**实 验 报 告 册**

2021-2022**学年 春**季 **学期**

专业年级 2020级计科06

学 号 2020204331

姓 名 王博文

课程名称 数据库原理

指导教师 侯国家

青岛大学计算机科学技术学院

**实验一：管理SQL Server 2008数据库**

**一、实验目的和要求**

1．了解数据库常用对象及组成；

2．熟悉SQL的基本概念和特点；

3．熟练掌握T-SQL 语句中的数据定义的使用方法；

4．熟悉在SQL Server 2008中创建及编辑数据库；

5．熟练掌握SQL的数据定义子句的使用方法。

**二、实验环境**

1．已安装SQL Server 2008企业版的计算机（60台）；

2．具有局域网网络环境，有固定ip地址。

**三、实验学时**

4学时

**四、实验内容及步骤**

1．要求在本地磁盘D创建一个学生-课程数据库（名称为student），只有一个数据文件和日志文件，文件名称分别为stu和stu\_log，物理名称为stu\_data.mdf 和stu\_log.ldf，初始大小都为5MB，增长方式分别为10%和1MB，数据文件最大为500MB，日志文件大小不受限制。

create database student

on

(name = student\_data,

filename = 'E:\数据库实验\Student\_data.mdf',

size = 5MB,

maxsize = 500MB,

filegrowth = 10%)

log on

(name = 'Student\_log',

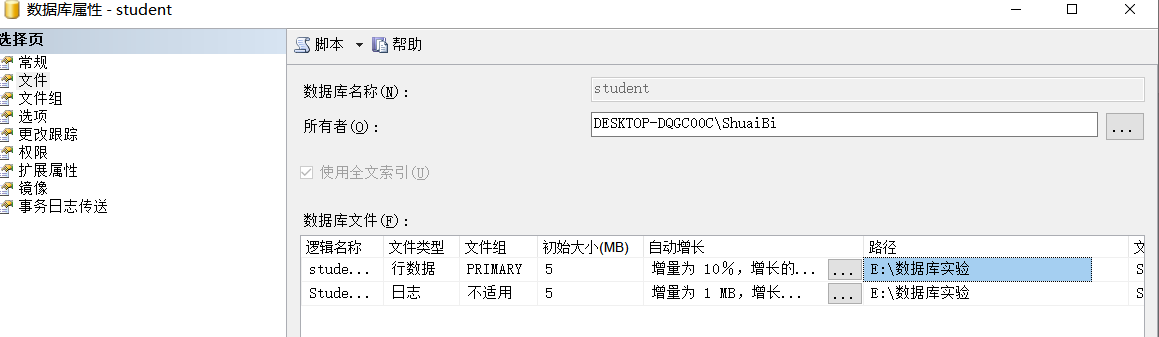
filename = 'E:\数据库实验\Student\_log.ldf',

size = 5MB,

maxsize = 500MB,

filegrowth = 1MB

)



2．创建一个Company数据库，该数据库的主数据文件逻辑名称为Company\_data，物理文件为Company.mdf，初始大小为10MB，最大尺寸为无限大，增长速度为10%；数据库的日志文件逻辑名称为Company\_log，物理文件名为Company.ldf，初始大小为1MB，最大尺寸为50MB，增长速度为1MB。

create database Company

on

(name = Company\_data,

filename = 'E:\数据库实验\Company\_data.mdf',

size = 10MB,

maxsize = UNLIMITED,

filegrowth = 10%)

log on

(name = 'Company\_log',

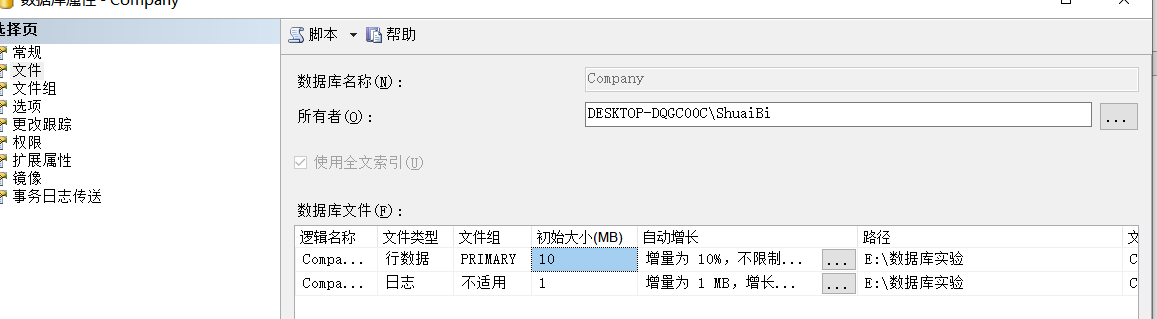
filename = 'E:\数据库实验\Company\_log.ldf',

size = 1MB,

maxsize = 50MB,

filegrowth = 1MB

)



3．创建数据库DB，具有2个数据文件，文件逻辑名分别为DB\_data1和DB\_data2，文件初始大小均为5MB，最大为100MB，按10%增长；只有一个日志文件，初始大小为7MB，按10%增长；所有文件都存储在D盘文件夹ceshi中；

create database DB

On primary

(name = DB\_data1,

filename = 'E:\数据库实验\ceshi\DB\_data1.mdf',

size = 5MB,

maxsize = 100MB,

filegrowth = 10%),

filegroup fg

(name = DB\_data2,

filename = 'E:\数据库实验\ceshi\DB\_data2.mdf',

size = 5MB,

maxsize = 100MB,

filegrowth = 10%)

log on

(name = 'DB\_log',

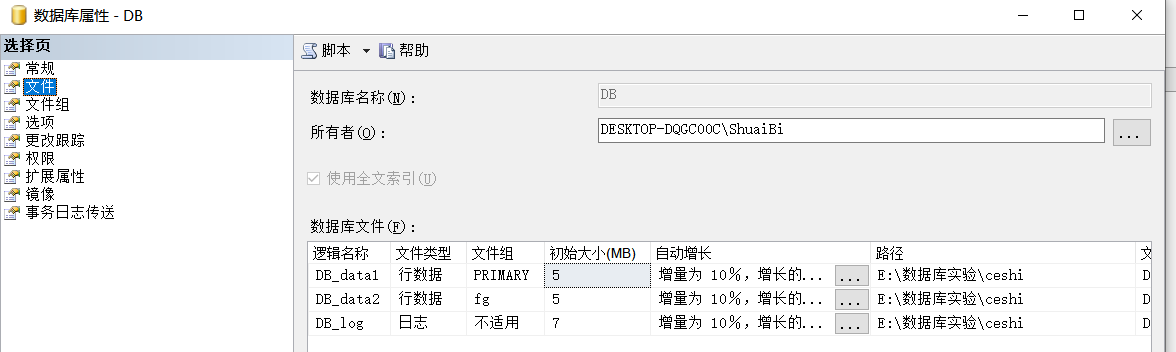
filename = 'E:\数据库实验\ceshi\DB\_log.ldf',

size = 7MB,

maxsize = UNLIMITED,

filegrowth = 10%

)



4.在数据库student中增加数据文件db2，初始大小为10MB，最大大小为50 MB，按10%增长；

alter database student

add file

(name = db2,

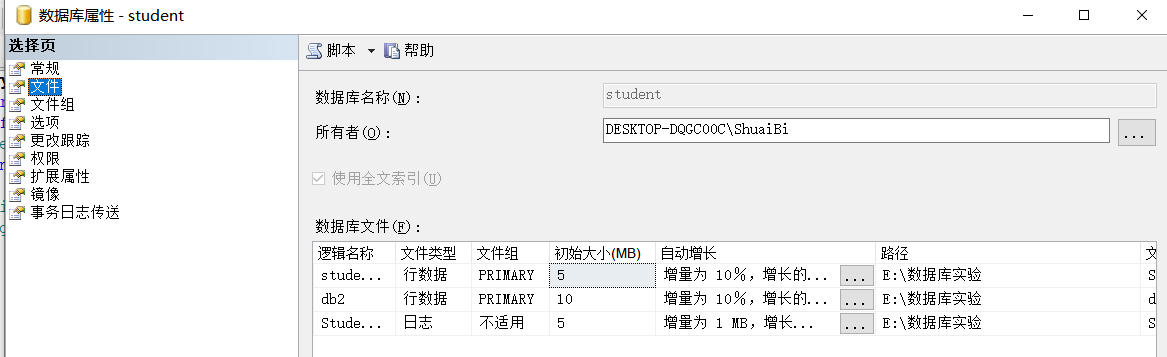
filename = 'E:\数据库实验\db2.ndf',

size = 10MB,

maxsize = 50MB,

filegrowth = 10%

)



1. 在数据库student中添加日志文件，保存在D盘中，初始大小为1MB，最大无限制，增长方式按照1MB增长

alter database student

add log file

(name = 'db2\_log',

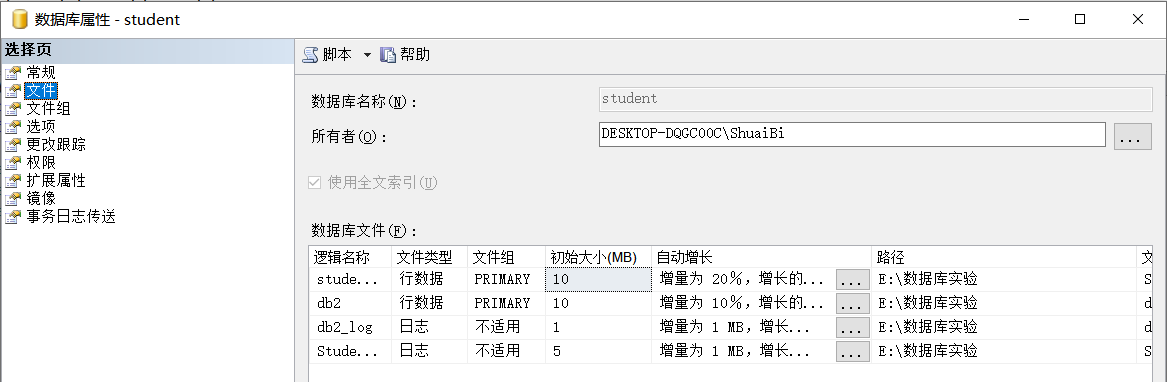
filename = 'E:\数据库实验\db2\_log.ldf',

size = 1MB,

maxsize = UNLIMITED,

filegrowth = 1MB

)



1. 修改数据库student主数据文件的大小，将主数据文件的初始大小修改为10Mb，增长方式为20%；

alter database student

modify file

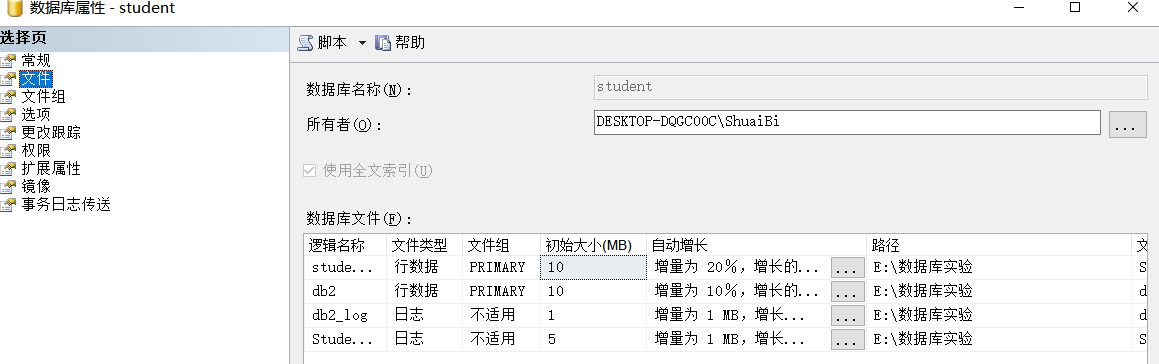
(

name = student\_data,

size = 10MB,

filegrowth = 20%

)



7.修改数据库student辅助数据文件初始大小为30MB，最大为100MB，按照10%增长，名称为db;

alter database student

modify file

(

name = db2,

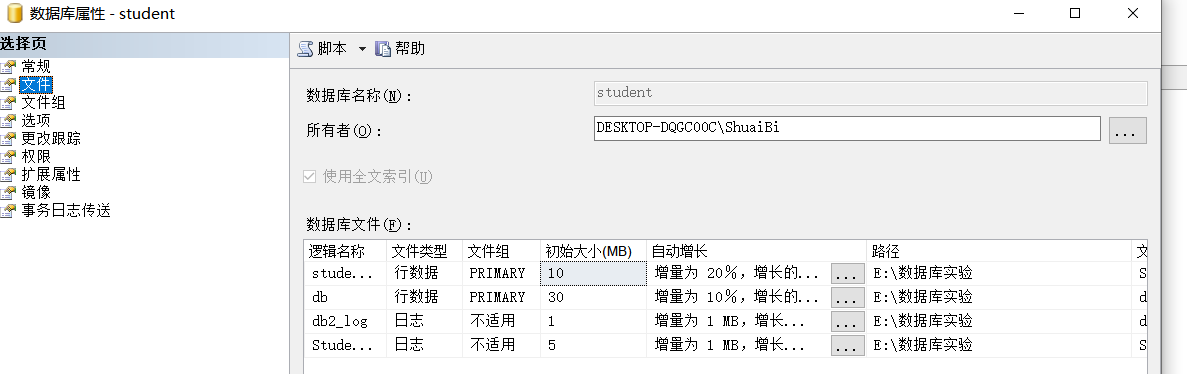
newname = db,

size = 30MB,

maxsize = 100MB,

filegrowth = 10%

)



8.删除数据库student辅助数据文件和第二个日志文件；

alter database student

remove file db

alter database student

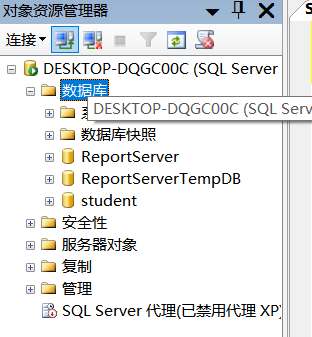
remove file db2\_log



9.删除数据库company和DB。

drop database DB

drop database Company



**五、出现的问题及解决办法**

1．主要的问题就是在建库中，没有理清好其中的关系，就像student是主数据文件，db是辅助数据文件。我把这几个文件和后面的company放在了一个文件夹里，虽说不会产生什么大错误（目前来说），但是还是不符合逻辑的存储格式

2．再就是文件名的紊乱。

**实验二：使用SQL Server管理数据表**

**一、实验目的和要求**

1．熟悉数据表的结构特点，；

2．了解SQL Server的基本数据类型；

3．熟悉在Management Stuio中创建、编辑及删除数据表；

4．熟悉使用T-SQL创建、编辑及删除数据表。

**二、实验环境**

1．已安装SQL Server 2008企业版的计算机（60台）；

2．具有局域网网络环境，有固定ip地址。

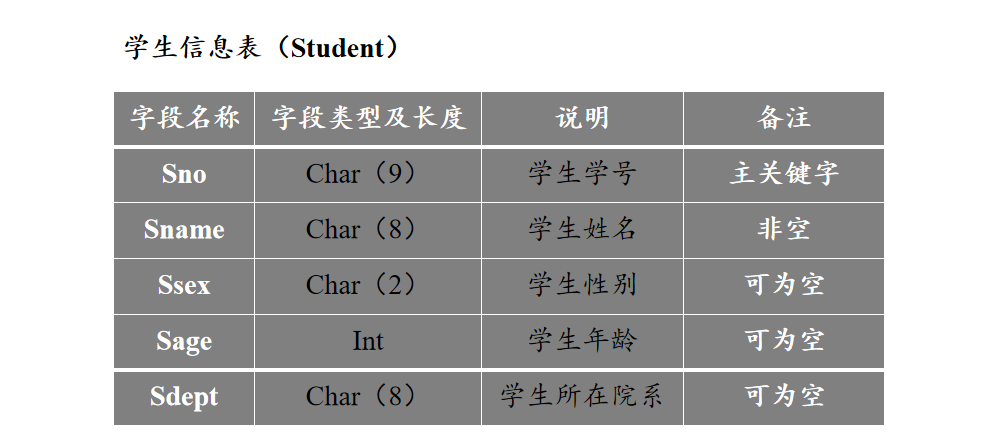
**三、实验学时**

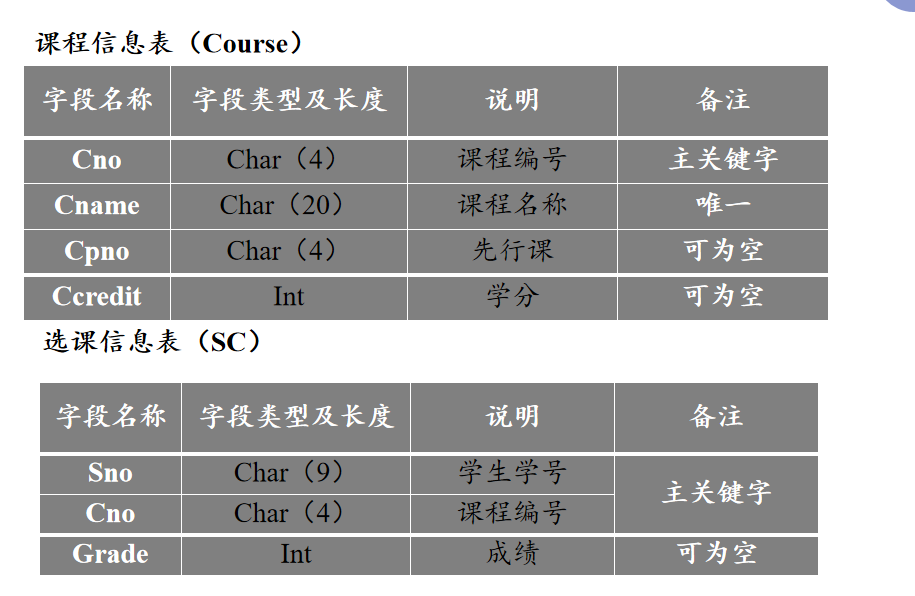
4学时

1. **实验内容及步骤**

请在指定数据库内完成以下内容：

现有学生-选课数据库SCD，数据库包含三张表，表结构如下所示





1. 依据数据表的结构创建相对应的数据表;

USE EX2

GO

CREATE TABLE Student

(

Sno char(9) PRIMARY KEY,

Sname char(8) NOT NULL,

Ssex char(2) ,

Sage int ,

Sdept char(8) ,

)

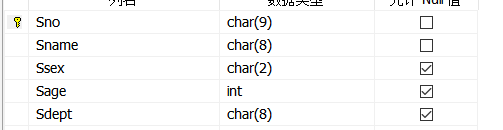
EXECUTE sp\_addextendedproperty N'MS\_Description', '学生学号', N'SCHEMA', N'dbo', N'table', N'Student', N'column', N'Sno'

EXECUTE sp\_addextendedproperty N'MS\_Description', '学生姓名', N'SCHEMA', N'dbo', N'table', N'Student', N'column', N'Sname'

EXECUTE sp\_addextendedproperty N'MS\_Description', '学生性别', N'SCHEMA', N'dbo', N'table', N'Student', N'column', N'Ssex'

EXECUTE sp\_addextendedproperty N'MS\_Description', '学生年龄', N'SCHEMA', N'dbo', N'table', N'Student', N'column', N'Sage'

EXECUTE sp\_addextendedproperty N'MS\_Description', '学生所在院系', N'SCHEMA', N'dbo', N'table', N'Student', N'column', N'Sdept'



USE EX2

GO

CREATE TABLE Course

(

Cno char(4) PRIMARY KEY,

Cname char(20) unique,

Cpno char(4) ,

Ccredit int,

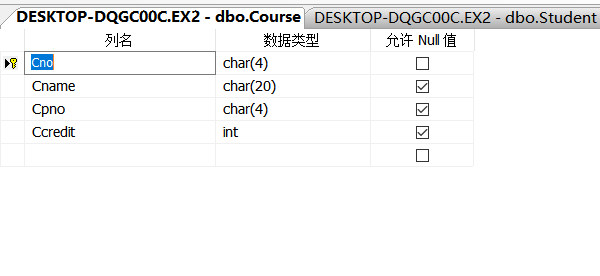
)

EXECUTE sp\_addextendedproperty N'MS\_Description', '课程编号', N'SCHEMA', N'dbo', N'table', N'Course', N'column', N'Cno'

EXECUTE sp\_addextendedproperty N'MS\_Description', '课程名称', N'SCHEMA', N'dbo', N'table', N'Course', N'column', N'Cname'

EXECUTE sp\_addextendedproperty N'MS\_Description', '先行课', N'SCHEMA', N'dbo', N'table', N'Course', N'column', N'Cpno'

EXECUTE sp\_addextendedproperty N'MS\_Description', '学分', N'SCHEMA', N'dbo', N'table', N'Course', N'column', N'Ccredit'



USE EX2

GO

CREATE TABLE SC

(

Sno char(9) ,

Cno char(4) ,

Grade int ,

primary key(Sno,Cno),

foreign key(Sno) references Student(Sno),

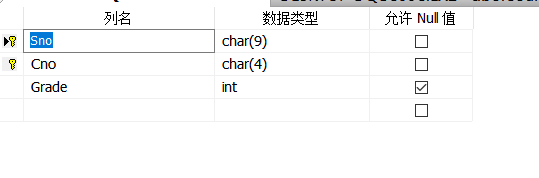
foreign key(Cno) references Course(Cno),

)

EXECUTE sp\_addextendedproperty N'MS\_Description', '学生学号', N'SCHEMA', N'dbo', N'table', N'SC', N'column', N'Sno'

EXECUTE sp\_addextendedproperty N'MS\_Description', '课程编号', N'SCHEMA', N'dbo', N'table', N'SC', N'column', N'Cno'

EXECUTE sp\_addextendedproperty N'MS\_Description', '成绩', N'SCHEMA', N'dbo', N'table', N'SC', N'column', N'Grade'



1. 在表Student中增加新字段 “班级名称（Sclass）、专业（Smajor）”；

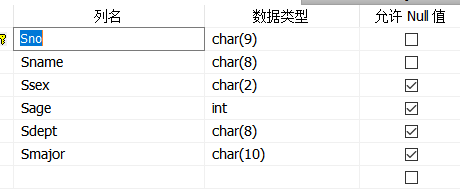
alter table Student add Sclass char(10);

alter table Student add Smajor char(10);



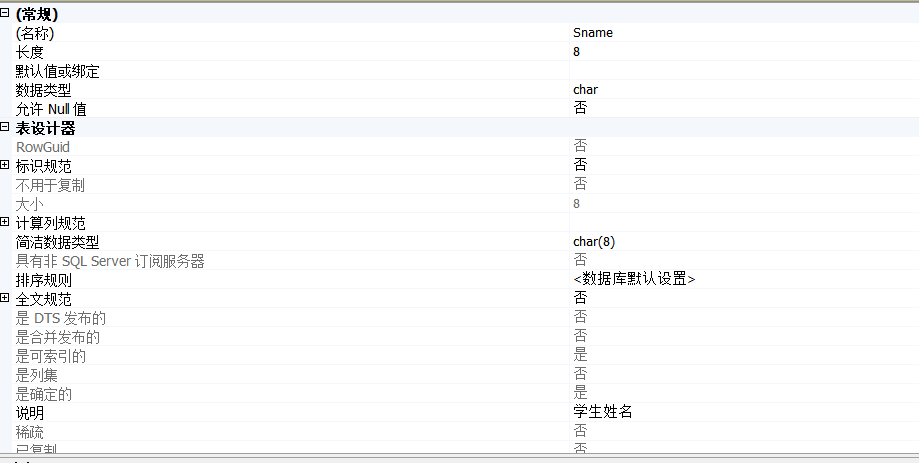
1. 在表Student中删除字段“班级名称（Sclass）”；

alter table Student drop Column Sclass;



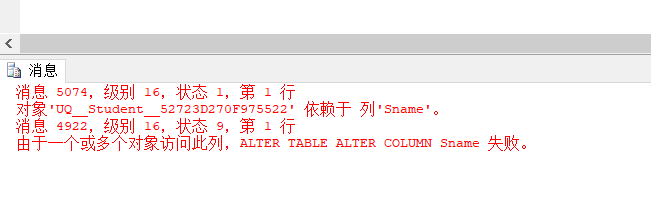
1. 增加姓名必须唯一的约束；

alter table Student ADD UNIQUE(Sname);



1. 修改表Student中字段名为“Sname”的字段长度由原来的8改为10；

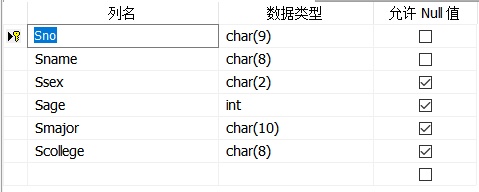
alter table Student alter column Sname char(10);



1. 修改表Student中字段“Sdept”名称为“Scollege”；

ALTER TABLE Student drop column Sdept

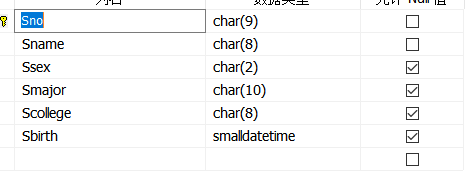
ALTER TABLE Student add Scollege char(8)



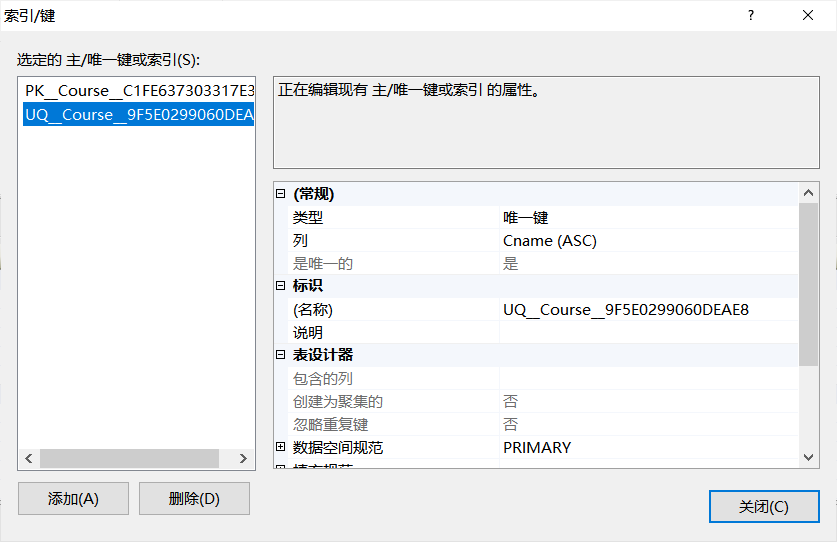
1. 修改表Student中Sage字段名称为Sbirth，类型为smalldatetime；

ALTER TABLE Student drop column Sage

ALTER TABLE Student add Sbirth smalldatetime

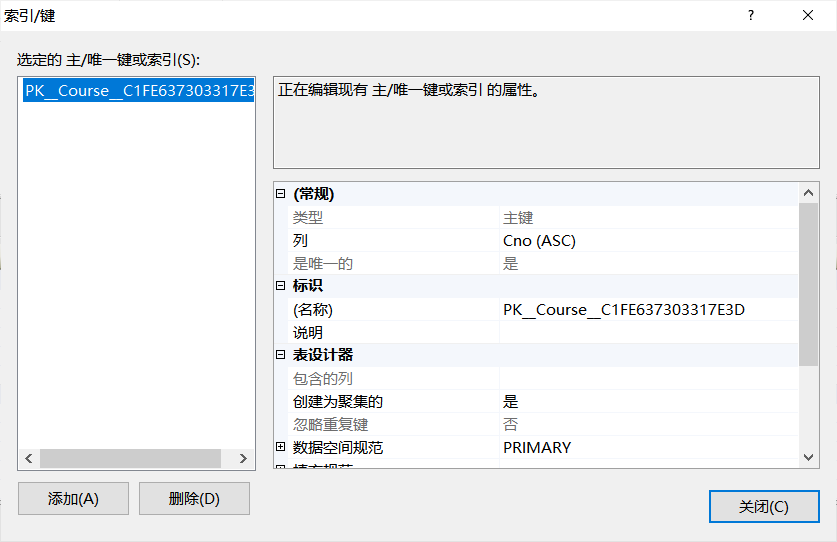


1. 删除课程名唯一的约束；



ALTER TABLE Course

DROP CONSTRAINT UQ\_\_Course\_\_9F5E0299060DEAE8

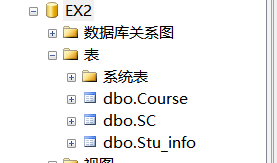


1. 修改表Student名称为Stu\_info；

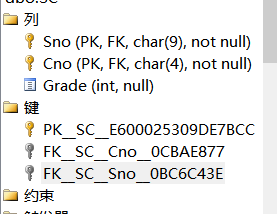
use EX2

GO

exec sp\_rename 'Student','Stu\_info'

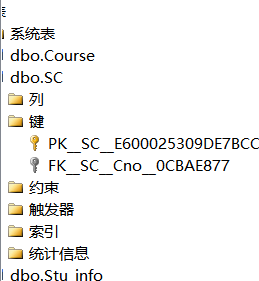


1. 删除基本表Stu\_info。

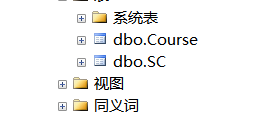


ALTER TABLE SC

DROP CONSTRAINT FK\_\_SC\_\_Sno\_\_0BC6C43E



DROP TABLE Stu\_info



**五、出现的问题及解决办法**

1．要对表的总体进行改变的时候，应该是在由约束条件之前，一旦此列拥有了约束条件，那么就无法对其进行修改，从（5）和（10）中就可以看出来，操作会繁琐很多。

2．说明好像没学过，所以就从网上学习了一下，感觉就是一些重复的代码

3．学会了对表的主码、外码、约束的查询，分别为PK 、 FK 、 UQ ……

**实验三：****管理SQL Server表数据**

**一、实验目的和要求**

1．熟悉数据表结构及使用特点；

2．熟悉使用Management Stuio界面方式管理数据表数据；

3．熟悉使用T-SQL语句管理数据表数据。

**二、实验环境**

1．已安装SQL Server 2008企业版的计算机（60台）；

2．具有局域网网络环境，有固定ip地址。

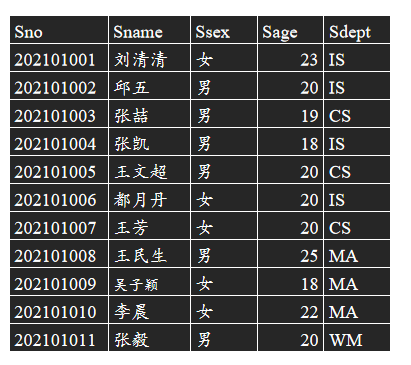
**三、实验学时**

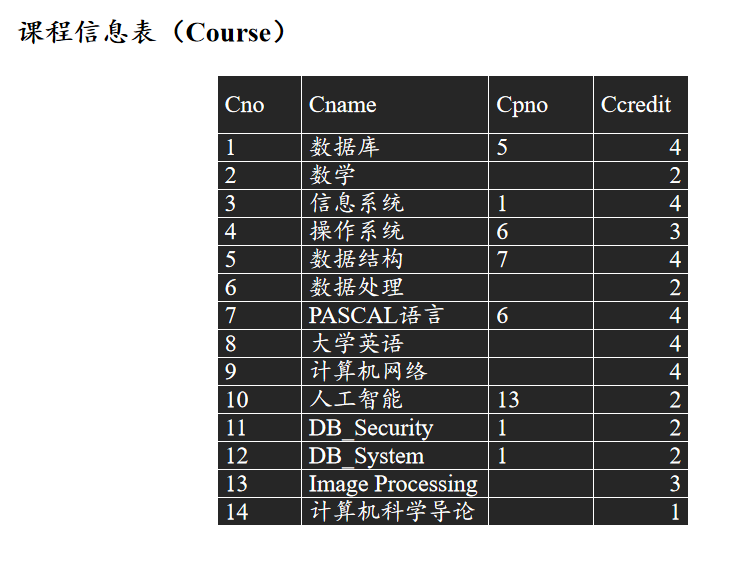
4学时

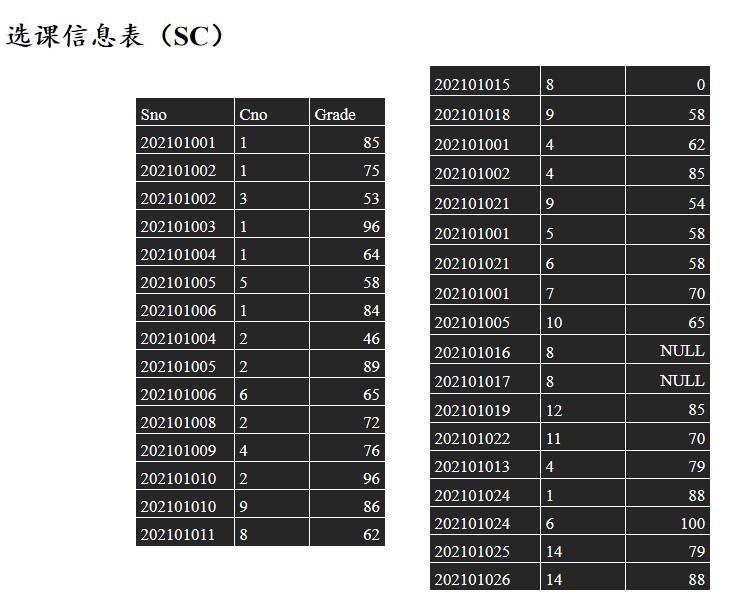
**四、实验内容及步骤**

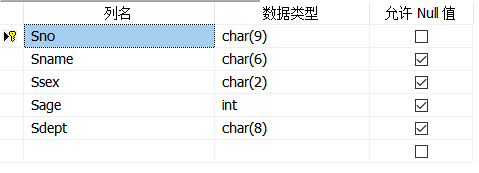
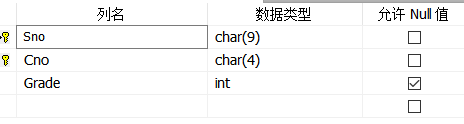
1. 向各个数据表中插入如下记录：

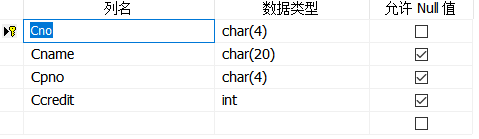
学生信息表（Student）







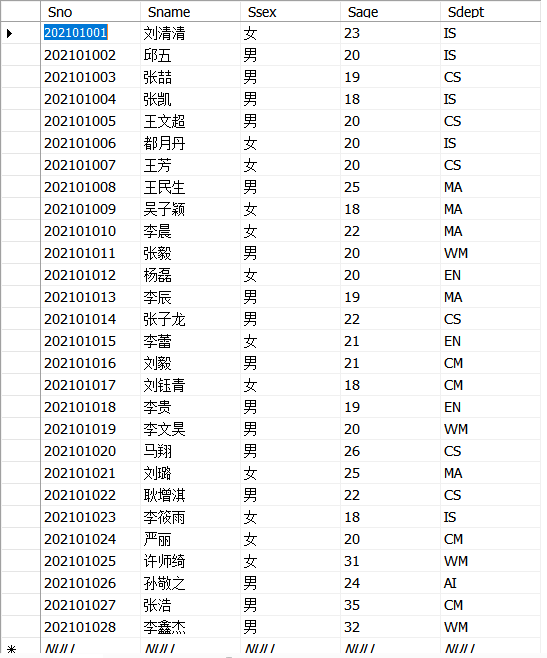




USE EX3

GO

INSERT INTO Student VALUES

 ('202101001','刘清清','女',23,'IS'),

('202101002','邱五','男',20,'IS'),

('202101003','张喆','男',19,'CS'),

('202101004','张凯','男',18,'IS'),

('202101005','王文超','男',20,'CS'),

('202101006','都月丹','女',20,'IS'),

('202101007','王芳','女',20,'CS'),

('202101008','王民生','男',25,'MA'),

('202101009','吴子颖','女',18,'MA'),

('202101010','李晨','女',22,'MA'),

('202101011','张毅','男',20,'WM'),

('202101012','杨磊','女',20,'EN'),

('202101013','李辰','男',19,'MA'),

('202101014','张子龙','男',22,'CS'),

('202101015','李蕾','女',21,'EN'),

('202101016','刘毅','男',21,'CM'),

('202101017','刘钰青','女',18,'CM'),

('202101018','李贵','男',19,'EN'),

('202101019','李文昊','男',20,'WM'),

('202101020','马翔','男',26,'CS'),

('202101021','刘璐','女',25,'MA'),

('202101022','耿增淇','男',22,'CS'),

('202101023','李筱雨','女',18,'IS'),

('202101024','严丽','女',20,'CM'),

('202101025','许师绮','女',31,'WM'),

('202101026','孙敬之','男',24,'AI'),

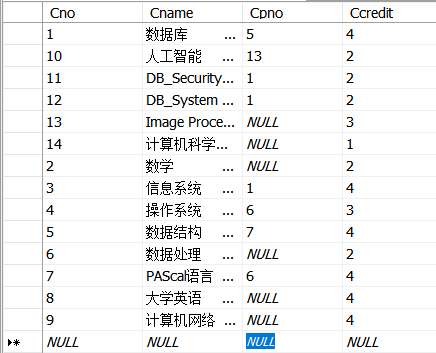
('202101027','张浩','男',35,'CM'),

('202101028','李鑫杰','男',32,'WM')

GO

use EX3

go

insert into Course values

('1','数据库','5',4),

('2','数学',null,2),

('3','信息系统','1',4),

('4','操作系统','6',3),

('5','数据结构','7',4),

('6','数据处理',null,2),

('7','PAScal语言','6',4),

('8','大学英语',null,4),

('9','计算机网络',null,4),

('10','人工智能','13',2),

('11','DB\_Security','1',2),

('12','DB\_System','1',2),

('13','Image Processing',null,3),

('14','计算机科学导论',null,1);

go

USE EX3

GO

INSERT INTO SC VALUES

('202101001','1',85),

('202101002','1',75),

('202101002','3',53),

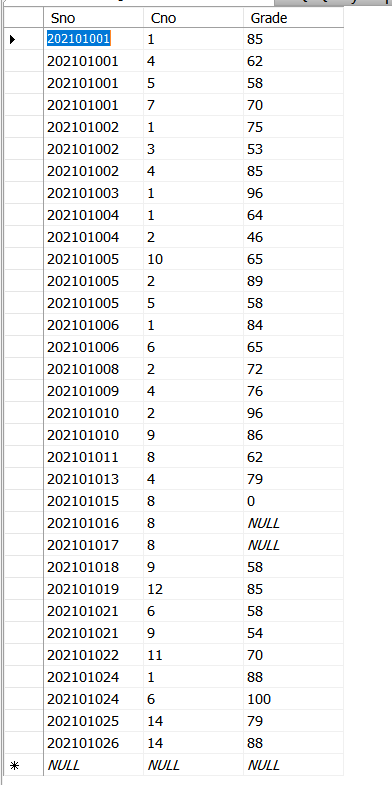
('202101003','1',96),

('202101004','1',64),

('202101005','5',58),

('202101006','1',84),

('202101004','2',46),

 ('202101005','2',89),

('202101006','6',65),

('202101008','2',72),

('202101009','4',76),

('202101010','2',96),

('202101010','9',86),

('202101011','8',62),

('202101015','8',0),

('202101018','9',58),

('202101001','4',62),

('202101002','4',85),

('202101021','9',54),

('202101001','5',58),

('202101021','6',58),

('202101001','7',70),

('202101005','10',65),

('202101016','8',null),

('202101017','8',null),

('202101019','12',85),

('202101022','11',70),

('202101013','4',79),

('202101024','1',88),

('202101024','6',100),

('202101025','14',79),

('202101026','14',88)

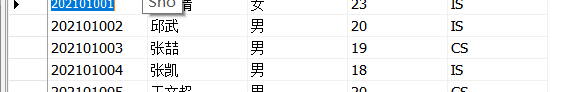
Go

1. 修改IS系姓名为“邱五”的学生姓名为“邱武”；

UPDATE Student

SET Sname = '邱武'

WHERE Sname = '邱五' and Sdept = 'IS'



1. 修改课程“操作系统”的学分为4学分；

UPDATE Course

SET Ccredit = 4

WHERE Cname = '操作系统'

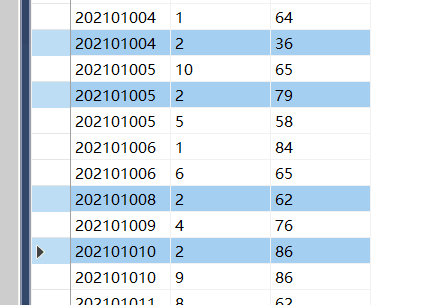


1. 将选修课程“2”的同学成绩减10分；

UPDATE SC

SET Grade = Grade-10

WHERE Cno = '2'



1. 将选修课程“数据库”和“计算机网络”的同学成绩加5分；

UPDATE SC

SET Grade = Grade+5

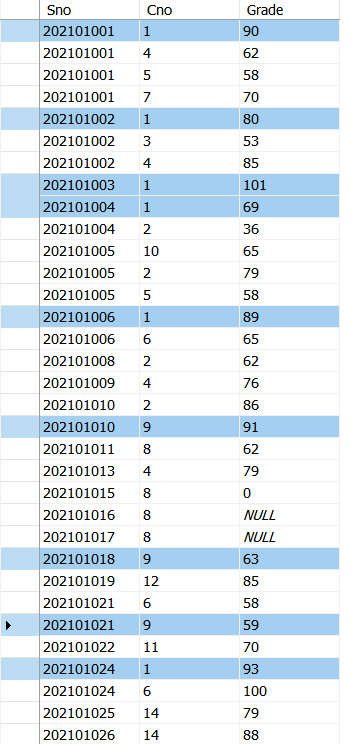
WHERE Cno IN

(SELECT Cno

FROM Course

WHERE Cname='数据库' OR Cname='计算机网络')

GO



1. 将学号为“202101012”的学生信息重新设置为“申恩娟、女、20、MA”；

use EX3

go

UPDATE Student

SET Sname = '申恩娟',

Ssex = '女',

Sage = 20,

Sdept = 'MA'

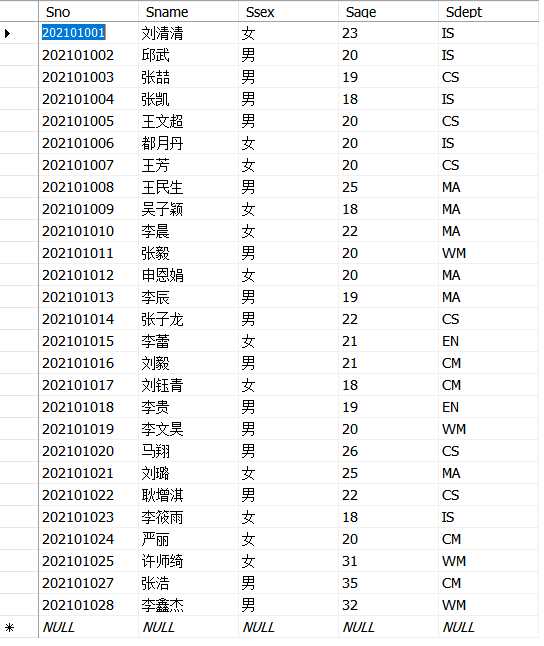
WHERE Sno = '202101012'

GO



1. 删除数据表Student中AI系的学生记录；

USE EX3

GO

DELETE

FROM SC

WHERE Sno in(

select Sno

from Student

where Sdept='AI'

)

GO

DELETE

from Student

where Sdept='AI'

GO

1. 删除数据表Student中WM系年龄大于30的男同学的记录；

USE EX3

GO

DELETE

FROM SC

WHERE Sno in(

select Sno

from Student

where Sdept='WM' AND Sage>30 AND Sage='男'

)

GO

DELETE

from Student

where Sdept='WM' AND Sage>30 AND Sage='男'

GO

1. 删除数据表Course中学分低于2学分的课程信息；

USE EX3

GO

DELETE

FROM SC

WHERE Cno in(

select Cno

from Course

where Ccredit<2

)

GO

DELETE

from Course

where Ccredit<2

GO

1. 删除成绩为NULL的所有女生的选课记录；

USE EX3

GO

DELETE

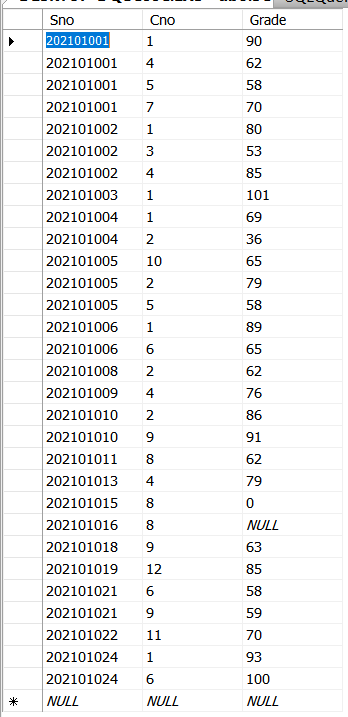
FROM SC

WHERE Sno in(

select Sno

from Student

where Ssex='女'

)AND Grade is null

GO

1. 删除选修“计算机科学导论”所有男生的选课信息。

USE EX3

GO

DELETE

FROM SC

WHERE Cno IN (SELECT Cno

FROM Course

WHERE Cname ='计算机科学导论')

AND

Sno IN( SELECT Sno

FROM Student

WHERE Ssex = '男')

GO

**五、出现的问题及解决办法**

1．主要的问题就是在建表的时候加了外码，使我以后的删除增加了一些难度，但是我跟同学对比了删除后的数据，都是一样的，于是就用着带外码的做，也想提升一下能力

2．建表的时间太长了。

**实验四：数据库单表查询**

**一、实验目的和要求**

1．掌握SELECT语句的基本语法和查询条件表示方法；

2．掌握查询条件表达式和使用方法；

3．掌握GROUP BY 子句的作用和使用方法；

4．掌握HAVING子句的作用和使用方法；

5．掌握ORDER BY子句的作用和使用方法。

**二、实验环境**

1．已安装SQL Server 2012企业版的计算机（60台）；

2．具有局域网网络环境，有固定ip地址。

**三、实验学时**

4学时

**四、实验内容及步骤**

以实验三数据库中数据为基础，请使用T-SQL 语句实现以下操作：

1. 查找CS系所有女生的信息；

USE EX3

GO

SELECT \*

FROM Student

WHERE Sdept='CS' AND Ssex='女'

Go



1. 列出所有不姓王的学生的学号和姓名；

USE EX3

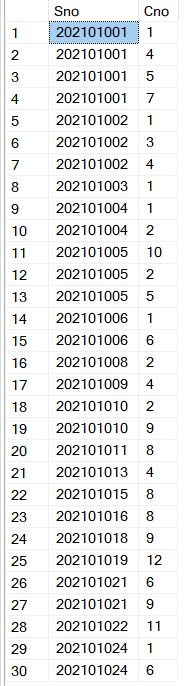
GO

SELECT Sno,Sname

FROM Student

WHERE Sname not like '王%'

GO

1. 查找选修了课程的学生的学号和课程号；

USE EX3

GO

SELECT Sno,Cno

FROM SC

GO

1. 列出姓“刘”且全名为2个汉字的学生信息；

USE EX3

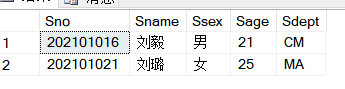
GO

SELECT \*

FROM Student

WHERE Sname like '刘\_'

GO



1. 所有学生的学号、姓名及出生年份；

USE EX3

GO

SELECT Sno,Sname,2022-Sage '出生年份'

FROM Student

GO

1. 查询选修了课程的学生人数；

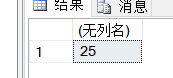
USE EX3

GO

SELECT COUNT(distinct Sno)

FROM Student

GO



1. 查询所有成绩为空的学生的学号和课程号；

USE EX3

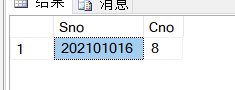
GO

SELECT Sno,Cno

FROM SC

WHERE Grade is null

GO



1. 列出选修了“8” 号课程的学生选课信息，按成绩的降序排列；

USE EX3

GO

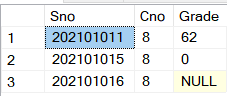
SELECT \*

FROM SC

WHERE Cno='8'

ORDER BY Grade DESC

GO



1. 查询选修“1”号课程中最高分和最低分的学生的成绩；

USE EX3

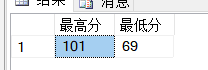
GO

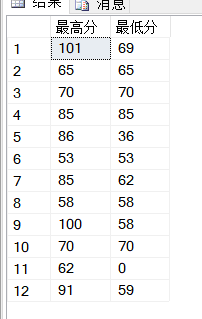
SELECT MAX(Grade)'最高分',MIN(Grade)'最低分'

FROM SC

WHERE Cno='1'

GO



1. 查询每一门课的课程号及其最高分、最低分和平均成绩。

USE EX3

GO

SELECT MAX(Grade)'最高分',MIN(Grade)'最低分'

FROM SC

Group by Cno

GO

1. 列出同时选修“1”号课程和“2”号课程的学生的学号；

USE EX3

GO

SELECT Sno

FROM SC

WHERE Sno in(

SELECT Sno

FROM SC

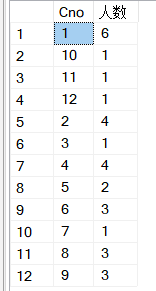
WHERE Cno='1'

)and Cno='2'

GO



1. 求各个课程号及相应的选课人数；

USE EX3

GO

SELECT Cno,COUNT(Sno)'人数'

FROM SC

GROUP BY Cno

GO

1. 查询平均成绩大于等于80分的学生学号和平均成绩；

USE EX3

GO

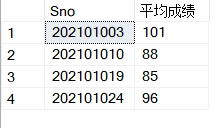
SELECT Sno,AVG(Grade)'平均成绩'

FROM SC

GROUP BY Sno

HAVING AVG(Grade)>=80

GO



1. 按照出生年份降序显示所有学生的学号、姓名及出生年份，在结果集中列标题分别指定为“学号，姓名，出生年份”；

USE EX3

GO

SELECT Sno'学号',Sname'姓名',2022-Sage '出生年份'

FROM Student

Order by 2022-Sage DESc

GO

1. 显示选修两门课以上课程的学生的学号和课程数；

USE EX3

GO

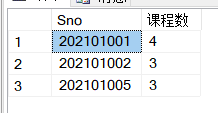
SELECT Sno,COUNT(Cno)'课程数'

FROM SC

group by Sno

HAVIng COUNT(Cno)>2

GO



1. 显示学号第九位是1、3、5、7或者9的学生的信息；

USE EX3

GO

SELECT \*

FROM Student

WHERE Sno like '%1' or Sno like '%3'or Sno like '%5'or Sno like '%7'or Sno like '%9'

GO



1. 查询与“刘清清”在同一个系的学生的信息；

USE EX3

GO

SELECT \*

FROM Student

WHERE Sdept in(

SELECT Sdept

FROM Student

WHERE Sname='刘清清'

)

GO



1. 显示平均成绩大于“202101006” 学生平均成绩的所有学生的学号和平均成绩；

USE EX3

GO

SELECT Sno,AVG(Grade)'平均成绩'

FROM SC

GROUP BY Sno

HAVING AVG(Grade)>(

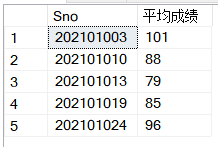
SELECT AVG(Grade)

FROM SC

WHERE Sno='202101006'

)

GO



1. 列出有二门以上课程成绩（含两门）在75分以上的学生的学号；

USE EX3

GO

SELECT Sno,AVG(Grade) '平均成绩'

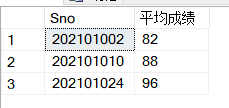
FROM SC

WHERE Grade>75

GROUP BY sno

HAVING COUNT (Cno)>=2

GO

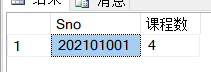


1. 查询选修课程数量最多的学生的学号和课程数；

USE EX3

GO

SELECT Sno,COUNT(Cno)'课程数'

FROM SC

GROUP BY Sno

HAVING COUNT(Cno)>=all

(SELECT COUNT(Cno)

FROM SC

GROUP BY Sno)

GO

21. 无人选修的课程的课程号。

USE EX3

GO

SELECT Cno

FROM Course

WHERE Cno not in(

SELECT Cno

FROM SC

)

GO



**五、出现的问题及解决办法**

1．本次实验几乎没有出现什么问题，唯一的几个难点就是后面几个题的代码的编写。

**实验五：数据库综合查询及完整性约束**

**一、实验目的和要求**

1. 掌握SELECT语句的基本语法和查询条件表示方法；

2. 掌握查询条件种类和表示方法；

3. 掌握连接查询的表示及使用；

4. 掌握嵌套查询的表示及使用；

5. 了解集合查询的表示及使用；

6. 掌握视图的创建及使用。

**二、实验环境**

1. 已安装SQL Server 2012企业版的计算机（60台）；

2．具有局域网网络环境，有固定ip地址。

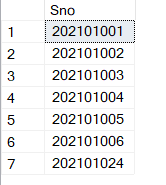
**三、实验学时**

4学时

**四、实验内容及步骤**

以数据库原理实验二数据库中数据为基础，请使用T-SQL 语句实现以下操作：

1. 查询选修了‘数据库’或者‘数据结构’的学生学号；

use EX3

go

select distinct Sno

from SC,Course

where SC.Cno=Course.Cno and

Cname in('数据库','数据结构')

1. 查询张喆同学所有选修的课程的名称；

use EX3

go

select Cname

from SC,Course,Student

where SC.Cno=Course.Cno and SC.Sno=Student.Sno and

Sname='张喆'

1. 查询以‘DB\_’开头，且倒数第4个字符为‘r’的课程的详细情况；

use EX3

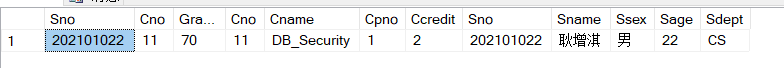
go

select \*

from SC,Course,Student

where SC.Cno=Course.Cno and SC.Sno=Student.Sno and

Cname like 'DB\\_%r\_\_\_' escape'\';



1. 查询缺少成绩的所有学生的学号和姓名；

use EX3

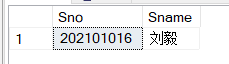
go

select Student.Sno,Sname

from SC,Student

where SC.Sno=Student.Sno and

Grade is null



1. 查询与‘都月丹’年龄不同的所有学生的信息；

use EX3

go

select \*

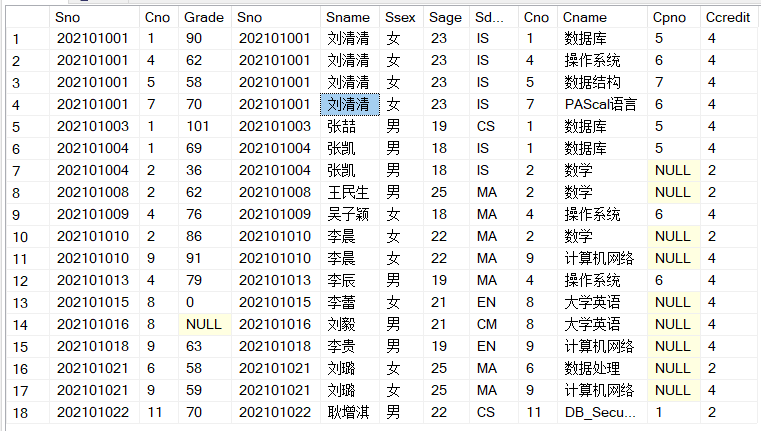
from SC,Student,Course

where SC.Sno=Student.Sno and SC.Cno=Course.Cno

and Sage not in(select Sage

from Student

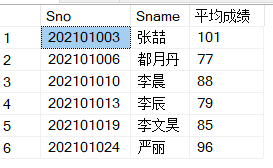
where Sname='都月丹')



1. 查询所选课程的平均成绩大于‘吴子颖’的平均成绩的学生学号、姓名及平均成绩；

use EX3

go

select student.Sno,Sname,AVG(Grade)平均成绩

from SC,Student

where SC.Sno=Student.Sno

group by Student.Sno,Sname

having AVG(Grade)>(select AVG(Grade)

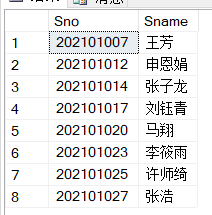
from SC,Student

where SC.Sno=Student.Sno

and Sname='吴子颖')

1. 查询没有选修课程的学生的学号和姓名；

use EX3

go

select student.Sno,Sname

from Student

where Student.Sno not in(

select Student.Sno

from SC,Student

where SC.Sno=Student.Sno

group by Student.Sno,Sname

having count(Cno)>0

)

1. 按照“学号，姓名，已修学分”的顺序列出学生学分的获得情况，其中已修学分为考试已经及格的课程学分之和；

use EX3

go

select student.Sno,Sname,SUM(Ccredit)已修学分

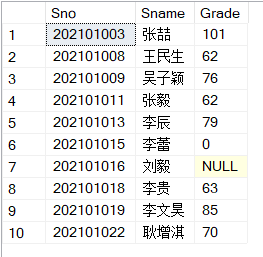
from Student,SC,Course

where SC.Sno=Student.Sno and SC.Cno=Course.Cno

and Grade>=60

group by student.Sno,Sname



1. 列出只选修一门课程的学生的学号、姓名及成绩；

use EX3

go

select student.Sno,Sname,Grade

from Student,SC

where Student.Sno=SC.Sno and Student.Sno in (select Sno

from SC

group by Sno

having count(SC.Cno)=1)

1. 至少选修‘数据库’或‘数据结构’课程的学生的基本信息；

use EX3

go

select Student.Sno,Sname,Ssex,Sage,Sdept

from SC,Student,Course

where SC.Sno=Student.Sno and SC.Cno=Course.Cno and

Cname in('数据库','数据结构')



1. 查询既选修了‘数据库’又选修了‘数据结构’课程的学生的基本信息；

use EX3

go

select Student.Sno,Sname,Ssex,Sage,Sdept

from SC,Student,Course

where SC.Sno=Student.Sno and SC.Cno=Course.Cno

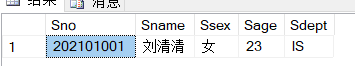
and Student.Sno in (select Student.Sno

from SC,Student,Course

where SC.Sno=Student.Sno and SC.Cno=Course.Cno

and Cname='数据库'

)and Cname='数据结构'



1. 查询只有2名学生选修的课程的课程号、课程名；

use EX3

go

select SC.Cno,Cname

from SC,Course

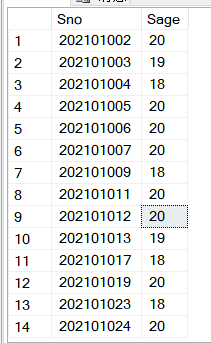
where SC.Cno=Course.Cno

group by SC.Cno,Cname

having count(Sno)=2



1. 列出其他院系中比EN系某一学生年龄小的学生的学号和年龄；

use EX3

go

select Student.Sno,Sage

from Student

where Sdept<>'EN' and Sage<any(

select Sage

from Student

where Sdept='EN'

)

1. 查询比李晨所选课程成绩都低的学生的学号、姓名、课程名和成绩；

select Student.Sno,Sname,Cname,Grade

from Student,SC,Course

where Grade<(select min(Grade)

from SC,Student

where Sname='李晨' and SC.Sno=Student.Sno)

and SC.Sno=Student.Sno and SC.Cno=Course.Cno



1. 使用集合查询列出IS系的学生以及性别为女的学生名单；

select \*

from Student

where Sdept='IS'

intersect

select \*

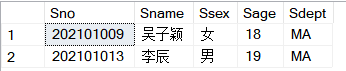
from student

where Ssex='女'

1. 使用集合查询列出MA系的学生与年龄不大于19岁的学生的交集、差集；

select \*

from Student

where Sdept='MA'

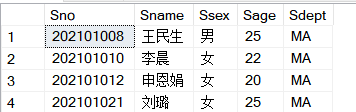
intersect

select \*

from student

where Sage<=19

select \*

from Student

where Sdept='MA'

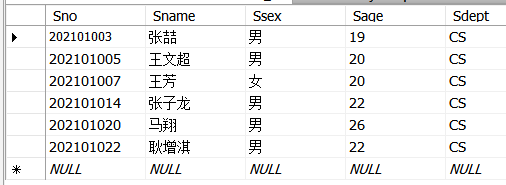
except

select \*

from student

where Sage<=19

1. 建立CS系学生的视图S\_CS；

create view S\_CS

as

select \*

from student

where Sdept='CS'

1. 将IS系的学生的学号、姓名、课程名和成绩定义为一个视图S\_IS；

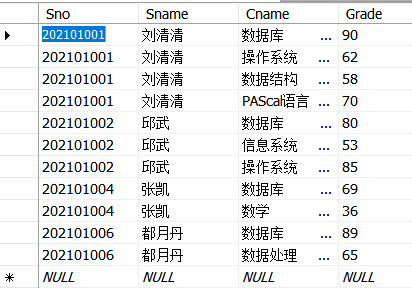
create view S\_IS

as

select Student.Sno,Sname,Cname,Grade

from Student,Course,SC

where Sdept='IS' and Student.Sno=SC.Sno and Course.Cno=SC.Cno



1. 将学生的学号、平均成绩定义成一个视图S\_G;

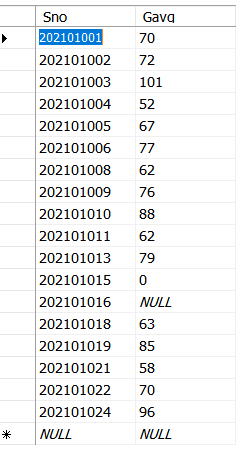
create view S\_G(Sno,Gavg)

as

select Sno,avg(Grade)

from SC

group by Sno



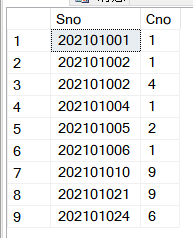
1. 找出每个学生超过他自己选修课程平均成绩的学号及课程号；(视图法)

select Student.Sno,SC.Cno

from S\_G,Student,SC

where Student.Sno=S\_G.Sno and SC.Sno=Student.Sno

and Grade>S\_G.Gavg

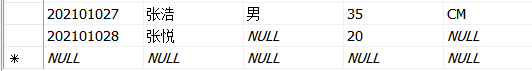


1. 将CS系学生（ ‘202101028’, ‘张悦’, 20）的信息插入到视图S\_CS中。

insert

into S\_CS

values('202101028','张悦',NULL,20,NULL)



**五、出现的问题及解决办法**

1．主要的问题就是前面除视图以外代码的编写，因为在你进行连接的时候，如果有的课虽然开了但是没选，连接以后这门课就会消失，在做某些题时就会得不到想要的结果，只能用嵌套来做。

2. 对某些符号的掌握不牢固，就好比不等于<>这个符号，我原先用的not like就不如<>此符号。

**实验六：数据库的完整性约束**

**一、实验目的和要求**

1. 掌握主键约束、外键约束及check约束的用法；

2. 掌握默认值约束的应用；

3. 了解规则、触发器的使用。

**二、实验环境**

1. 已安装SQL Server 2008企业版的计算机（60台）；

2. 具有局域网网络环境，有固定ip地址。

**三、实验学时**

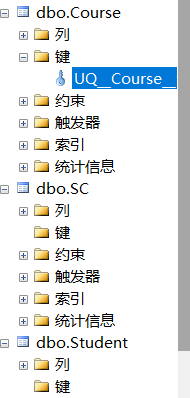
4学时

**四、实验内容及步骤**

以数据库原理实验三数据库中数据为基础，并使用T-SQL语句实现以下操作；

1. 分别删除S-T数据库中Student表、Course表、SC表的主键；

use EX3

go

ALTER TABLE SC

DROP CONSTRAINT PK\_\_SC\_\_E600025309DE7BCC

use EX3

go

ALTER TABLE Course

DROP CONSTRAINT PK\_\_Course\_\_C1FE637303317E3D

use EX3

go

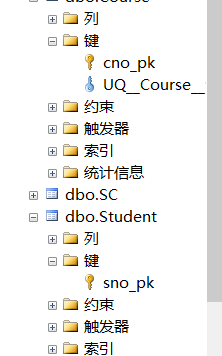
ALTER TABLE Student

DROP CONSTRAINT PK\_\_Student\_\_CA1FE4647F60ED59

1. 分别定义S-T数据库中Student表的主键Sno，约束名为sno\_pk；Course的Cno字段定义为主键，约束名称为cno\_pk;

use EX3

go

ALTER TABLE Student

add constraint sno\_pk primary key(Sno)

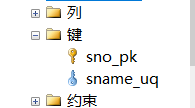
use EX3

go

ALTER TABLE Course

add constraint cno\_pk primary key(Cno)

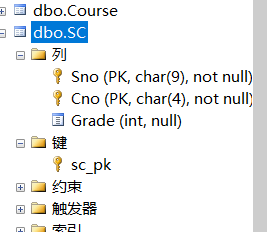
1. 为表Student中的字段Sname添加唯一值约束sname\_uq；

use EX3

go

ALTER TABLE Student

add constraint sname\_uq unique(Sname)

1. 将数据库S-T的表SC的Sno及Cno字段组合定义为主键，约束名称为sc\_pk;

use EX3

go

ALTER TABLE SC

add constraint sc\_pk primary key(Sno,Cno)

1. 对于数据表SC的Sno、Cno字段定义为外码，使之与表Student的主码Sno及表Course的主码Cno对应，实现如下参照完整性：

1)删除Student表中记录的同时删除sc表中与该记录Sno字段值相同的记录；

2)修改Student表某记录的Sno时，若sc表中与该字段值对应的有若干条记录，则拒绝修改；

3)修改Course表Cno字段值时，该字段在SC表中的对应值也应修改；

4)删除Course表一条记录时，若该字段在在SC表中存在，则删除该字段对应的记录；

5)向SC表添加记录时，如果该记录的Sno字段的值在Student中不存在，则拒绝插入；

use EX3

go

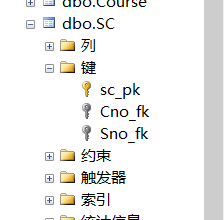
alter table SC

Add constraint Sno\_fk

Foreign key (Sno) references Student (Sno)

on update cascade

on delete cascade



use EX3

go

alter table SC

Add constraint Cno\_fk

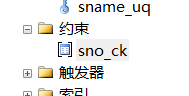
Foreign key (Cno) references Course (Cno)

on update cascade

on delete cascade

1. 定义check约束sno\_ck，要求学生学号Sno必须为9位数字字符，且不能以0开头，第二(三)位皆为0；

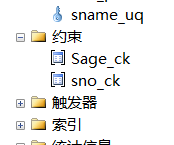
这个题有问题，已存入的数据第三位无法满足为零的情况，所以或有冲突，我把第三位的约束去掉了

use EX3

go

ALTER TABLE Student

add constraint sno\_ck check(Sno not like '0%' and Sno like '\_0\_\_\_\_\_\_\_')

1. 定义S-T数据库中Student表中学生年龄值在16-25范围内；

use EX3

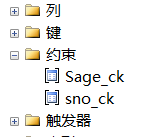
go

ALTER TABLE Student

with nocheck//这点很重要，对于已有数据不满足的情况

add constraint Sage\_ck check(Sage<=25 and Sage >=16)

1. 定义S-T数据库中Student表中学生年龄值在16-31范围内；

use EX3

go

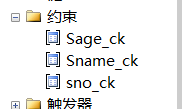
ALTER TABLE Student

drop constraint Sage\_ck

ALTER TABLE Student

with nocheck

add constraint Sage\_ck check(Sage<=16 and Sage >=31)

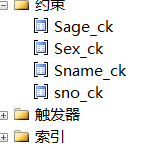
1. 定义S-T数据库中Student表中学生姓名长度在2-8之间；

use EX3

go

ALTER TABLE Student

add constraint Sname\_ck check(len(Sname)>=2 and len(Sname)<=8)

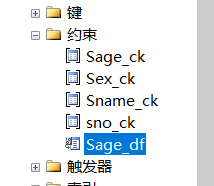
1. 定义S-T数据库中Student表中学生性别列中只能输入“男”或“女”；

use EX3

go

ALTER TABLE Student

add constraint Sex\_ck check(Ssex='男'or Ssex='女')

1. 定义S-T数据库Student表中学生年龄值默认值为20；

use EX3

go

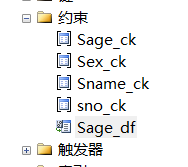
ALTER TABLE Student

add constraint Sage\_df default '20'

for Sage

1. 修改Student表学生的年龄值约束可以为15-35范围内；

use EX3

go

ALTER TABLE Student

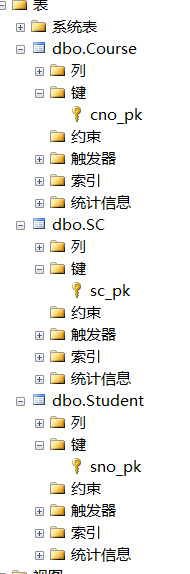
drop constraint Sage\_ck

ALTER TABLE Student

with nocheck

add constraint Sage\_ck check(Sage<=15 and Sage >=35)

1. 删除上述唯一值约束、外键约束及check约束。

use EX3

go

ALTER TABLE Student

drop constraint Sage\_ck,Sex\_ck,Sname\_ck,sno\_ck,Sage\_df,sname\_uq

use EX3

go

ALTER TABLE SC

drop constraint Cno\_fk,Sno\_fk

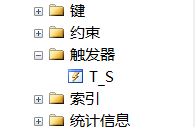
use EX3

go

ALTER TABLE Course

drop constraint UQ\_\_Course\_\_9F5E0299060DEAE8

1. 为Student表创建触发器T\_S，当删除一条学生记录信息时，自动删除该学生的选课记录。

create trigger T\_S

on Student

after delete

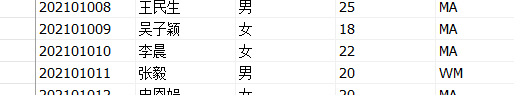
as

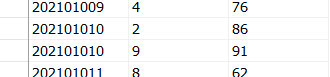
delete from SC

where Sno in

(select Sno from deleted)

1. 删除Student表中“李晨”同学的信息，验证SC表其选课记录是否同时被删除。





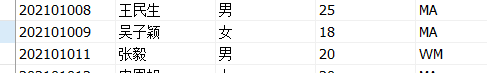
use EX3

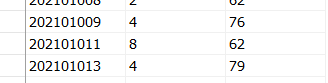
go

delete

from Student

where Sname LIKE '李晨'





**五、出现的问题及解决办法**

1.本次实验出现的主要问题就是在添加约束的操作上，因为本来的数据与约束会有冲突，所以约束会建立不成功，于是我就在每个题的添加约束上加上了with nocheck ，以此来保证成功添加约束。

2.添加默认值这一步骤不是很会，通过自学了解了方法

3.添加触发器这方面，也不是很会，也是自学