
  semester tinyint,
  credit tinyint,
)

CREATE TABLE student_course
(
  sno char(8) NOT NULL,
  tcid smallint NOT NULL,
  score tinyint
  primary key(sno,tcid)
)

CREATE TABLE teacher
(
  tno char(8) PRIMARY KEY NOT NULL,
  tname char(8) NOT NULL,
  sex char(2),
  birthday date,
  dno char(6),
  pno tinyint,
  home varchar(40),
  zipcode char(6),
  tel varchar(40),
  email varchar(40),
)

CREATE TABLE teacher_course
(
  tcid smallint PRIMARY KEY NOT NULL,
  tno char(8),
  spno char(8),
  classno char(4),
  cno    char(10) NOT NULL,
  semester char(6),
  schoolyear char(10),
  classtime varchar(40),
  classroom varchar(40),
  weektime tinyint
)
```

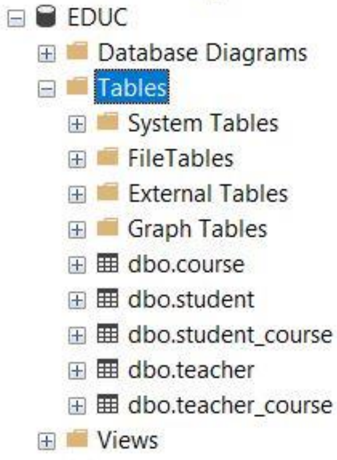
执行结果如下图:



```
CREATE TABLE student
(
  sno    char(8)  PRIMARY KEY,    --学号(主键)
  sname char (8)  NOT NULL, --姓名
  sex    char(2),       --性别
  native     char(20),    --籍贯
  birthday  smalldatetime,  --出生日期
  dno    char(6),       --所在院系
  spno   char(8),       --专业代码(外键)
  classno    char(4),     --班级号
  entime     smalldatetime,  --入校时间
  home   varchar (40),    --家庭住址
  tel    varchar (40)     --联系电话
)

CREATE TABLE course
(
  cno char(10) PRIMARY KEY NOT NULL,
  spno char(8),
  cname char(20)NOT NULL,
  ctno tinyint,
  experiment tinyint,
  lecture tinyint,
  semester tinyint,
  credit tinyint,
)

CREATE TABLE student_course
(
  sno char(8) NOT NULL,
  tcid smallint NOT NULL,
  score tinyint
```

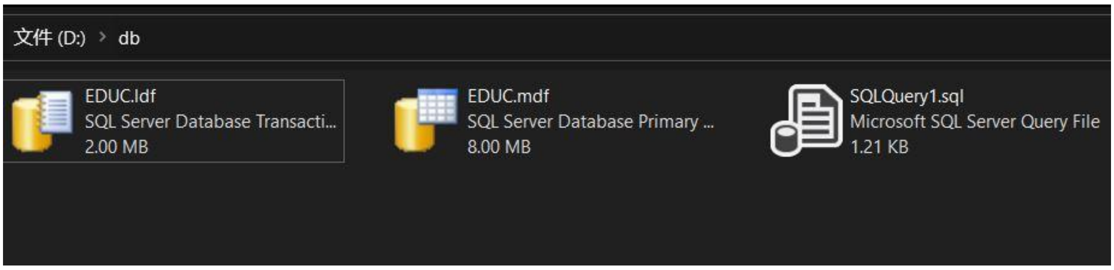
```
Commands completed successfully.

Completion time: 2019-10-10T20:46:18.9858255+08:00
```

Query executed successfully.



3. 问题 6

用快捷键 Ctrl+S，在弹出的窗口中点击保存，结果如图：



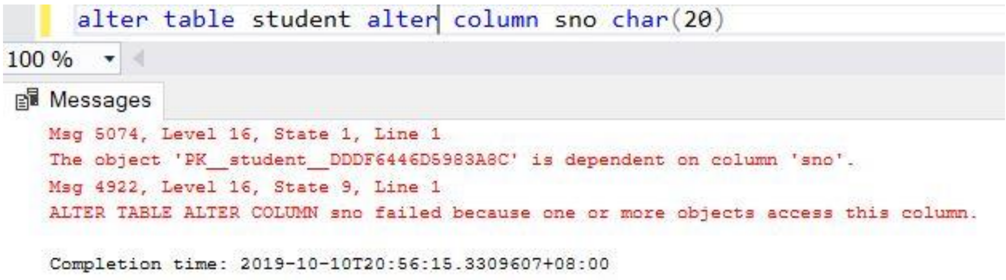
文件 (D:) › db

EDUC.ldf SQL Server Database Transacti... 2.00 MB	EDUC.mdf SQL Server Database Primary ... 8.00 MB	SQLQuery1.sql Microsoft SQL Server Query File 1.21 KB

三、实验五

1. 问题 1

如果直接用 sql 语句更改，会报如下错误：



```
alter table student alter column sno char(20)
```

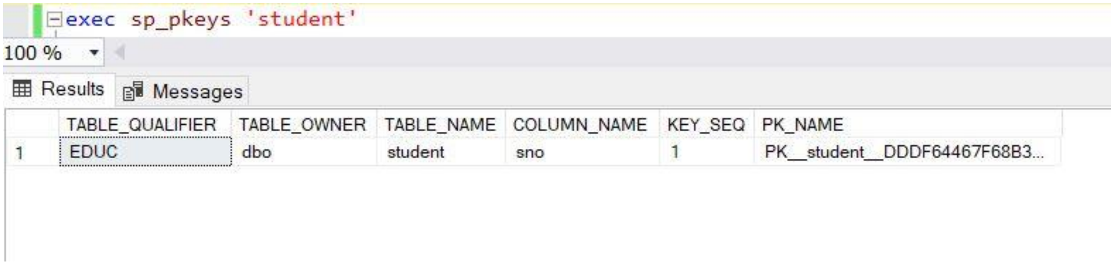
```
Msg 5074, Level 16, State 1, Line 1
The object 'PK__student__DDDF6446D5983A8C' is dependent on column 'sno'.
Msg 4922, Level 16, State 9, Line 1
ALTER TABLE ALTER COLUMN sno failed because one or more objects access this column.

Completion time: 2019-10-10T20:56:15.3309607+08:00
```

因而在更改前必须更改其主键属性，正确步骤如下：

首先通过执行

```
exec sp_pkeys 'student'
```

获得主键名：



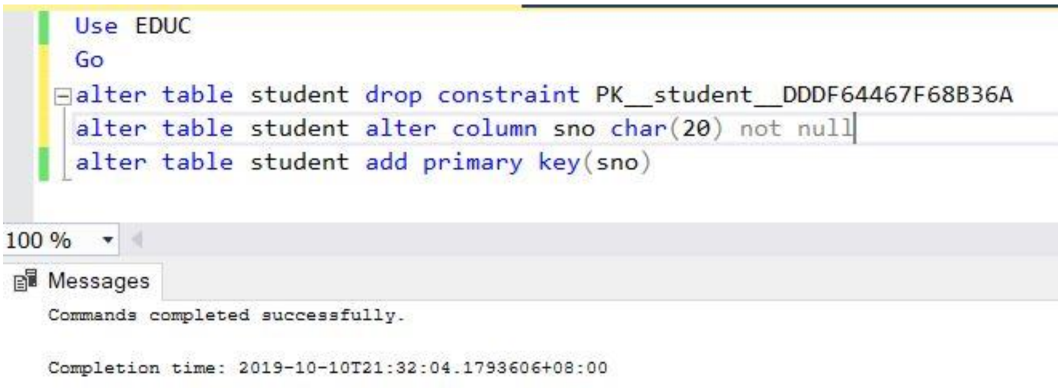
```
exec sp_pkeys 'student'
```

	TABLE_QUALIFIER	TABLE_OWNER	TABLE_NAME	COLUMN_NAME	KEY_SEQ	PK_NAME
1	EDUC	dbo	student	sno	1	PK__student__DDDF64467F68B3...

然后执行下列 sql 语句：

```
Use EDUC
Go
alter table student drop constraint PK__student__DDDF64467F68B36A
alter table student alter column sno char(20) not null
alter table student add primary key(sno)
```

运行成功：



```
Use EDUC
Go
alter table student drop constraint PK__student__DDDF64467F68B36A
alter table student alter column sno char(20) not null
alter table student add primary key(sno)
```

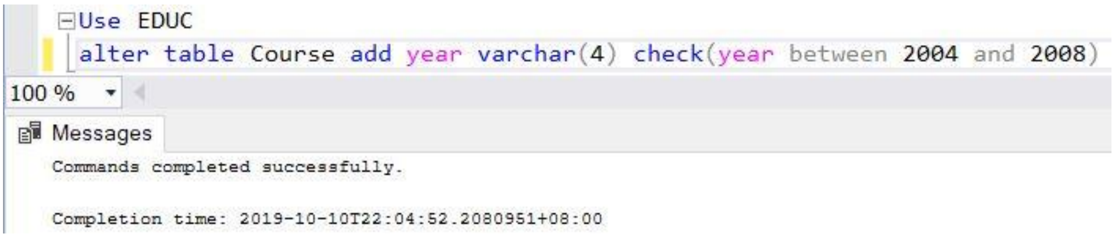
```
Commands completed successfully.

Completion time: 2019-10-10T21:32:04.1793606+08:00
```

2. 问题 2

使用如下 sql 语句：

```
Use EDUC
alter table Course add year varchar(4) check(year between 2004 and 2008)
```

执行结果：



```
Use EDUC
alter table Course add year varchar(4) check(year between 2004 and 2008)
```

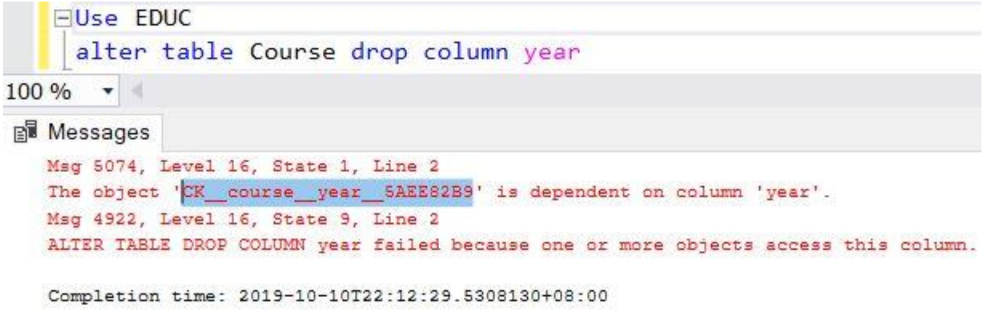
```
Commands completed successfully.

Completion time: 2019-10-10T22:04:52.2080951+08:00
```

3. 问题 3

首先通过执行

```
Use EDUC
alter table Course drop column year
```

报错，所以必须将该 key 删除：



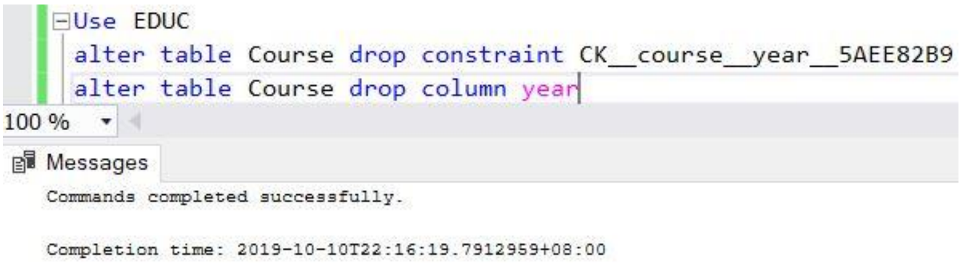
```
Use EDUC
alter table Course drop column year
```

```
Msg 5074, Level 16, State 1, Line 2
The object 'CK__course__year__5AEE82B9' is dependent on column 'year'.
Msg 4922, Level 16, State 9, Line 2
ALTER TABLE DROP COLUMN year failed because one or more objects access this column.

Completion time: 2019-10-10T22:12:29.5308130+08:00
```

通过错误信息获取 key 名，执行下列 sql 语句：

```
Use EDUC
alter table Course drop constraint CK__course__year__5AEE82B9
alter table Course drop column year
```

运行成功：



```
Use EDUC
alter table Course drop constraint CK__course__year__5AEE82B9
alter table Course drop column year
```

```
Commands completed successfully.

Completion time: 2019-10-10T22:16:19.7912959+08:00
```

四、实验六

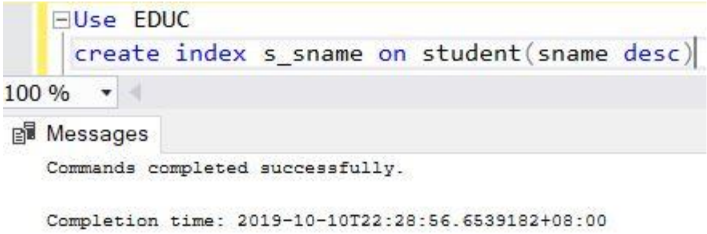
1. 问题 1

1) 在 student 表的 sname 列上建立普通降序索引

执行下列 sql 语句：

```
Use EDUC
create index s_sname on student(sname desc)
```

执行结果：***成功***



```
Use EDUC
create index s_sname on student(sname desc)
```

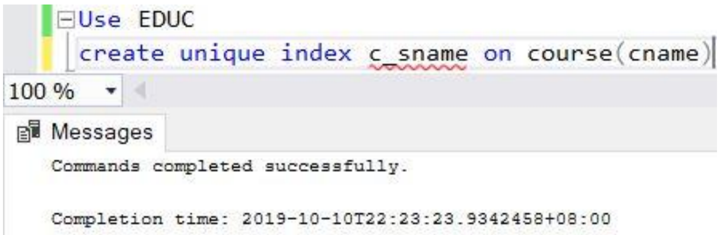
```
Commands completed successfully.

Completion time: 2019-10-10T22:28:56.6539182+08:00
```

2) 在 course 表的 cname 列上建立唯一索引

执行下列 sql 语句：

```
Use EDUC
create unique index c_sname on course(cname)
```

执行结果：***成功***



```
Use EDUC
create unique index c_sname on course(cname)
```

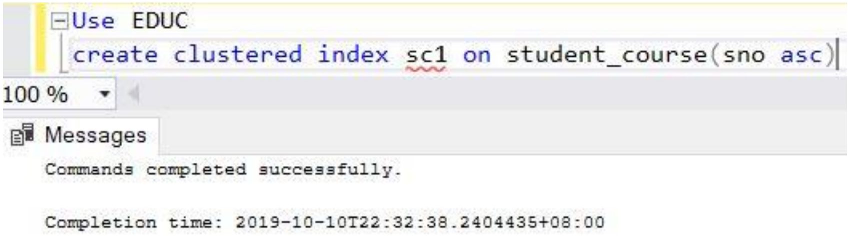
```
Commands completed successfully.

Completion time: 2019-10-10T22:23:23.9342458+08:00
```

3) 在 student_course 表的 sno 列上建立聚集索引

执行下列 sql 语句：

```
Use EDUC
create clustered index sc1 on student_course(sno asc)
```

执行结果：***成功***



```
Use EDUC
create clustered index sc1 on student_course(sno asc)
```

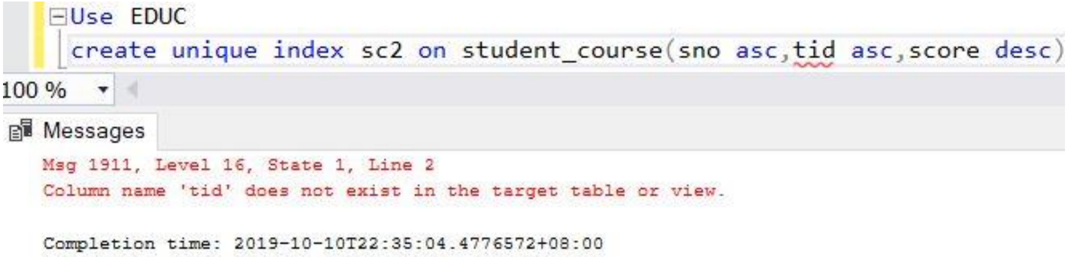
```
Commands completed successfully.

Completion time: 2019-10-10T22:32:38.2404435+08:00
```

4) 在 student_course 表的 sno(升序), tid(升序)和 score(降序)三列上建立一个普通索引

执行下列 sql 语句:

```
Use EDUC
create unique index sc2 on student_course(sno asc,tid asc,score desc)
```

执行结果: ***失败***



```
Use EDUC
create unique index sc2 on student_course(sno asc,tid asc,score desc)
```

```
Msg 1911, Level 16, State 1, Line 2
Column name 'tid' does not exist in the target table or view.

Completion time: 2019-10-10T22:35:04.4776572+08:00
```

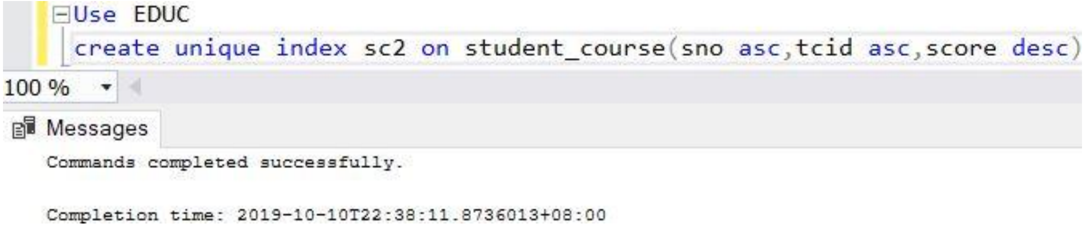
不存在 tid 列，应为 tcid。

改正:

执行下列 sql 语句:

```
Use EDUC
create unique index sc2 on student_course(sno asc,tcid asc,score desc)
```

执行结果: ***成功***



```
Use EDUC
create unique index sc2 on student_course(sno asc,tcid asc,score desc)
```

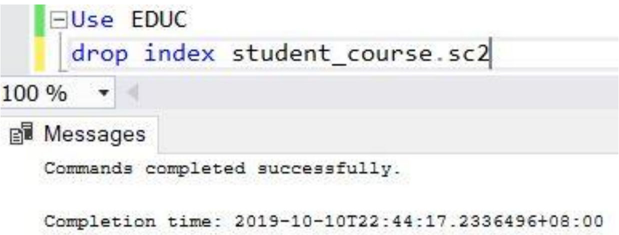
```
Commands completed successfully.

Completion time: 2019-10-10T22:38:11.8736013+08:00
```

2. 问题 2

执行下列 sql 语句:

```
Use EDUC
drop index student_course.sc2
```

执行结果: ***成功***



```
Use EDUC
drop index student_course.sc2
```

```
Commands completed successfully.

Completion time: 2019-10-10T22:44:17.2336496+08:00
```