Lecture 3， Fall 2017/2018

数据库系统实验

Yubao Liu (刘玉葆)
School of Data and Computer Science
Sun Yat-sen University

- 本节课提纲

  - 实验目的
  - 实验内容
  - 实验示例
  - 练习

- 实验目的

熟悉SQL的**数据查询语言**，

能够使用SQL语句对数据库进行**单表查询、连接查询**

- 实验内容

1）单表查询

- 查询的目标表达式为**所有列**、**指定列**或**指定列的运算**。
- 用 **DISTINCT**保留字消除重复行
- 对查询结果**排序**和**分组**。
- **集合分组**使用**集函数**进行各项统计

2)**连接查询**

- 笛卡儿连接和**等值连接**。
- 自连接
- **外连接**
- 复合条件连接
- 多表连接

- # 实验示例

以 school数据库为例，在该数据库中存在4张表格，分别为

- students (sid, sname , email , grade)
- teachers (tid, tname , email , salary)
- courses (cid, cname , hour)
- choices (no, sid , tid , cid , score)

在数据库中，存在这样的关系，学生可以选择课程。一个课程对应一个教师。

在CHOICES表中保存学生的选课记录。

- 实验示例

一. 单表查询

1.查询年级为2001所有学生的名称，按编号升序排列。

SQLQuery2.sql - (...BRTDQO\dgn (53))*

```
1  SELECT SNAME
2  FROM STUDENTS
3  WHERE GRADE='2001'
4  ORDER BY SID
```

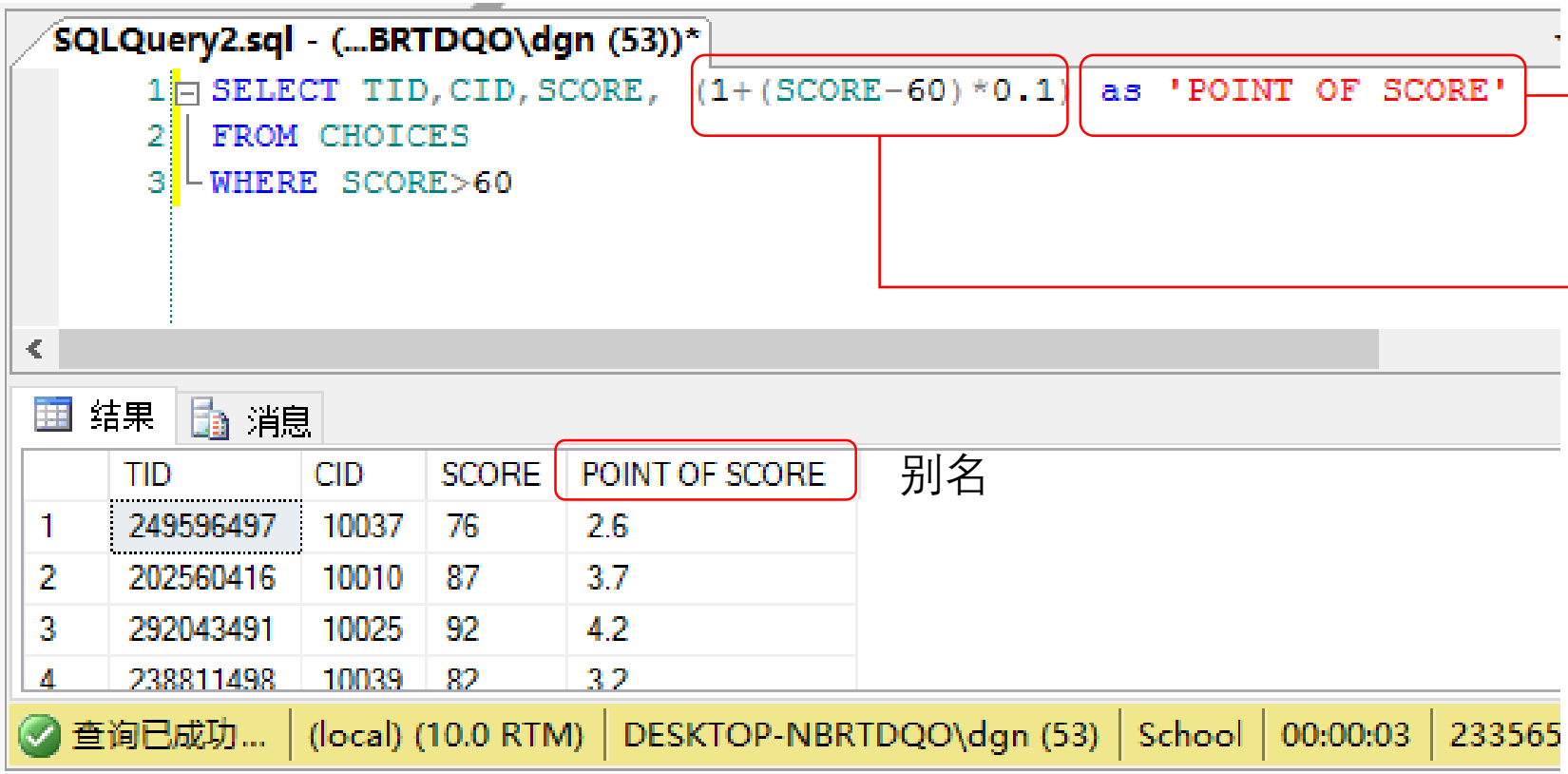
结果 消息

	TID	CID	SCORE	POINT OF SCORE
1	249596497	10037	76	2.6
2	202560416	10010	87	3.7
3	292043491	10025	92	4.2
4	238811498	10039	82	3.2

查询已成功... | (local) (10.0 RTM) | DESKTOP-NBRTDQO\dgn (53) | School | 00:00:03 | 233565 行

- # 实验示例

2. 查询学生的选课成绩合格的课程成绩，并把成绩换算为积点。（60分对应积点1，每增加1分，积点增加0.1）

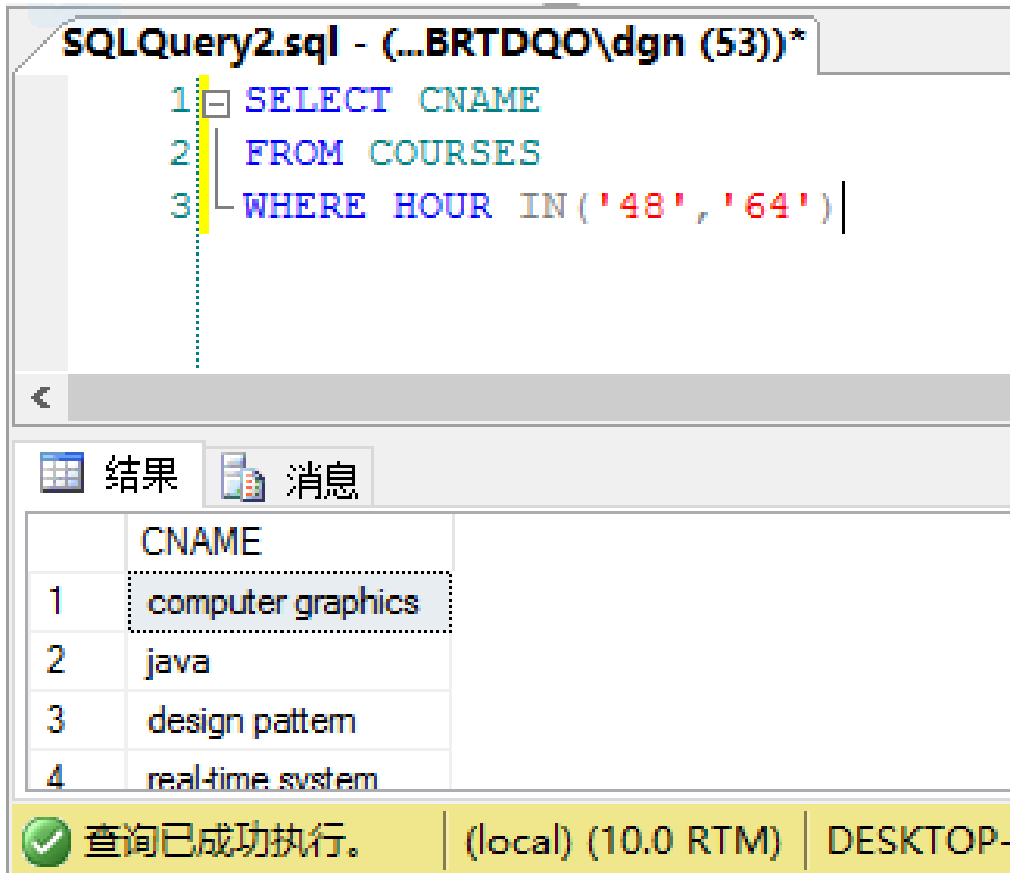


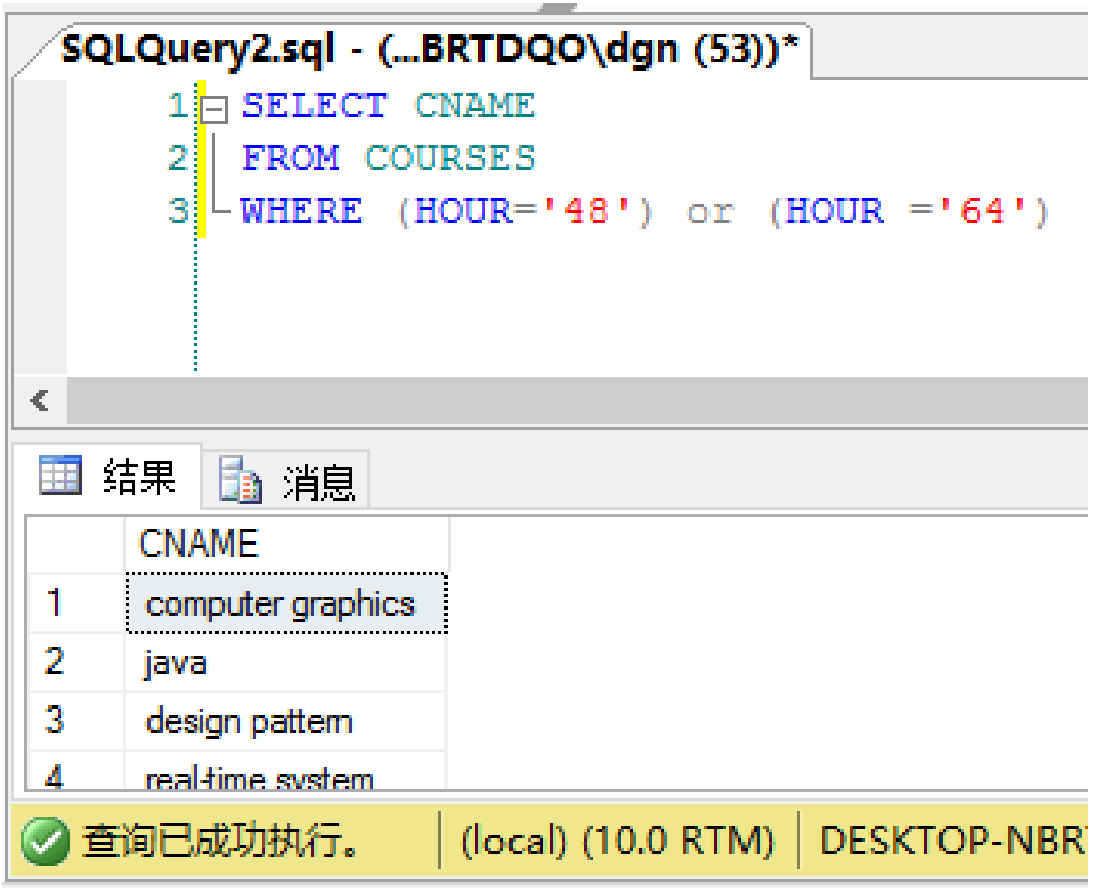
- 相当于给‘积点’取了个别名。
- 原本表中无定义“积点”这个属性，但可以通过score属性计算而得。

- # 实验示例

3. 查询课时是48或者64的课程名称

（注：可使用 in 或者 or 语句）





- # 实验示例

4. 查询所有课程名称中**含有data**的课程编号

（注：使用模糊查询like）

SQLQuery2.sql - (...BRTDQO\dgn (53))*

```
1  SELECT CNAME
2  FROM COURSES
3  WHERE CNAME like '%data%'
```

结果 消息

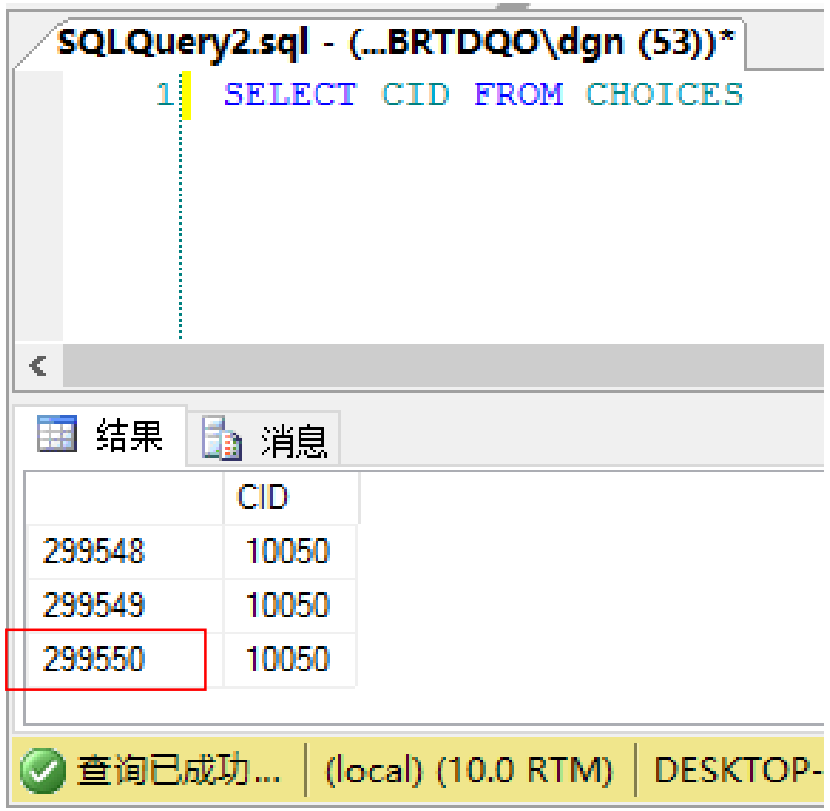
	CNAME
1	database
2	data structure
3	data mining
4	data warehouse

查询已成功执行。 (local) (10.0 RTM) DESKTOP-NBRTDQO\dgn (53) School 00:00:00 4 行

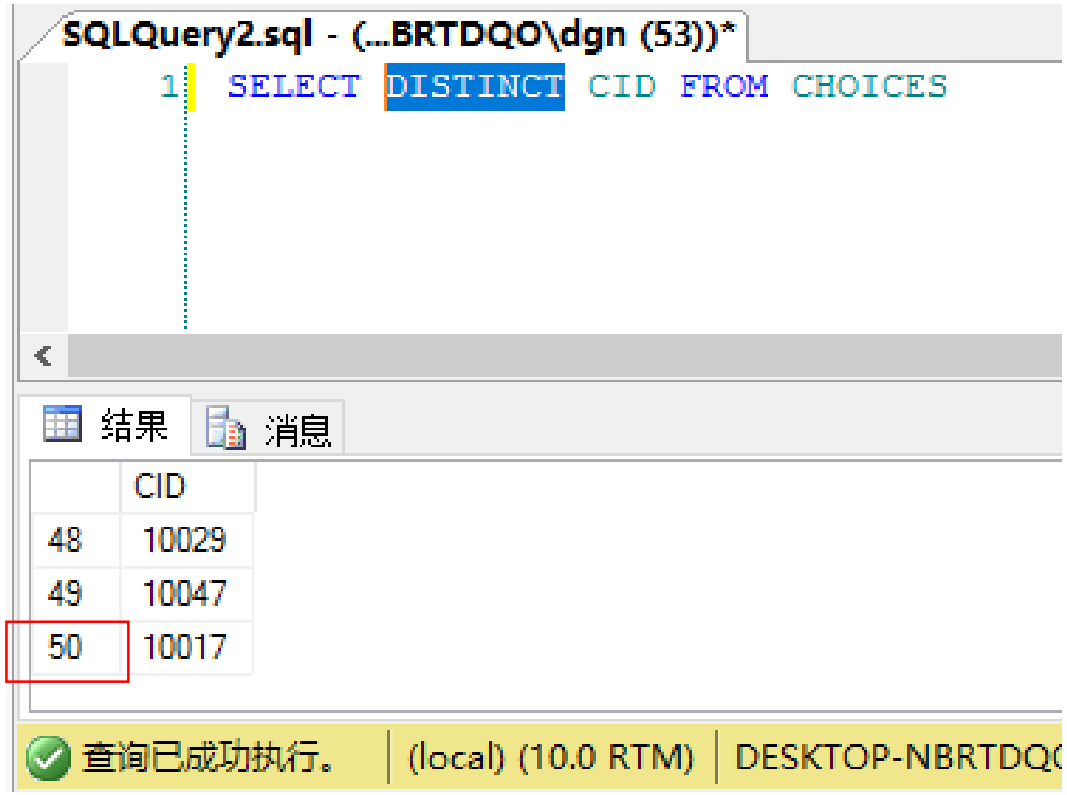
- # 实验示例

5. 查询所有选课表中的课程号（不重复显示）

（注：使用distinct去重）



去重前记录数

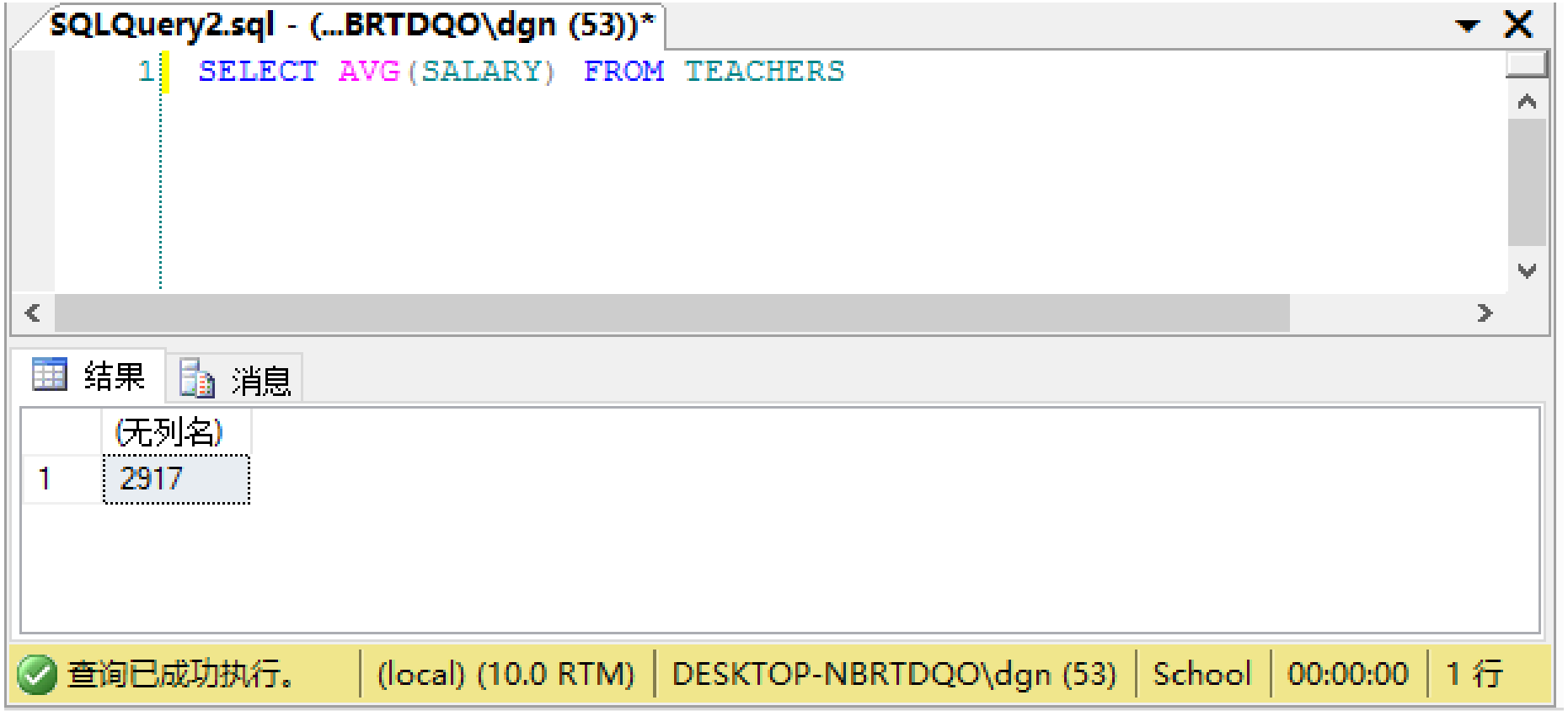


去重后记录数

- # 实验示例

6.统计所有老师的平均工资

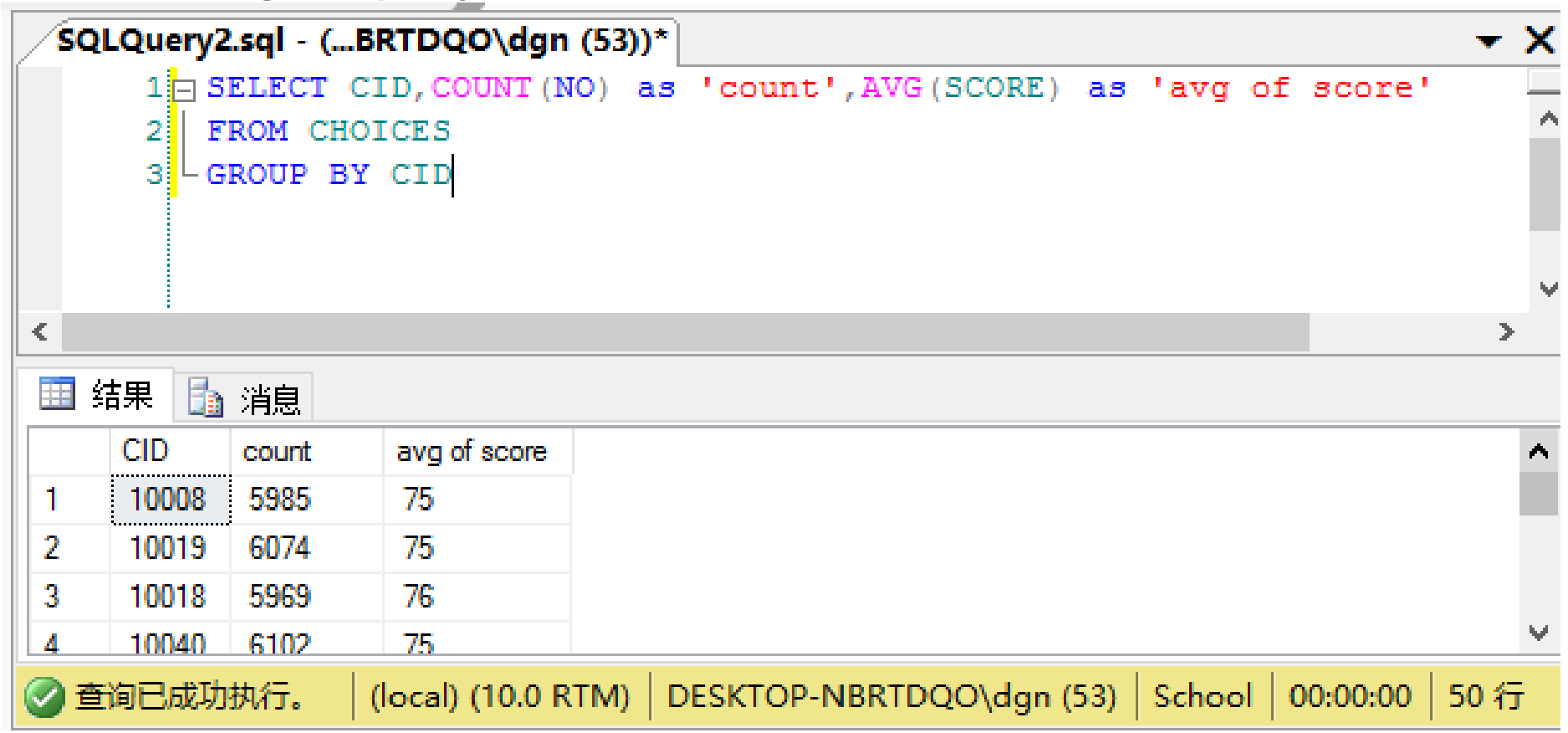
（注：用AVG( )函数求平均数）



- 实验示例

7.统计每个课程的选课人数和平均成绩

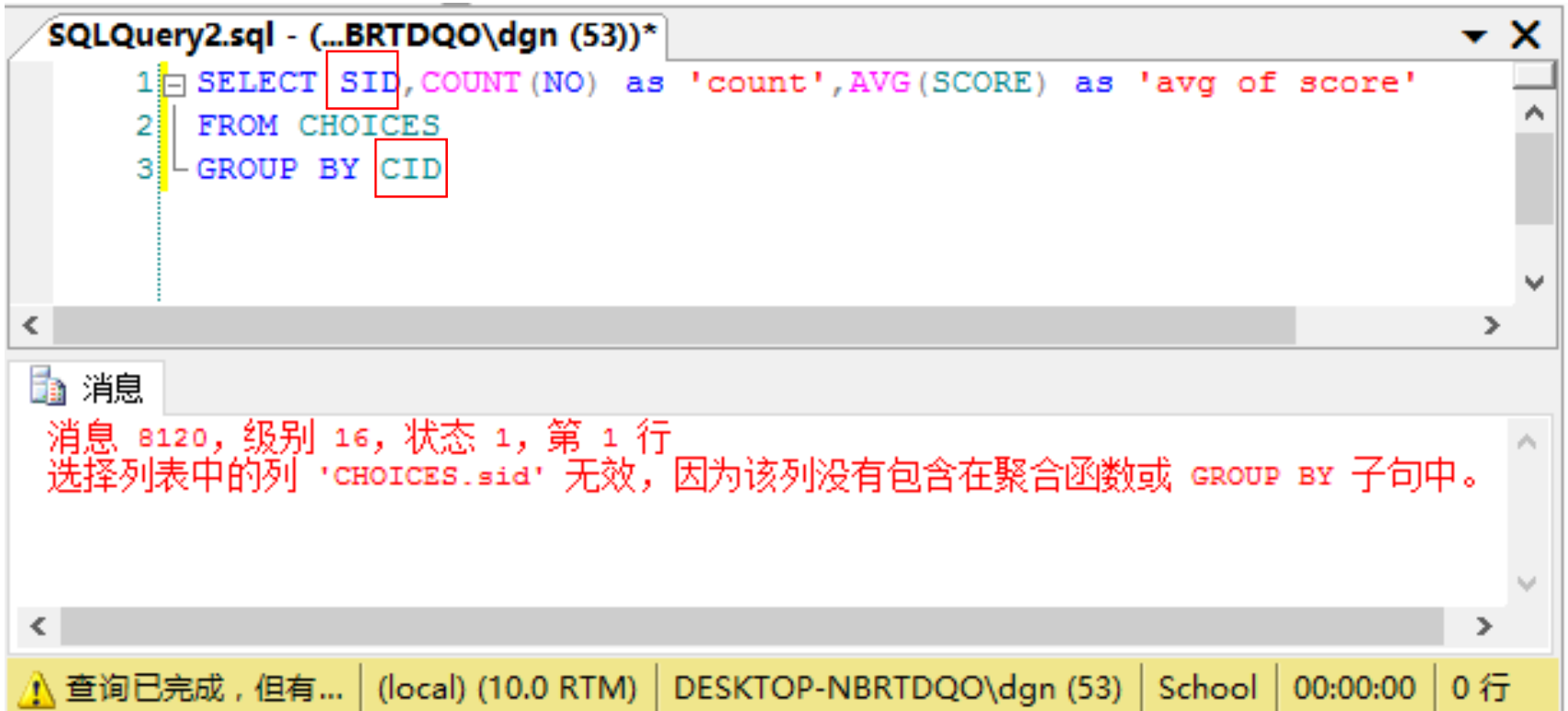
（注：使用group by ）



- 实验示例

注意：如果有group by语句，则seclect语句允许**groupby字句中出现的字段**和**集合函数表达式**，**不允许出现包含其他字段**的表达式。

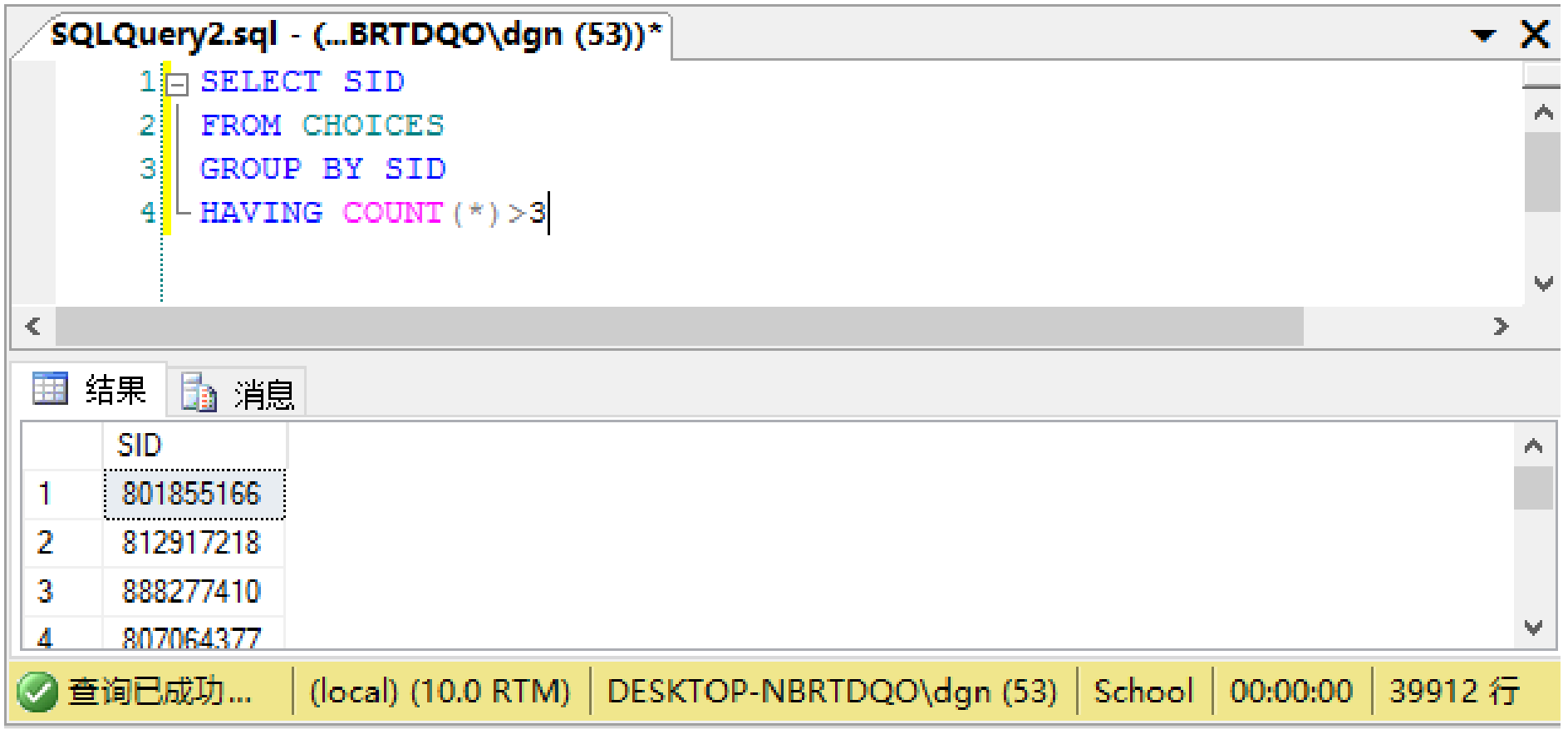
（比如将上题中select语句中 CID 替换成 SID则报错，因为groupby 是CID，select语句中出现SID会报错）



- 实验示例

8. 查询至少选修了三门课程的学生编号

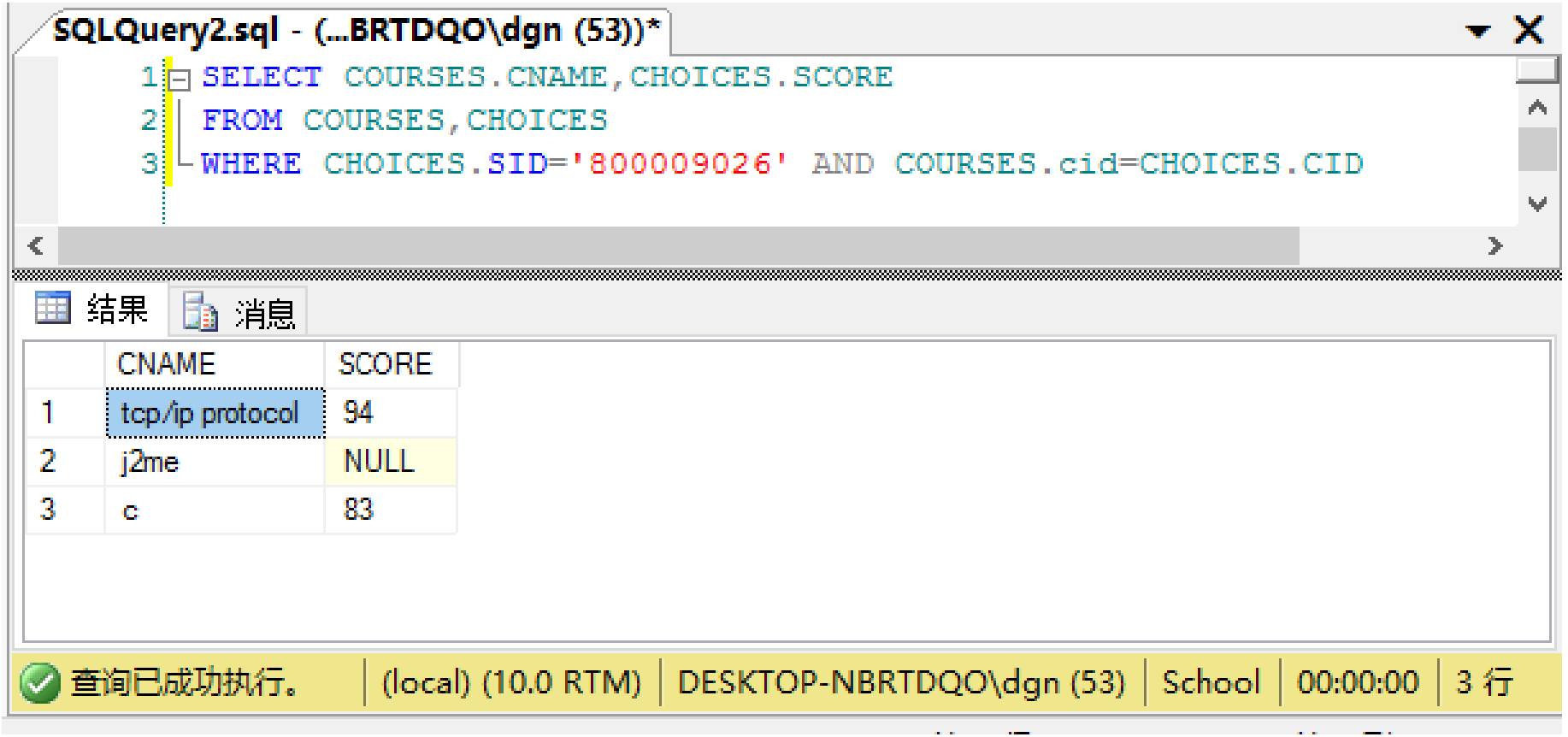
(注：分组后还要求按照一定条件对这些组进行筛选，则使用having语句指定筛选条件)



- 实验示例

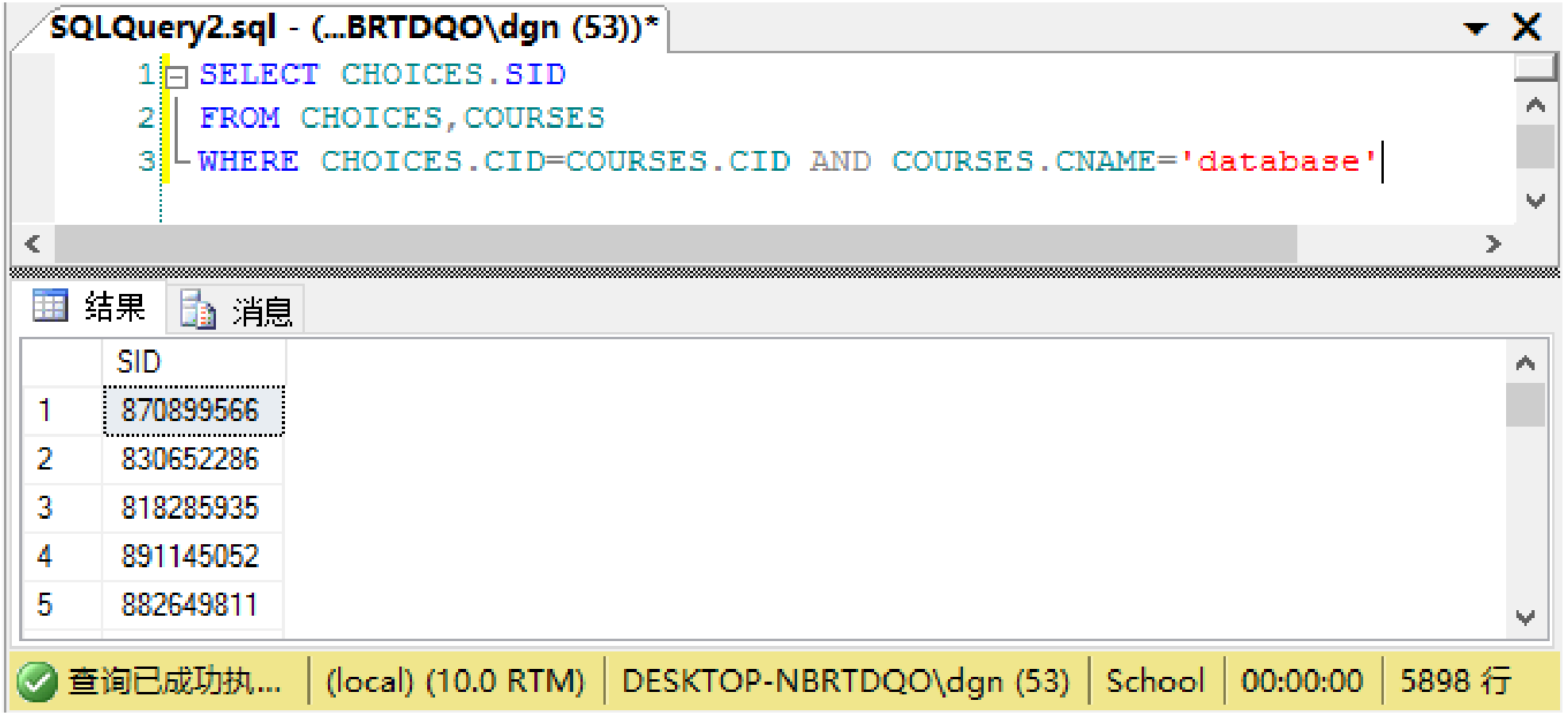
二. 连接查询

1. 查询编号800009026的学生所选的全部课程的课程名和成绩。



- 实验示例

2.查询所有选择了database课程的学生的编号。



- 实验示例

3.查询选择了同一个课程的学生对。
（注：因为要查询同一课程的所有学生对，需要对表CHOICES进行自身连接，需要两个不同的名称来标志同一个表CHOICES，可以通过给表取不同别名实现）

SQLQuery2.sql - (...BRTDQO\dgn (53))* SQLQuery1.sql - (l...dgn (52)) 正在执行...*

```
1  SELECT X.SID, Y.SID
2  FROM CHOICES AS X, CHOICES AS Y
3  WHERE X.CID=Y.CID AND X.SID!=Y.SID
```

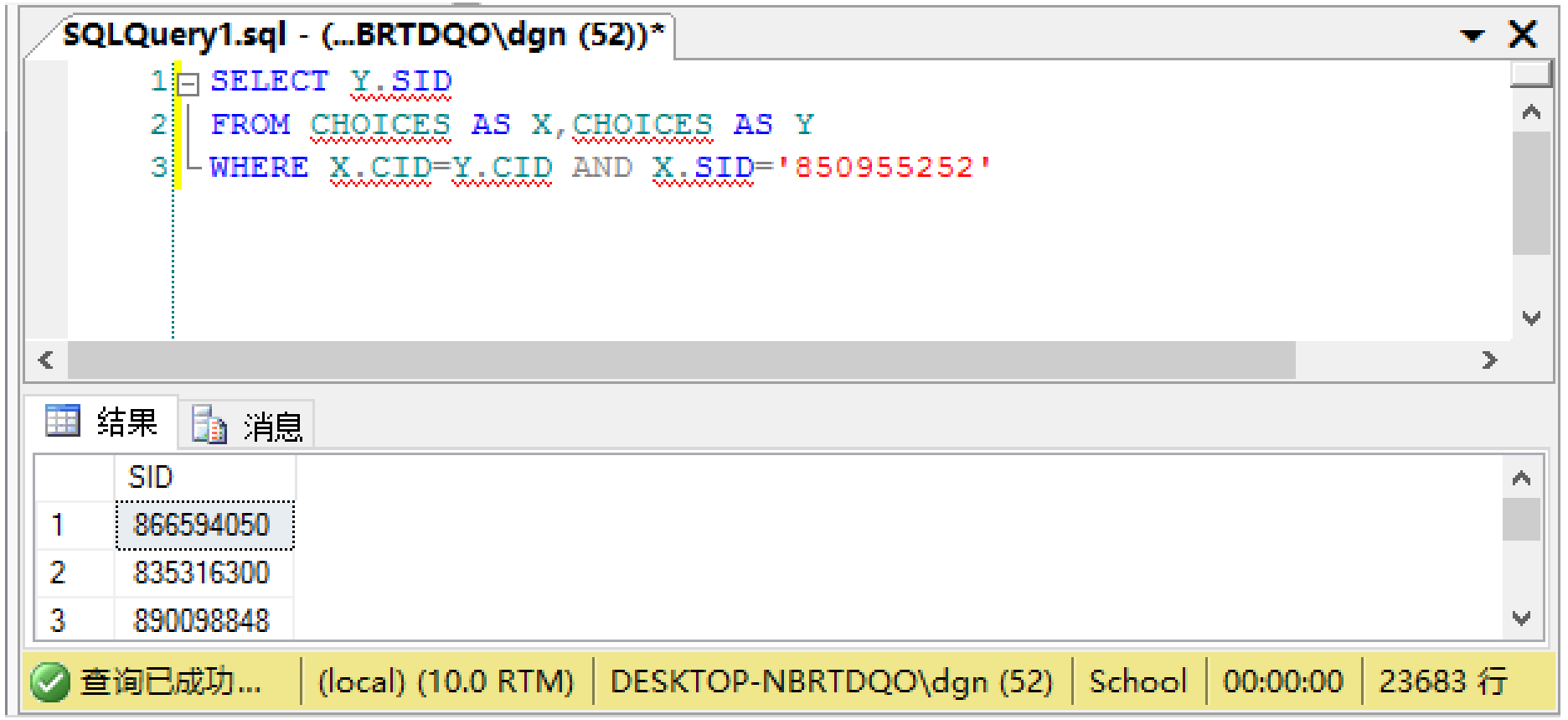
该语句能确保不存在 学生自己与自己组成对

结果 消息

	SID	SID
1	870899566	830652286
2	870899566	818285935
3	870899566	891145052

- 实验示例

4.查询与编号850955252的学生选修至少一门相同课程的学生的编号。



- # 实验示例

5.查询学生的基本信息以及选修课程编号和成绩（外连接）
（注：若某同学没有选课，则只输出其基本情况，其选课信息为空值即可，即用外连接）

SQLQuery1.sql - ...NBRTDQO\dgn (52))

```
1  SELECT STUDENTS.SID,STUDENTS.SNAME,STUDENTS.GRADE,CHOICES.CID,CHOICES.SCORE
2  FROM STUDENTS JOIN CHOICES ON STUDENTS.SID=CHOICES.SID
```

结果 消息

	SID	SNAME	GRADE	CID	SCORE
1	823069829	pxfys	1994	10037	76
2	850955252	baqzmo	2001	10021	54
3	847061074	qxkbh	1994	10025	92

- 实验示例

6.查询编号为850955252学生的姓名和选修课程名称以及成绩。
（注：多表连接）

SQLQuery1.sql - (...BRTDQO\dgn (52))*

```
1  SELECT STUDENTS.SNAME,COURSES.CNAME,CHOICES.SCORE
2  FROM STUDENTS,COURSES,CHOICES
3  WHERE STUDENTS.SID=CHOICES.SID AND COURSES.CID=CHOICES.CID AND STUDENTS.SID='850955252'
```

结果 消息

	SNAME	CNAME	SCORE
1	baqzmo	j2me	54
2	baqzmo	data mining	62
3	baqzmo	embeded system	94
4	baqzmo	project manag...	76

查询已成功执行。 (local) (10.0 RTM) DESKTOP-NBRTDQO\dgn (52) School 00:00:00 4 行

- 练习

(1)查询全部课程的详细记录;

(2)查询所有有选修课的学生的编号;

(3)查询课时<88(小时)的课程的编号;

(4)请找出总分超过400分的学生;

(5)查询课程的总数;

(6)查询所有课程和选修该课程的学生总数;

(7)查询选修成绩超过60的课程超过两门的学生编号;

(8)统计各个学生的选修课程数目和平均成绩;

(9)查询选修Java的所有学生的编号及姓名;

(10)查询姓名为ssht的学生所选的课程的编号和成绩;

(11)查询其他课时比课程C++多的课程的名称;