# 实验三 数据查询实验

班级：2015211305 姓名：顾溢杰 学号：2015211257

## 实验目的

通过对实验二中建立的数据库关系表和视图的各种查询的操作，加深对SQL语言和Transact SQL查询语言的了解，掌握相关查询语句的语法和使用方法。

## 实验内容

**数据库关系表查询：**

1. 简单的查询操作，包括单表的查询、选择条件、结果排序等的练习；
2. 多表的连接查询，包括等值连接、自然连接等；
3. 复杂的查询操作，包括使用分组函数等库函数的查询操作；
4. 练习带有IN、比较符的嵌套查询。

具体内容包括：

1. **简单查询:**

(1) 查询班号为g00401班的学生的学号和姓名；

(2) 查询“SQL Server数据库开发技术”课程的学分；

(3) 查询选修了课程编号为“dep04\_s003”的学生的学号和成绩，并将成绩按降序输出；(4) 查询学号为“g9940205”的学生选修的课程号和成绩；

(5) 查询选修了课程编号为“dep04\_s001”且成绩高于85分的学生的学号和成绩。

**在多表连接的查询实验中，在SQL　SERVER提供的交互式语言环境下用Transact SQL语句完成以下查询操作：**

(1)查询选修了课程编号为“dep04\_s002”且成绩高于85分的学生的学号、姓名和成绩；

(2) 查询所有学生的学号、姓名、选修的课程名称和成绩；

(3) 查询计算机科学系林红同学选修的课程名称、学分和成绩。(考试成绩>＝60　有学分，否则无学分。**)**

**在复杂查询实验中，在SQL　SERVER提供的交互式语言环境下用Transact SQL语句完成以下查询操作：**

(1)查询至少选修了三门课程的学生的学号和姓名；

(2) 查询选修课程号为“dep04\_b001”的学生的平均成绩；

(3) 查询所有学生的学号和他选修课程的最高成绩，要求他的选修课程中没有成绩为空的；

(4) 查询严为老师2001/2002学年所授“软件开发技术”课程的最高成绩及此学生的学号、姓名、班级；

(5) 查询数据库开发技术课程用过的教材名称，作者和出版社；

(6) 查询计算机科学系讲授过“JAVA程序设计与开发”的老师姓名和职称。

1. **在嵌套查询实验中，在SQL　SERVER提供的交互式语言环境下用Transact SQL语句完成以下查询操作，要求写嵌套查询语句：**

(1)查询选修了“计算机基础”的学生的学号和姓名；

(2) 查询没有选修 “计算机基础”的学生的学号和姓名；

(3) 查询至少选修了学号为“g9940201”的学生所选修的所有课程的学生的学号和姓名。

**建立如下视图：**

学生选修课程信息视图，包括以下内容：

1）学生学号、姓名、所在系、授课老师姓名、课程名称、课程教材名称、出版社、学分、选课成绩

2）修改以上视图，增加学生所在班级信息。

3）对以上视图进行相关的查询操作：

(1) 查询选修了计算机基础的学生的学号和成绩；

(2) 查询所有学生的学号、姓名、选修的课程名称和成绩；

(3) 查询选修了乔红老师课程的学生的学号和姓名、课程名称。

## 实验环境

Windows 10操作系统和Microsoft SQL Server 2008 R2

## 实验步骤、结果及分析

##### 表结构

book表

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 是否可以为NULL |
| [主键]book\_id | varchar(20) | 否 |
| book\_name | varchar(50) | 否 |
| publish\_company | varchar(50) | 否 |
| author | varchar(10) | 否 |
| price | decimal(5,1) | 是 |

course表

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 是否可以为NULL |
| [主键]course\_id | varchar(20) | 否 |
| course\_name | varchar(50) | 否 |
| book\_id | varchar(20) | 是 |
| total\_perior | smallint | 否 |
| week\_perior | smallint | 否 |
| credit | tinyint | 是 |

class表

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 是否可以为NULL |
| [主键]class\_id | varchar(10) | 否 |
| monitor | varchar(10) | 是 |
| classroom | varchar(20) | 否 |
| department\_id | varchar(10) | 否 |

class\_course表

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 是否可以为NULL |
| [主键]class\_id | varchar(10) | 否 |
| [主键]course\_id | varchar(20) | 否 |

department表

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 是否可以为NULL |
| [主键]department\_id | varchar(10) | 否 |
| department\_name | varchar(50) | 否 |
| department\_header | varchar(10) | 是 |
| teacher\_num | smallint | 是 |

student表

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 是否可以为NULL |
| [主键]student\_id | varchar(10) | 否 |
| student\_name | varchar(10) | 否 |
| sex | varchar(2) | 否 |
| birth | date | 是 |
| class\_id | varchar(10) | 否 |
| entrance\_date | date | 否 |
| home\_addr | varchar(200) | 是 |

student\_course表

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 是否可以为NULL |
| [主键]course\_id | varchar(20) | 否 |
| [主键]student\_id | varchar(10) | 否 |
| grade | tinyint | 是 |
| credit | tinyint | 否 |
| semester | tinyint | 否 |
| school\_year | varchar(10) | 否 |

teacher表

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 是否可以为NULL |
| [主键]teacher\_id | varchar(10) | 否 |
| teacher\_name | varchar(10) | 否 |
| sex | varchar(2) | 否 |
| birth | date | 是 |
| department\_id | varchar(10) | 否 |
| profession | varchar(10) | 否 |
| telephone | decimal(15,0) | 否 |
| home\_addr | varchar(200) | 是 |
| postalcode | varchar(10) | 是 |

teacher\_course\_class表

|  |  |  |
| --- | --- | --- |
| 属性名 | 类型 | 是否可以为NULL |
| [主键]teacher\_id | varchar(10) | 否 |
| [主键]course\_id | varchar(20) | 否 |
| [主键]class\_id | varchar(10) | 否 |
| semester | tinyint | 否 |
| school\_year | varchar(10) | 否 |
| course\_schedule | decimal(10,0) | 否 |
| course\_classroom | varchar(10) | 否 |
| book\_id | varchar(20) | 是 |

##### 简单查询

（1）查询班号为g00401班的学生的学号和姓名

**涉及的表:**

student表

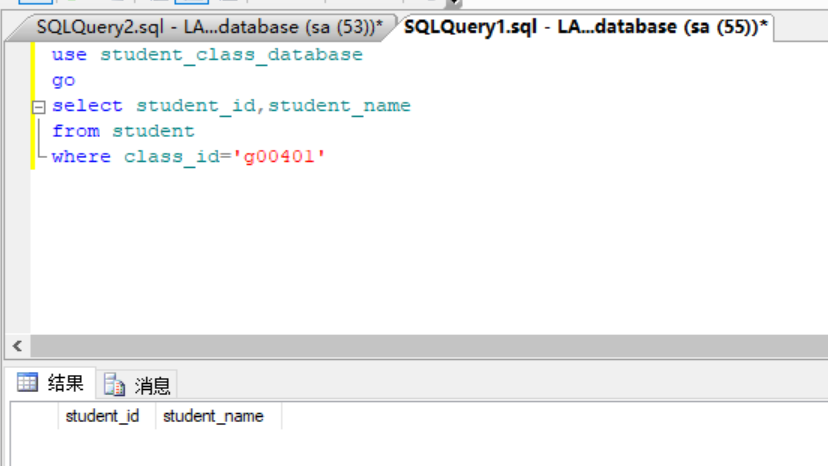
**T-SQL语句：**

select student\_id,student\_name

from student

where class\_id='g00401'

**结果：**



**分析：**

查询到0条记录，说明该数据库中暂未收录班级id为g00401的学生信息。

但在class表中，可以查询到该班级。

（2）查询“SQL Server数据库开发技术”课程的学分

**涉及的表：**

course表

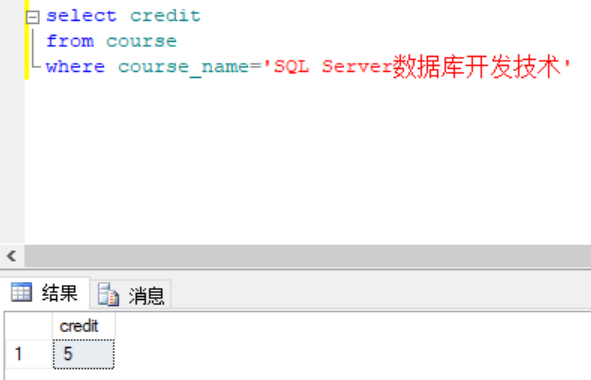
**T-SQL语句：**

select credit

from course

where course\_name='SQL Server数据库开发技术'

**结果：**



**分析：**

查询到该课程的学分为5

（3）查询选修了课程编号为“dep04\_s003”的学生的学号和成绩，并将成绩按降序输出

**涉及的表：**

student\_course表

**T-SQL代码：**

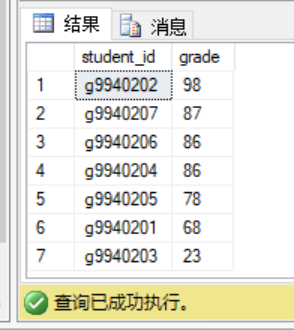
select student\_id,grade

from student\_course

where course\_id='dep04\_s003'

order by grade DESC

**结果：**



**分析：**

共查询到七条记录，按成绩降序排列。

（4）查询学号为“g9940205”的学生选修的课程号和成绩

**涉及的表：**

student\_course表

**T-SQL代码：**

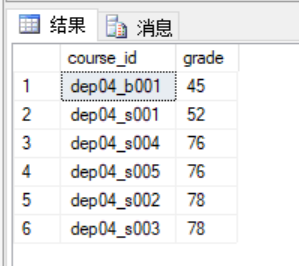
select course\_id,grade

from student\_course

where student\_id='g9940205'

order by grade ASC

**结果：**



**分析：**

该学生选修了六门课，按照成绩升序排列。

（5）查询选修了课程编号为“dep04\_s001”且成绩高于85分的学生的学号和成绩

**涉及的表：**

student\_course

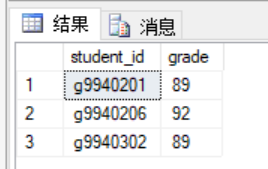
**T-SQL代码：**

select student\_id,grade

from student\_course

where course\_id='dep04\_s001' and grade>85

**结果：**



**分析：**

共有三位选修该课程的学生成绩高于85分。

##### 多表连接的查询

1. 查询选修了课程编号为“dep04\_s002”且成绩高于85分的学生的学号、姓名和成绩

**涉及的表：**

student\_course表、student表

**T-SQL代码：**

select SC.student\_id,S.student\_name,SC.grade

from student AS S,student\_course AS SC

where course\_id='dep04\_s002' and grade>85 and SC.student\_id=S.student\_id

**结果：**



**分析：**

共有两名学生选修了该课程，并且成绩在85分以上。

（2）查询所有学生的学号、姓名、选修的课程名称和成绩

**涉及的表：**

student\_course表、student表、course表

**T-SQL代码：**

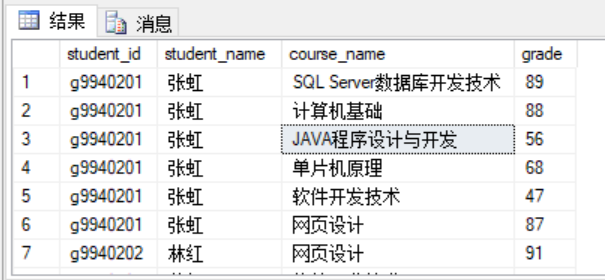
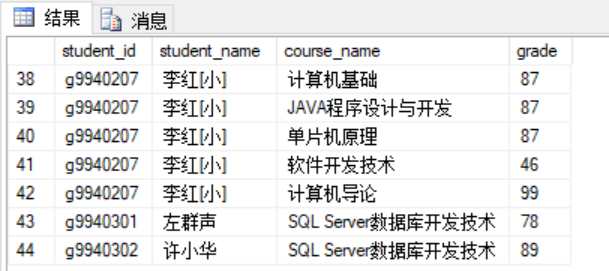
select S.student\_id,S.student\_name,C.course\_name,SC.grade

from student AS S, student\_course AS SC, course AS C

where S.student\_id=SC.student\_id and C.course\_id=SC.course\_id

order by student\_id ASC

**结果：**

**分析：**

共查询到44条记录，与student\_course表的记录相符合。

（3）查询计算机科学系林红同学选修的课程名称、学分和成绩

考试成绩>＝60有学分，否则无学分。

**涉及的表：**

student\_course表、student表、course表

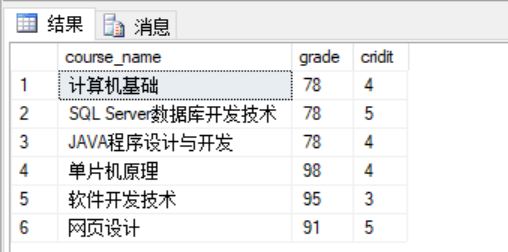
**T-SQL代码：**

select C.course\_name,grade,cridit=case when SC.grade>=60 then SC.credit else 0 end

from student AS S, student\_course AS SC, course AS C

where S.student\_id=SC.student\_id and C.course\_id=SC.course\_id and S.student\_name='林红'

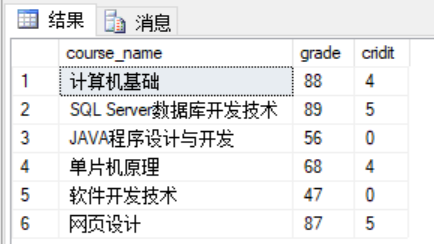
**结果：**



**分析：**

根据分数的不同，对应的学分会有区别，因此需要用到分支语句。

可以看到，林红选修的六门课全部高于60分，所以均取得了对应的学分。我们改为查询张虹的相关信息，结果如下：



可见张虹的“JAVA程序涉及与开发”和“软件开发技术”这两门课均低于60分，对应的学分为0。

查询了一下原始数据，发现在student\_course表中已包含已对学分做出处理，所以只需要普通查询即可，T-SQL代码如下：

select C.course\_name,SC.grade,SC.credit

from student AS S, student\_course AS SC, course AS C

where S.student\_id=SC.student\_id and C.course\_id=SC.course\_id and S.student\_name='林红'

##### 复杂查询实验

（1）查询至少选修了三门课程的学生的学号和姓名

**涉及的表：**

student\_course表、student表

**T-SQL代码：**

select S.student\_id,S.student\_name,course\_number=COUNT(SC.course\_id)

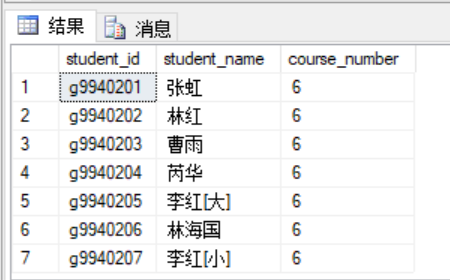
from student\_course as SC,student as S

where S.student\_id=SC.student\_id

group by S.student\_id,S.student\_name

having COUNT(SC.course\_id)>=3

**结果：**



**分析：**

共有七名学生选修了超过三门课，他们均选修了六门课。

（2）查询选修课程号为“dep04\_b001”的学生的平均成绩

**涉及的表：**

student\_course表

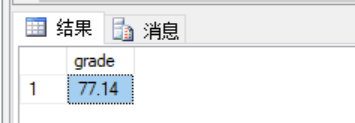
**T-SQL代码：**

select grade=cast(AVG(cast(SC.grade as float)) as numeric(5,2))

from student\_course AS SC

where SC.course\_id='dep04\_b001'

**结果：**



**分析：**

由于student\_course表的grade按照int类型进行存储，所以需要转化为浮点型再求平均值，最终结果保留两位小数。可见选修课程号为“dep04\_b001”的学生的平均成绩约为77.14。

（3）查询所有学生的学号和他选修课程的最高成绩

要求他的选修课程中没有成绩为空

**涉及的表：**

student\_course表

**T-SQL代码：**

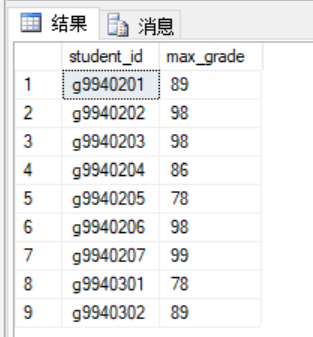
select student\_id,max\_grade=MAX(grade)

from student\_course

group by student\_id

having COUNT(\*)=COUNT(grade)

**结果：**



**分析：**

求最高成绩只需要按照学号分组后运用MAX聚合函数即可，但是要求学生不能包含成绩为NULL的选课记录，因此统计分组后的记录数和grade不为NULL的记录数，如果相等，则说明该学生没有成绩为空的选课记录。

（4）查询严为老师2001/2002学年所授“软件开发技术”课程的最高成绩及此学生的学号、姓名、班级

**涉及的表：**

teacher表、student表、course表、teacher\_course\_class表、student\_course表

**T-SQL代码：**

with target\_course\_id as(

select C.course\_id

from teacher\_course\_class AS TCC, teacher as T, course as C

where TCC.teacher\_id=T.teacher\_id and TCC.course\_id=C.course\_id and

T.teacher\_name='严为' and C.course\_name='软件开发技术' and

TCC.school\_year='2001/2002'

)

select S.student\_id,S.student\_name,S.class\_id

from student\_course as SC, student as S

where SC.student\_id=S.student\_id and SC.school\_year='2001/2002' and SC.course\_id=(select \* from target\_course\_id) and SC.grade>=(

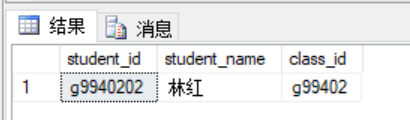
select MAX(grade)

from student\_course as TSC

where TSC.school\_year='2001/2002' and TSC.course\_id=(select \* from target\_course\_id)

)

**结果：**



**分析：**

先利用teacher\_course\_class表、teacher表、course表查出严为老师2001/2002学年所授“软件开发技术”课程的课程id。然后利用该课程id及student\_course表找出2001/2002学年该课程的最高成绩。最后利用student表和student\_course表找出成绩为该最高成绩的学生信息。

1. 查询数据库开发技术课程用过的教材名称，作者和出版社

**涉及的表：**

course表、book表

**T-SQL代码：**

select B.book\_name, B.author, B.publish\_company

from book as B, course as C

where B.book\_id=C.book\_id and C.course\_name='SQL Server数据库开发技术'

**结果：**



**分析：**

数据库开发技术课程用过的教材共一条记录，教材名为《SQL Server数据库开发技术》。

（6）查询计算机科学系讲授过“JAVA程序设计与开发”的老师姓名和职称

**涉及的表：**

teacher\_course\_class、teacher、course、department

**T-SQL代码：**

with target\_departement\_id as(

select department\_id

from department

where department\_name='计算机科学'

)

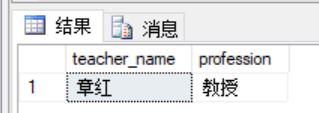
select T.teacher\_name,T.profession

from teacher\_course\_class as TCC, teacher as T, course as C

where T.department\_id=(select \* from target\_departement\_id) and TCC.teacher\_id=T.teacher\_id and

TCC.course\_id=C.course\_id and C.course\_name='JAVA程序设计与开发'

**结果：**



**分析：**

先利用department表找出计算机科学系的department\_id，再利用该department\_id，及teacher\_course\_class、teacher、course表查出最终结果。

##### 嵌套查询

(1) 查询选修了“计算机基础”的学生的学号和姓名

**涉及的表：**

student\_course表、student表

**T-SQL代码：**

select S.student\_id,S.student\_name

from student as S, student\_course as SC

where S.student\_id=SC.student\_id and SC.course\_id=(

select course\_id

from course

where course\_name='计算机基础'

)

**结果：**



**分析：**

子查询先查出名为计算机基础的课程id，然后再利用该id及student\_course表和student表查出最终结果。

(2) 查询没有选修 “计算机基础”的学生的学号和姓名

**涉及的表：**

student\_course表、student表

**T-SQL代码：**

select S.student\_id, S.student\_name

from student as S

except

select S.student\_id, S.student\_name

from student\_course as SC, student as S

where SC.student\_id=S.student\_id and SC.course\_id=(

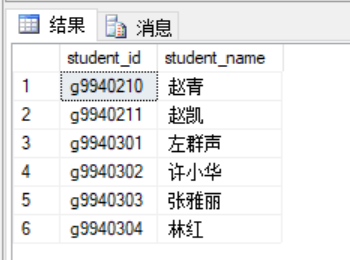
select course\_id

from course

where course\_name='计算机基础'

)

**结果：**



**分析：**

利用（2）中结果和求差集的方法。

(3) 查询至少选修了学号为“g9940201”的学生所选修的所有课程的学生的学号和姓名

**涉及的表：**

student\_course表、student表

**T-SQL代码：**

select distinct S.student\_id, S.student\_name

from student\_course as SC, student as S

where SC.student\_id=S.student\_id and

not exists(

select TSC.course\_id

from student\_course as TSC

where TSC.student\_id='g9940201'

except

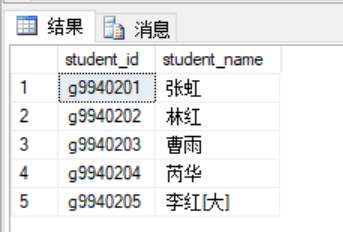
select TTSC.course\_id

from student\_course as TTSC

where TTSC.student\_id=SC.student\_id

)

**结果：**



**分析：**

利用差集判断其它学生的选修课程是否包含g9940201同学的选修课程。我们看到结果中，显示了g9940201这个学生本身。加上SC.student\_id!=’g9940102’的条件即可去掉他本身。

##### 视图查询实验

1. 建立视图

包含学生学号、姓名、所在系、授课老师姓名、课程名称、课程教材名称、出版社、学分、选课成绩

**涉及的表：**

student表、student\_course表、course表、book表、class表、department表、teacher\_course\_class表、teacher表

**T-SQL代码：**

CREATE VIEW my\_view

AS

SELECT S.student\_id, S.student\_name, D.department\_name, T.teacher\_name, CO.course\_name, B.book\_name, B.publish\_company, SC.credit, SC.grade

FROM student as S join student\_course as SC on S.student\_id=SC.student\_id

join course as CO on SC.course\_id=CO.course\_id

join book as B on CO.book\_id=B.book\_id

join class as CL on S.class\_id=CL.class\_id

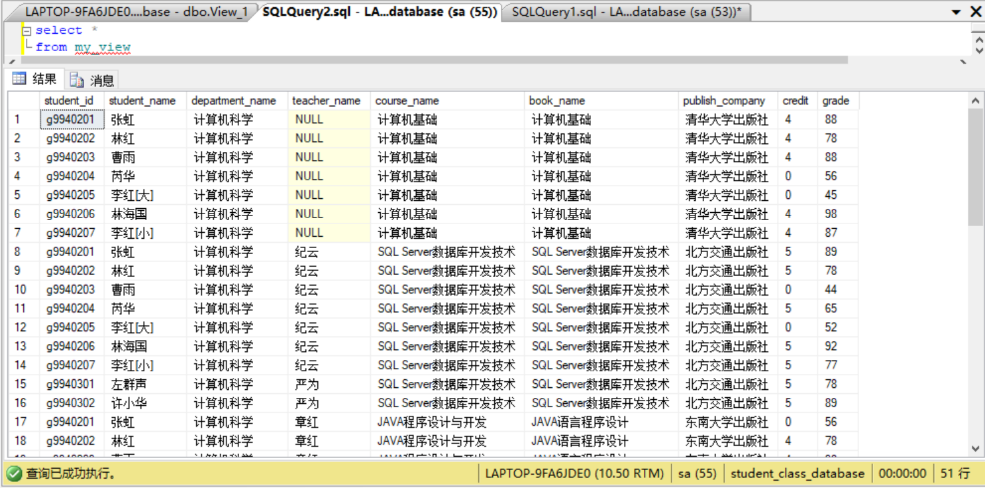
join department as D on CL.department\_id=D.department\_id

left outer join teacher\_course\_class as TCC on SC.course\_id=TCC.course\_id and TCC.class\_id=CL.class\_id

left outer join teacher as T on TCC.teacher\_id=T.teacher\_id

GO

**结果：**



**分析：**

由于g99402班级的dep04\_s006、dep04\_s007、dep04\_b001课程没有教师信息，因此必须采用外连接的方式，才可以防止信息丢失。

共查到51条记录，但是student\_course实际上只有44条记录，多出来7条记录，这是因为在teacher\_course\_class表中，g99402班级的dep04\_s004课程有两位教师的信息，这使得g99402班选修了该门课程的七名同学同时选修了这两位教师开设的同一门课程，我们也注意到teacher\_course\_class表中，这两位教师的上课schedule是不同的，因此可以推测这两位老师共同/轮流给这个班的学生上该门课程，这在现实中也是可能出现的。

（2）修改视图

增加学生所在班级信息。

**涉及的表：**

student表、student\_course表、course表、book表、class表、department表、teacher\_course\_class表、teacher表

**T-SQL代码：**

ALTER VIEW my\_view

AS

SELECT S.student\_id, S.student\_name, CL.class\_id, D.department\_name, T.teacher\_name, CO.course\_name, B.book\_name, B.publish\_company, SC.credit, SC.grade

FROM student as S join student\_course as SC on S.student\_id=SC.student\_id

join course as CO on SC.course\_id=CO.course\_id

join book as B on CO.book\_id=B.book\_id

join class as CL on S.class\_id=CL.class\_id

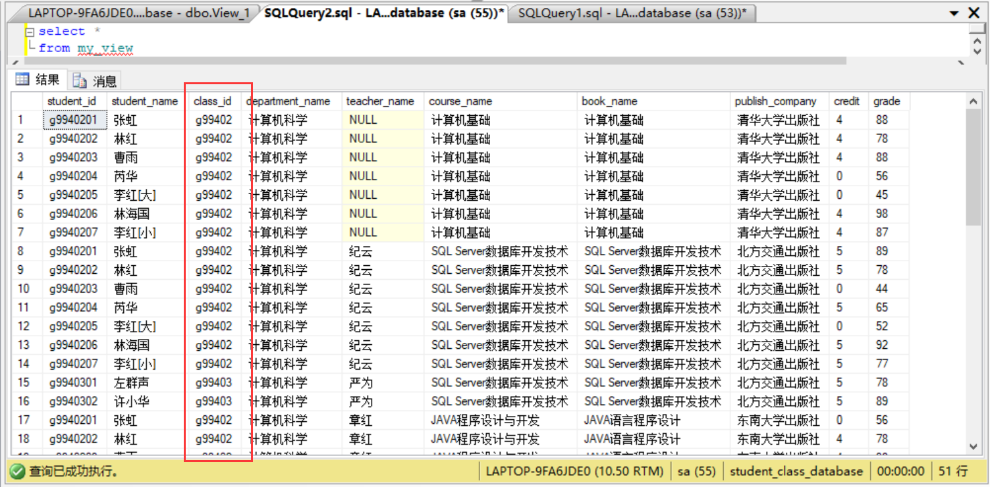
join department as D on CL.department\_id=D.department\_id

left outer join teacher\_course\_class as TCC on SC.course\_id=TCC.course\_id and TCC.class\_id=CL.class\_id

left outer join teacher as T on TCC.teacher\_id=T.teacher\_id

GO

**结果：**



**分析：**

修改视图的语法与创建视图的语法类似，只需要将CREATE关键字改为ALTER即可。另外，在视图的属性中新增了班级id，由于之间已经将class表连接，所以只需要在SELECT字段增加class\_id即可。

（3）查询选修了计算机基础的学生的学号和成绩

**涉及的视图：**

my\_view

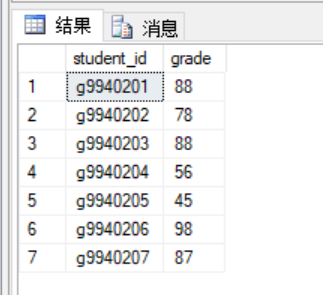
**T-SQL代码：**

select student\_id,grade

from my\_view

where course\_name='计算机基础'

**结果：**



**分析：**

共查到七名学生的选课成绩

（4）查询所有学生的学号、姓名、选修的课程名称和成绩

**涉及的视图：**

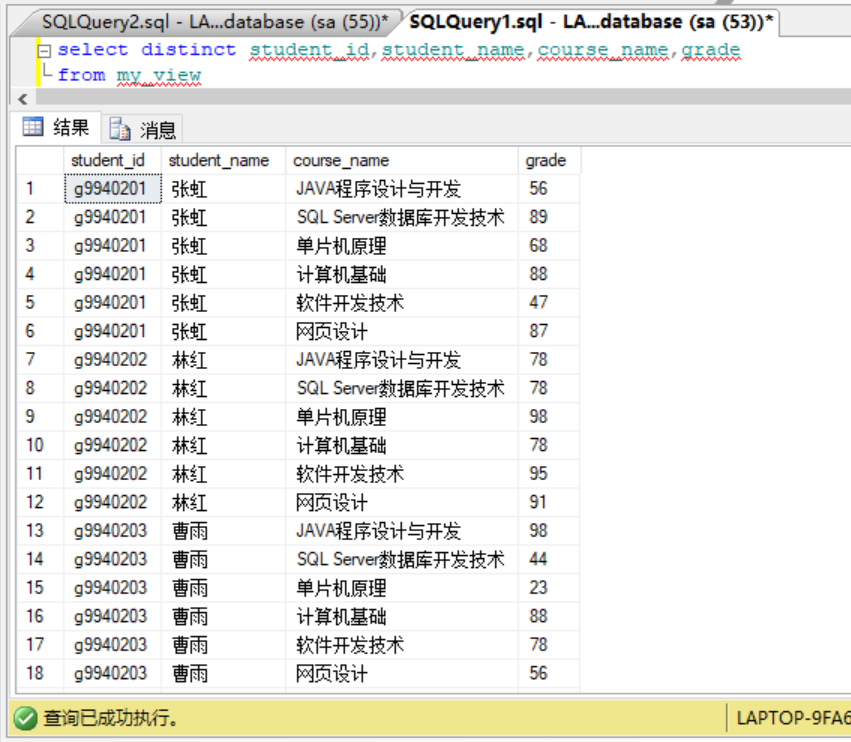
my\_view

**T-SQL代码：**

select distinct student\_id,student\_name,course\_name,grade

from my\_view

**结果：**



**分析：**

由于存在同一个班的同一门课有多名老师授课的情况，所以需要加上disticnt关键字，共查询到44条结果。

（5）查询选修了乔红老师课程的学生的学号和姓名、课程名称

**涉及的视图：**

my\_view

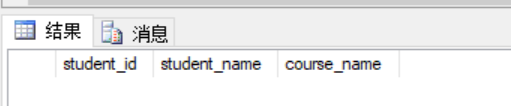
**T-SQL代码：**

select student\_id,student\_name,course\_name

from my\_view

where teacher\_name='乔红'

**结果：**



**分析：**

暂无选修了乔红老师课程的学生记录。

## 实验总结

本次实验的查询语句由浅入深，从简单到复杂，从单一到嵌套，一步一步地引导我掌握了SQL查询语句的编写方法，同时也学会了视图的创建和修改，以及在视图中进行查询的方法。