

实验二 SQL 数据查询

XX 班 100002000001 许愿

【实验目的】

能熟练地利用 SQL 查询语句进行单表数据查询和数据排序处理、利用 SQL 查询语句进行多表之间的数据联结查询、嵌套查询及组合查询等。

【实验环境，主要仪器及软件】

Windows 11, SQL Server 2022 + SQL Server Management Studio 20 / Navicat

【实验内容】

创建表格并使用 SELECT 语句完成所要求的查询。

【实验步骤】

(1) 在实验用数据库 exp2 中创建实验所需的表，表内容如下：

TEACHER 表

NO	NAME	SEX	BIRTHDAY	PROF	DEPART
804	李城	男	12/02/58	副教授	计算机系
856	张旭	男	03/12/69	讲师	电子工程系
825	王萍	女	05/05/72	助教	计算机系
831	刘冰	女	08/14/77	助教	电子工程系

COURSES 表

CNO	CNAME	TNO
3-105	计算机导论	825
3-245	操作系统	804
6-166	数字电路	856
9-888	高等数学	856

SCORE 表

NO	CNO	DEGREE
103	3-245	86
105	3-245	75

109	3-245	68
101	3-105	92
107	3-105	88
109	3-105	76
108	3-105	78
101	6-166	85
107	6-166	79
108	6-166	81


STUDENTS 表

NO	NAME	SEX	BIRTHDAY	CLASS
108	曾华	男	09/01/77	95033
105	李明	男	10/02/75	95031
107	王利	女	01/23/76	95033
109	李军	男	02/20/76	95033
103	王芳	女	02/10/75	95031
101	陆君	男	06/03/74	95031

最终效果如图所示（粗略配置，供题目使用。由于此表无法满足所有查询要求，实验中部分需要单独配置表的查询会单独列出使用的表信息）：

对象	TEACHER @ex...	TEACHER @ex...	COURSES @ex...	COURSES @ex...	SCORE @exp2...	SCORE @ex...
保存	添加字段	删除字段	主键			
字段	索引	外键	唯一键	检查	触发器	选项
名	类型	大小	比例	不是 null	键	注释
NO	int			<input checked="" type="checkbox"/>	1	
NAME	char	10		<input type="checkbox"/>		
SEX	char	2		<input type="checkbox"/>		
BIRTHDAY	date			<input type="checkbox"/>		
PROF	char	10		<input type="checkbox"/>		
DEPART	char	10		<input type="checkbox"/>		

对象	TEACHER @ex...	TEACHER @ex...	COURSES @ex...	COURSES @ex...	SCO
开始事务	文本	筛选	排序	列	导入
导出	数据生成	创建图表			
NO	NAME	SEX	BIRTHDAY	PROF	DEPART
804	李城	男	1958-12-02	副教授	计算机系
825	王萍	女	1972-05-05	助教	计算机系
831	刘冰	女	1977-08-14	助教	电子工程系
856	张旭	男	1969-03-12	讲师	电子工程系

对象		TEACHER @ex...	TEACHER @ex...	COURSES @ex...	COURSES @ex...	SCORE @exp2...	SCOR		
保存		添加字段		删除字段		主键			
字段	索引	外键	唯一键	检查	触发器	选项	存储	注释	SQL 预览
名	类型				大小	比例	不是 null	键	注释
▶ CNO	char				10		<input checked="" type="checkbox"/>	 1	
CNAME	char				10		<input type="checkbox"/>		
TNO	int						<input type="checkbox"/>		

对象

TEACHER @ex...

TEACHER @ex...

COURSES @ex...

COURSES @ex...

SCORE @exp2...

SCORE @exp2...

STUDENTS @e

保存

添加外键

删除外键

字段

索引

外键

唯一键

检查

触发器

选项

存储

注释

SQL 预览

名	字段	被引用的数据库	被引用的表 (父)	被引用的字段	删除时	更新时	启用	不适用于复制
tno	TNO	dbo	TEACHER	NO	NO ACTION	NO ACTION	<input checked="" type="checkbox"/>	<input type="checkbox"/>

对象

TEACHER @ex...

TEACHER @ex...

COURSES @ex...

COURSES @ex...

开始事务

文本

筛选

排序

列

导入

导出

数据生成

创建图表

CNO	CNAME	TNO
3-105	计算机导论	825
3-245	操作系统	804
6-166	数字电路	856
9-888	高等数学	856

对象

TEACHER @ex...

TEACHER @ex...

COURSES @ex...

COURSES @ex...

SCORE @exp2...

S

保存

添加字段

删除字段

主键

字段

索引

外键

唯一键

检查

触发器

选项

存储

注释

SQL 预览

名	类型	大小	比例	不是 null	键	注释
NO	int			<input checked="" type="checkbox"/>	1	
CNO	char	10		<input checked="" type="checkbox"/>	2	
DEGREE	int			<input type="checkbox"/>		

开始事务

文本

筛选

排序

列

导入

导出

NO	CNO	DEGREE
101	3-105	92
101	6-166	85
103	3-245	86
105	3-245	75
107	3-105	88
107	6-166	79
108	3-105	78
108	6-166	79
109	3-105	76
109	3-245	68

对象	TEACHER @ex...	TEACHER @ex...	COURSES @ex...	COURSES @ex...	SCORE @exp2....	SC
保存	添加字段	删除字段	主键			
字段	索引	外键	唯一键	检查	触发器	选项
名	类型	大小	比例	不是 null	键	注释
NO	int			<input checked="" type="checkbox"/>	1	
NAME	char	10		<input type="checkbox"/>		
SEX	char	2		<input type="checkbox"/>		
BIRTHDAY	date			<input type="checkbox"/>		
CLASS	char	10		<input type="checkbox"/>		

对象	TEACHER @ex...	TEACHER @ex...	COURSES @ex...	C
开始事务	文本	筛选	排序	列
导入	导出	数据生		
NO	NAME	SEX	BIRTHDAY	CLASS
101	陆君	男	1974-06-03	95031
103	王芳	女	1975-02-10	95031
105	李明	男	1975-10-02	95031
107	王利	女	1976-01-23	95033
108	曾华	男	1977-09-01	95033
109	李军	男	1976-02-20	95033

(2) 完成查询

1、显示 SCORE 表中成绩在 60 到 80 之间的所有记录

SQLQuery1.sql -...(XUAN\Xuan (54))*

```
use exp2;  
  
select * from score where degree between 60 and 80
```

100 %

结果 消息

	NO	CNO	DEGREE
1	105	3-245	75
2	107	6-166	79
3	108	3-105	78
4	108	6-166	79
5	109	3-105	76
6	109	3-245	68

2、显示 SCORE 表中成绩为 85, 86 或 88 的记录

SQLQuery1.sql -...(XUAN\Xuan (54))*

```
use exp2;
select * from score where degree in (85,86,88)
```

100 %

结果 消息

	NO	CNO	DEGREE
1	101	6-166	85
2	103	3-245	86
3	107	3-105	88

3、显示“95031”班或性别为“女”的同学记录

SQLQuery1.sql -...(XUAN\Xuan (54))*

```
use exp2;
select * from students where class='95031' or sex='女'
```

100 %

结果 消息

	NO	NAME	SEX	BIRTHDAY	CLASS
1	101	陆君	男	1974-06-03	95031
2	103	王芳	女	1975-02-10	95031
3	105	李明	男	1975-10-02	95031
4	107	王利	女	1976-01-23	95033

4、以 CLASS 降序显示 STUDENT 表的所有记录

SQLQuery1.sql -...(XUAN\Xuan (54))*

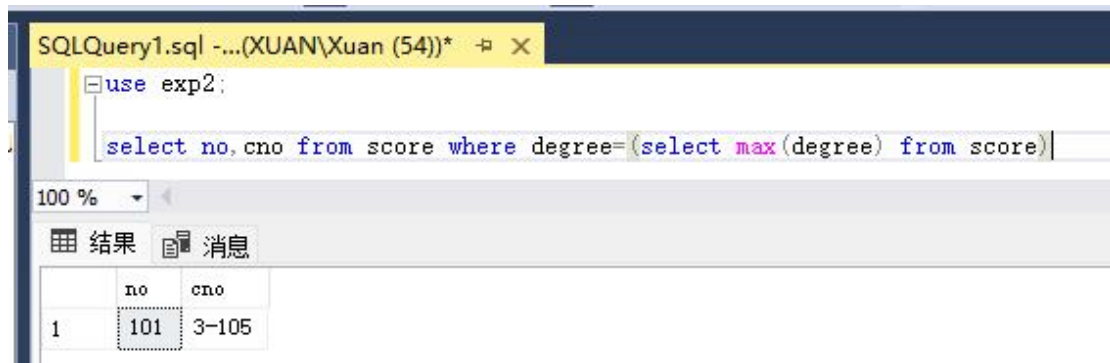
```
use exp2;
select * from students order by class desc
```

100 %

结果 消息

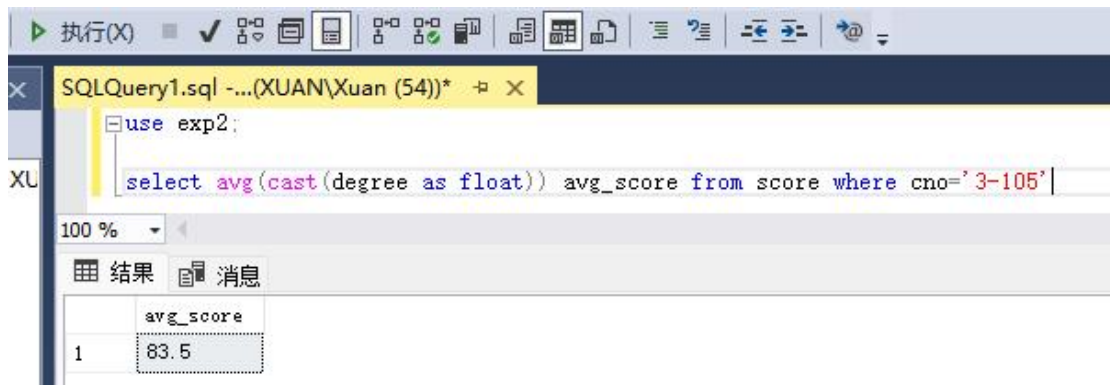
	NO	NAME	SEX	BIRTHDAY	CLASS
1	107	王利	女	1976-01-23	95033
2	108	曾华	男	1977-09-01	95033
3	109	李军	男	1976-02-20	95033
4	101	陆君	男	1974-06-03	95031
5	103	王芳	女	1975-02-10	95031
6	105	李明	男	1975-10-02	95031

5、显示 SCORE 表中最高分的学生学号和课程号



6、显示 3-105 号课程的平均分

此处由于原列 DEGREE 的数值类型为 INT, 在计算均分时会自动转化导致所求的均分被取整。此处使用 CAST()函数将分数转化为浮点数, 由此得出 83.5 的精确均分。(若不进行转化, 所求的均分为 83)



7、显示 SCORE 表中至少有 5 名学生选修的并以 3 开头的课程号的平均分

(此处改变 SCORE 表的内容使有课程被 5 名学生选修, 新增条目已选中)

对象 SCORE @exp2.dbo (本地SQL) - 表

开始事务 文本 筛选 排序 列

NO	CNO	DEGREE
101	3-105	92
101	6-166	85
103	3-245	86
105	3-245	75
107	3-105	88
107	6-166	79
108	3-105	78
108	6-166	79
109	3-105	76
109	3-245	68
103	3-105	100

SQLQuery1.sql -...(XUAN\Xuan (54))*

```
use exp2;
select cno, avg(cast(degree as float)) avg_score from score group by cno having count(*)>=5 and cno like '3%'
```

100 %

结果 消息

cno	avg_score
3-105	86.8

8、显示最低分大于 70，最高分小于 90 的 NO 列

SQLQuery1.sql -...(XUAN\Xuan (54))*

```
use exp2;
select no from score group by no having max(degree)<90 and min(degree)>70
```

100 %

结果 消息

	no
1	103
2	105
3	107
4	108

9、显示所有同学的 NAME， CNAME 和 DEGREE 列

SQLQuery1.sql -...(XUAN\Xuan (54))*

```

use exp2;

select name, cname, degree
from score, students, courses
where score.cno=courses.cno and score.no=students.no

```

100 %

结果 消息

	name	cname	degree
1	陆君	计算机导论	92
2	陆君	数字电路	85
3	王芳	操作系统	86
4	李明	操作系统	75
5	王利	计算机导论	88
6	王利	数字电路	79
7	曾华	计算机导论	78
8	曾华	数字电路	79
9	李军	计算机导论	76
10	李军	操作系统	68

10、显示 95033 班所选课程的平均分

* 此处产生歧义：是所有 95033 班学生所选课程的总平均分，还是所有 95033 班学生有选择的课程的均分？

所有 95033 班学生所选课程的总平均分

SQLQuery1.sql -...(XUAN\Xuan (54))*

```

use exp2;

select avg(cast(degree as float)) total_avg
from score where no in (select no from students where class='95033')

```

100 %

结果 消息

	total_avg
1	78

所有 95033 班学生有选择的课程的均分

SQLQuery1.sql -...(XUAN\Xuan (54))*

```

use exp2;

select cno, avg(cast(degree as float)) avg
from score where no in (select no from students where class='95033') group by cno

```

100 %

结果 消息

	cno	avg
1	3-105	80.6666666666667
2	3-245	68
3	6-166	79

11、显示选修 3-105 课程的成绩高于 109 号同学成绩的所有同学的记录

SQLQuery1.sql -...(XUAN\Xuan (54))*

```

use exp2;

select * from score
where cno='3-105' and degree>(select degree from score where no=109 and cno='3-105')

```

100 %

结果 消息

	NO	CNO	DEGREE
1	101	3-105	92
2	107	3-105	88
3	108	3-105	78

12、显示 SCORE 中选学多门课程的同学中分数为非最高分成绩的记录（两种方法）

第一种方法（COUNT 函数可以去除，因为排除掉了最高分即排除掉了最大值）

SQLQuery1.sql -...(XUAN\Xuan (54))*

```

use exp2;

select score.*
from score, (select no,max(degree) max_score from score where no=score.no group by no having count(*)>=2) as max_table
where score.no=max_table.no and score.degree!=max_table.max_score

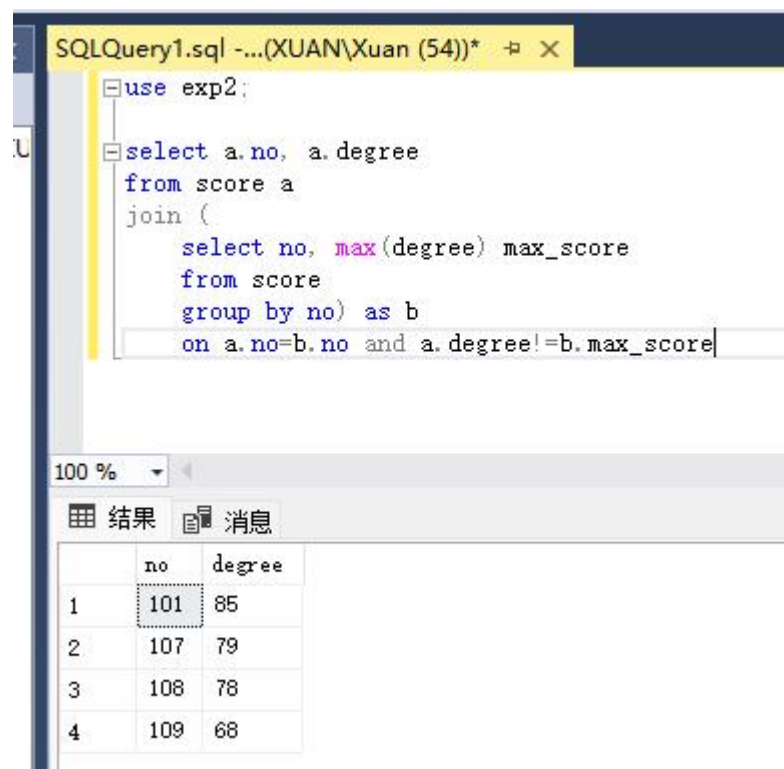
```

100 %

结果 消息

	NO	CNO	DEGREE
1	101	6-166	85
2	107	6-166	79
3	108	3-105	78
4	109	3-245	68

第二种方法



The screenshot shows a SQL query window titled 'SQLQuery1.sql' with the following code:

```
use exp2;  
  
select a.no, a.degree  
from score a  
join (  
    select no, max(degree) max_score  
    from score  
    group by no) as b  
on a.no=b.no and a.degree!=b.max_score
```

Below the query window, the 'Results' tab is active, displaying the following data:

	no	degree
1	101	85
2	107	79
3	108	78
4	109	68

13、查询同名同姓的学生学号和姓名

(此处改变 STUDENTS 表的内容使有学生同名，修改条目已选中)

对象

SCORE @exp2.dbo (本地SQL) - 表

STUDENTS @exp2.dbo (本地SQL) - 表

开始事务

文本

筛选

排序

列

导入

导出

数据生成

创建图表

NO	NAME	SEX	BIRTHDAY	CLASS
101	陆君	男	1974-06-03	95031
103	王芳	女	1975-02-10	95031
105	李明	男	1975-10-02	95031
107	王利	女	1976-01-23	95033
108	曾华	男	1977-09-01	95033
109	王利	男	1976-02-20	95033

SQLQuery1.sql -...(XUAN\Xuan (54))*

```

use exp2;
select no,name from students where name in
(select name from students group by name having count(*)>=2)

```

100 %

结果 消息

	no	name
1	107	王利
2	109	王利

14、显示“张旭”老师任课的学生成绩

SQLQuery1.sql -...(XUAN\Xuan (54))*

```

use exp2;
select * from score where cno in
(select cno from courses where tno=(select no from teacher where name='张旭'))

```

100 %

结果 消息

	NO	CNO	DEGREE
1	101	6-166	85
2	107	6-166	79
3	108	6-166	79

15、显示选修某课程的同学人数多于 5 人的教师姓名

(此处改变 SCORE 表的内容使有课程被 5 名学生选修，新增条目已选中)

对象 SCORE @exp2.dbo (本地SQL) - 表

开始事务 文本 筛选 排序 列

NO	CNO	DEGREE
101	3-105	92
101	6-166	85
103	3-245	86
105	3-245	75
107	3-105	88
107	6-166	79
108	3-105	78
108	6-166	79
109	3-105	76
109	3-245	68
103	3-105	100

SQLQuery1.sql -...(XUAN\Xuan (54))*

```

use exp2;
select name from teacher where no in
(select tno from courses where cno in
(select cno from score group by cno having count(*)>=5))

```

100 %

结果 消息

name
1 王萍

16、列出 95033 班和 95031 班全体学生的记录

执行(X) 运行 停止 刷新 保存 打印 窗口 帮助

SQLQuery1.sql -...(XUAN\Xuan (54))*

```

use exp2;
select * from students where class in (95033,95031)

```

100 %

结果 消息

	NO	NAME	SEX	BIRTHDAY	CLASS
1	101	陆君	男	1974-06-03	95031
2	103	王芳	女	1975-02-10	95031
3	105	李明	男	1975-10-02	95031
4	107	王利	女	1976-01-23	95033
5	108	曾华	男	1977-09-01	95033
6	109	李军	男	1976-02-20	95033

17、列出“计算机系”教师所教课程的成绩表

SQLQuery1.sql -...(XUAN\Xuan (54))*

```
use exp2;  
select * from score where cno in  
(select cno from courses where tno in  
(select no from teacher where depart='计算机系'))
```

100 %

结果 消息

	NO	CNO	DEGREE
1	101	3-105	92
2	103	3-245	86
3	105	3-245	75
4	107	3-105	88
5	108	3-105	78
6	109	3-105	76
7	109	3-245	68

18、列出“计算机系”与“电子工程系”不同职称的教师的 NAME 和 PROF

SQLQuery1.sql -...(XUAN\Xuan (54))*

```
use exp2;  
select teacher.name, teacher.prof from teacher  
where prof != (select a.prof from teacher a,teacher b  
where a.depart='电子工程系' and b.depart='计算机系' and a.prof=b.prof)  
and depart in ('计算机系','电子工程系')
```

100 %

结果 消息

	name	prof
1	李城	副教授
2	张旭	讲师

19、列出编号为“3-105”课程且成绩至少高于选修编号为“3-245”的同学 CNO、NO 和 DEGREE，并按 DEGREE 从高到低排序

SQLQuery1.sql -...(XUAN\Xuan (54))*

```

use exp2;
select a.cno, a.no, a.degree
from score a, score b
where a.no=b.no and a.cno='3-105' and b.cno='3-245' and a.degree>b.degree
order by degree desc

```

100 %

结果 消息

	cno	no	degree
1	3-105	109	76

20、列出所有教师 and 同学的 NAME 、SEX 和 BIRTHDAY

SQLQuery1.sql -...(XUAN\Xuan (54))*

```

use exp2;
select '教师' type, name, sex, birthday from teacher
union(select '学生' type, name, sex, birthday from students)

```

100 %

结果 消息

	type	name	sex	birthday
1	教师	李城	男	1958-12-02
2	教师	刘冰	女	1977-08-14
3	教师	王萍	女	1972-05-05
4	教师	张旭	男	1969-03-12
5	学生	李军	男	1976-02-20
6	学生	李明	男	1975-10-02
7	学生	陆君	男	1974-06-03
8	学生	王芳	女	1975-02-10
9	学生	王利	女	1976-01-23
10	学生	曾华	男	1977-09-01

21、显示所有任课老师 the name and depart

SQLQuery1.sql -...(XUAN\Xuan (54))*

```

use exp2;
select name, depart from teacher
where no in (select tno from courses)

```

100 %

结果 消息

	name	depart
1	李城	计算机系
2	王萍	计算机系
3	张旭	电子工程系

22、显示所有未讲课的老师的 name 和 depart



SQLQuery1.sql -...(XUAN\Xuan (54))*

```
use exp2;  
  
select name,depart from teacher  
where no not in (select tno from courses)
```

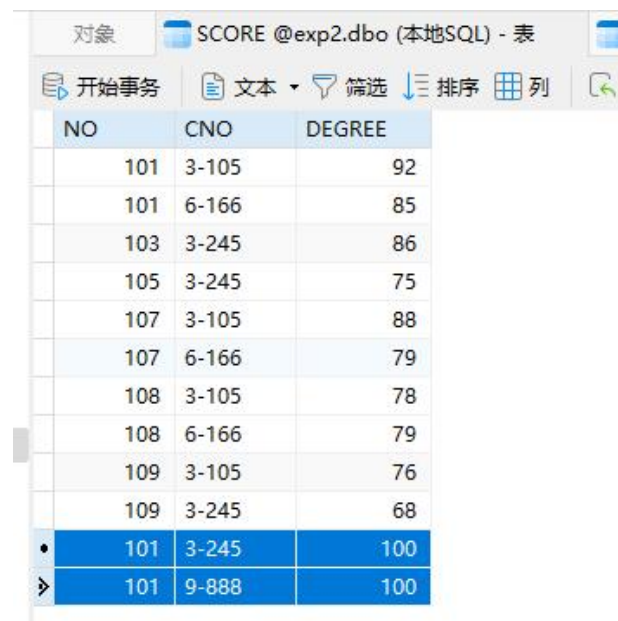
100 %

结果 消息

	name	depart
1	刘冰	电子工程系

23、查询选修所有课程的学生学号和姓名（两种方法）

(此处改变 SCORE 表的内容使有学生选修所有课程，新增条目已选中)

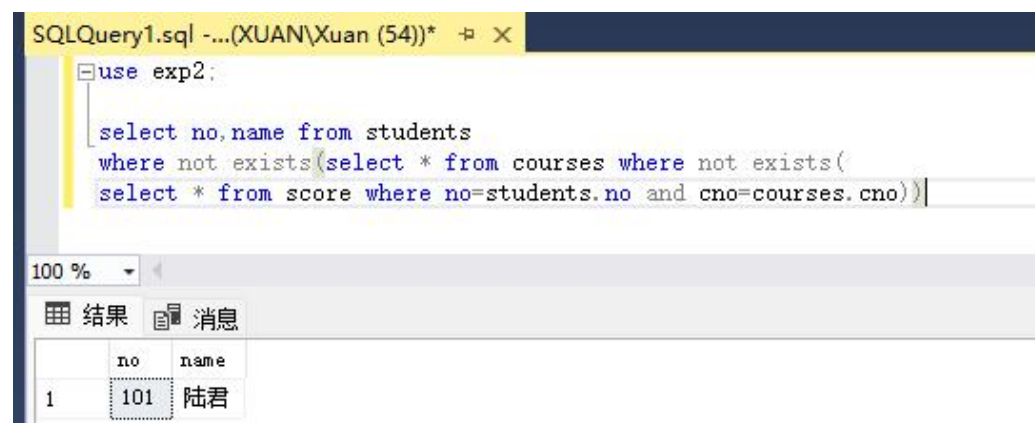


对象 SCORE @exp2.dbo (本地SQL) - 表

开始事务 文本 筛选 排序 列

NO	CNO	DEGREE
101	3-105	92
101	6-166	85
103	3-245	86
105	3-245	75
107	3-105	88
107	6-166	79
108	3-105	78
108	6-166	79
109	3-105	76
109	3-245	68
101	3-245	100
101	9-888	100

方法一



SQLQuery1.sql -...(XUAN\Xuan (54))*

```
use exp2;  
  
select no,name from students  
where not exists(select * from courses where not exists(  
select * from score where no=students.no and cno=courses.cno))
```

100 %

结果 消息

	no	name
1	101	陆君

方法二

SQLQuery1.sql -...(XUAN\Xuan (54))*

```
use exp2;  
select no,name from students  
where no in (  
select no from score group by no having count(cno)=(select count(*) from courses))
```

100 %

结果 消息

	no	name
1	101	陆君