

# 实验报告

姓名：胡瑞康

学号：22336087

## 数据库表结构

- **students (sid, sname, email, grade)**  
学生表，包含学生的编号、姓名、邮箱和年级。
- **teachers (tid, tname, email, salary)**  
教师表，包含教师的编号、姓名、邮箱和薪水。
- **courses (cid, cname, hour)**  
课程表，包含课程的编号、名称和课时。
- **choices (no, sid, tid, cid, score)**  
选课表，包含选课记录的编号、学生编号、教师编号、课程编号和成绩。

### (1) 查询全部课程的详细记录

```
SELECT *  
FROM courses;
```

查询并显示所有课程的详细信息，包括课程编号、课程名称和课时。

The screenshot shows the SQL Server Enterprise Manager interface. The top pane displays a query in the 'Console' window:

```
SELECT *  
FROM courses;
```

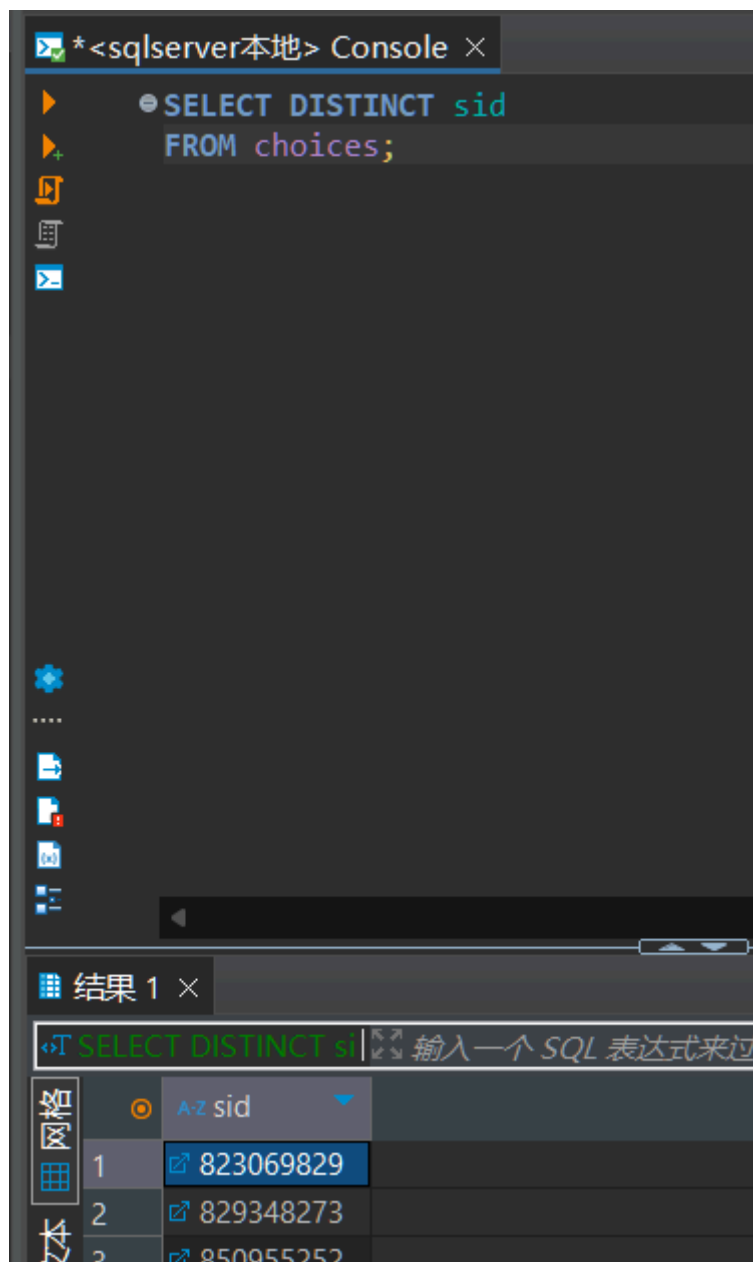
The bottom pane shows the results of the query, titled '结果 1'. The results are displayed in a table with the following columns: 'A-z cid', 'A-z cname', and '123 hour'. The table contains 6 rows of data:

	A-z cid	A-z cname	123 hour
1	10001	database	96
2	10002	operating system	88
3	10003	computer graphics	48
4	10004	java	48
5	10005	c++	60
6	10006	design pattern	48

## (2) 查询所有有选修课的学生的编号

```
SELECT DISTINCT sid  
FROM choices;
```

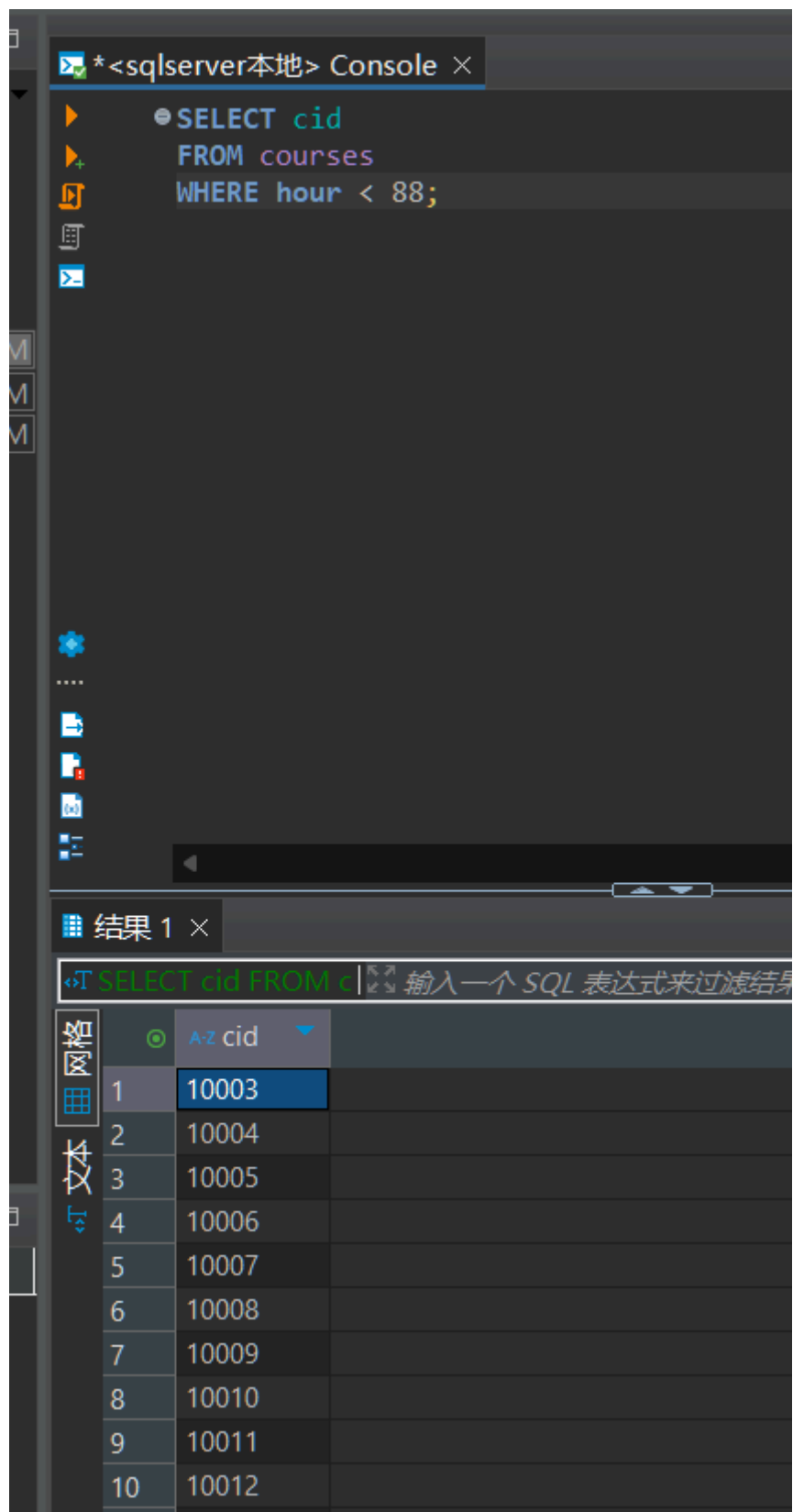
查询并显示所有有选修课程的学生的编号，使用 `DISTINCT` 关键字去除重复的学生编号。



### (3) 查询课时<88(小时)的课程编号

```
SELECT cid
FROM courses
WHERE hour < 88;
```

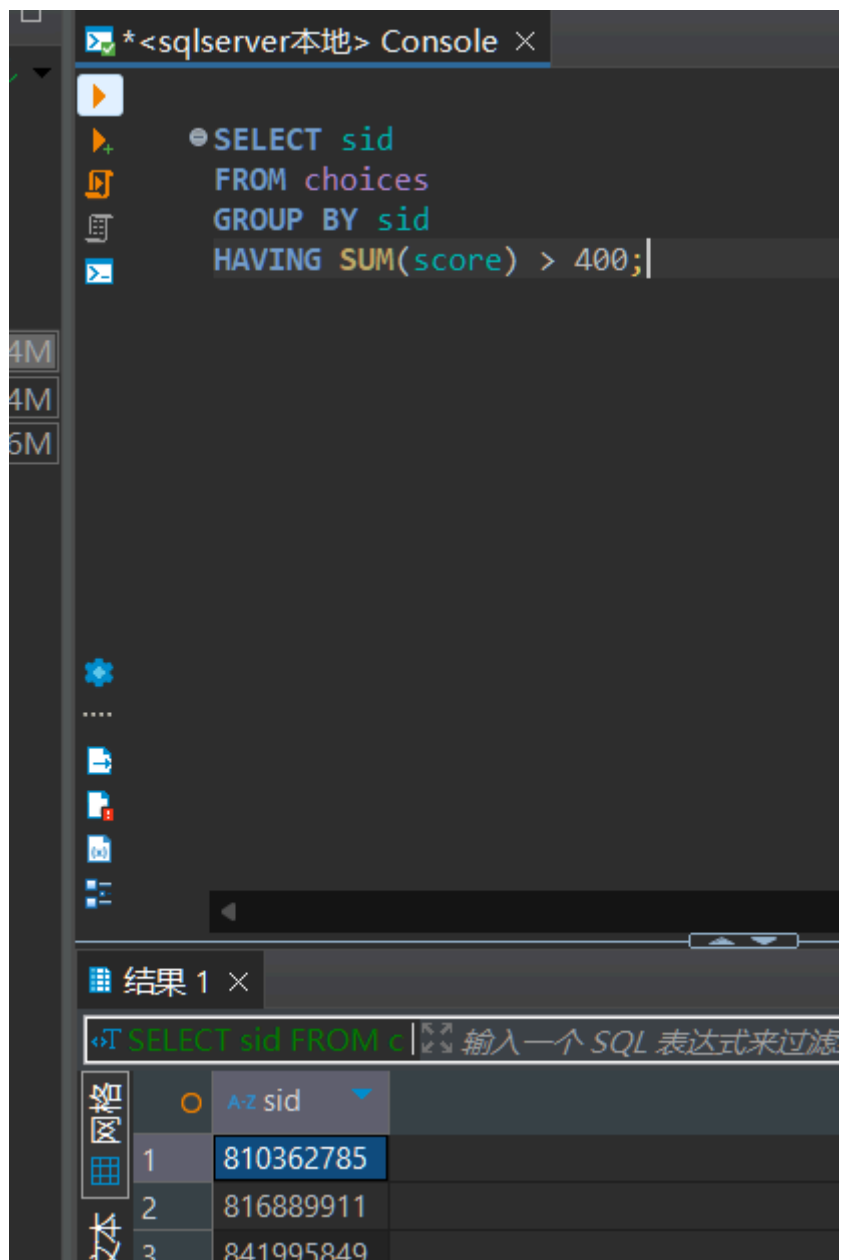
查询并显示课时小于88小时的课程编号。



#### (4) 请找出总分超过400分的学生

```
SELECT sid
FROM choices
GROUP BY sid
HAVING SUM(score) > 400;
```

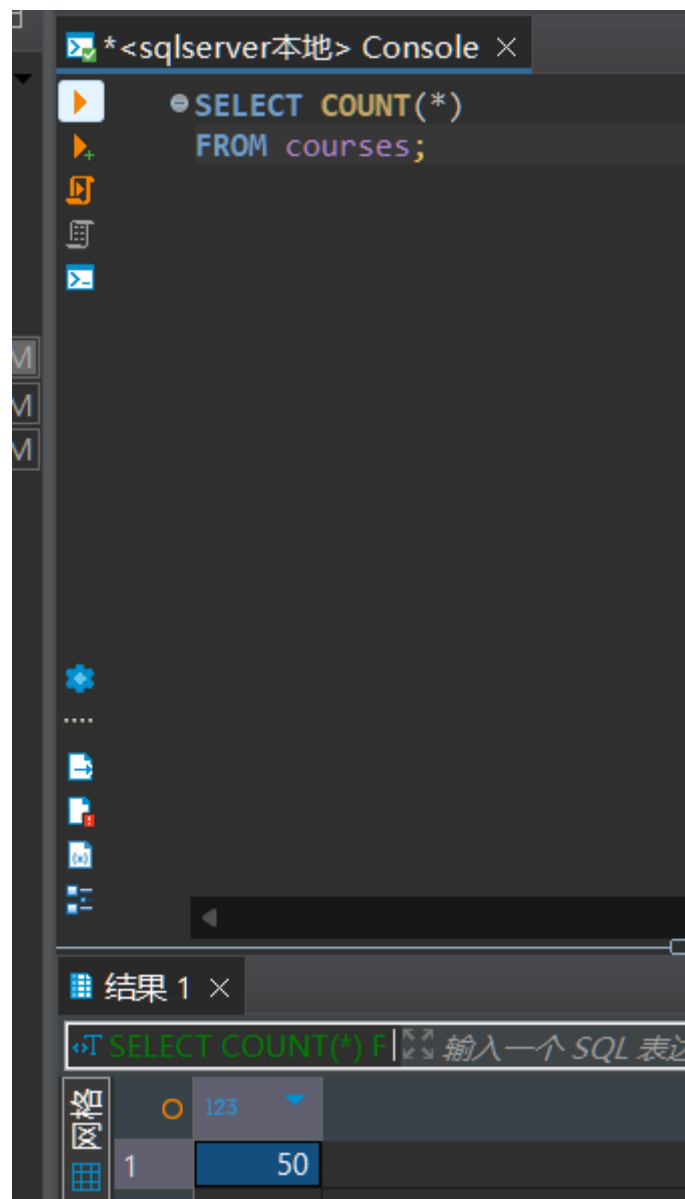
查询并显示总分超过400分的学生编号，通过 `GROUP BY` 和 `HAVING` 子句对学生进行分组并筛选出总分超过400分的学生。



## (5) 查询课程的总数

```
SELECT COUNT(*)
FROM courses;
```

查询并显示课程的总数，使用 `COUNT(*)` 函数统计课程表中的记录数。



## (6) 查询所有课程和选修该课程的学生总数

```
SELECT courses.cid, courses.cname, COUNT(choices.sid) AS snum
FROM courses
LEFT JOIN choices ON courses.cid = choices.cid
GROUP BY courses.cid, courses.cname;
```

查询并显示所有课程的详细信息（包括课程编号、课程名称和课时）以及选修该课程的学生总数，并将其命名为 `snum`。通过 `LEFT JOIN` 操作连接 `courses` 和 `choices` 表，确保即使某门课程没有学生选修也能显示课程信息。使用 `GROUP BY` 对课程进行分组，并使用 `COUNT` 函数统计每组的学生数。

Console x COURSES STUDENTS CHOICES

```

SELECT courses.cid, courses.cname, COUNT(choices.sid) AS snum
FROM courses
LEFT JOIN choices ON courses.cid = choices.cid
GROUP BY courses.cid, courses.cname;

```

结果 1 x

SQL SELECT courses.cid, courses.cname, COUNT(choices.sid) 输入一个 SQL 表达式来过滤结果 (使用 Ctrl+F)

	A-Z cid	A-Z cname	123 snum
1	10001	database	5,898
2	10002	operating system	6,013
3	10003	computer graphics	5,975
4	10004	java	6,110
5	10005	c++	6,031

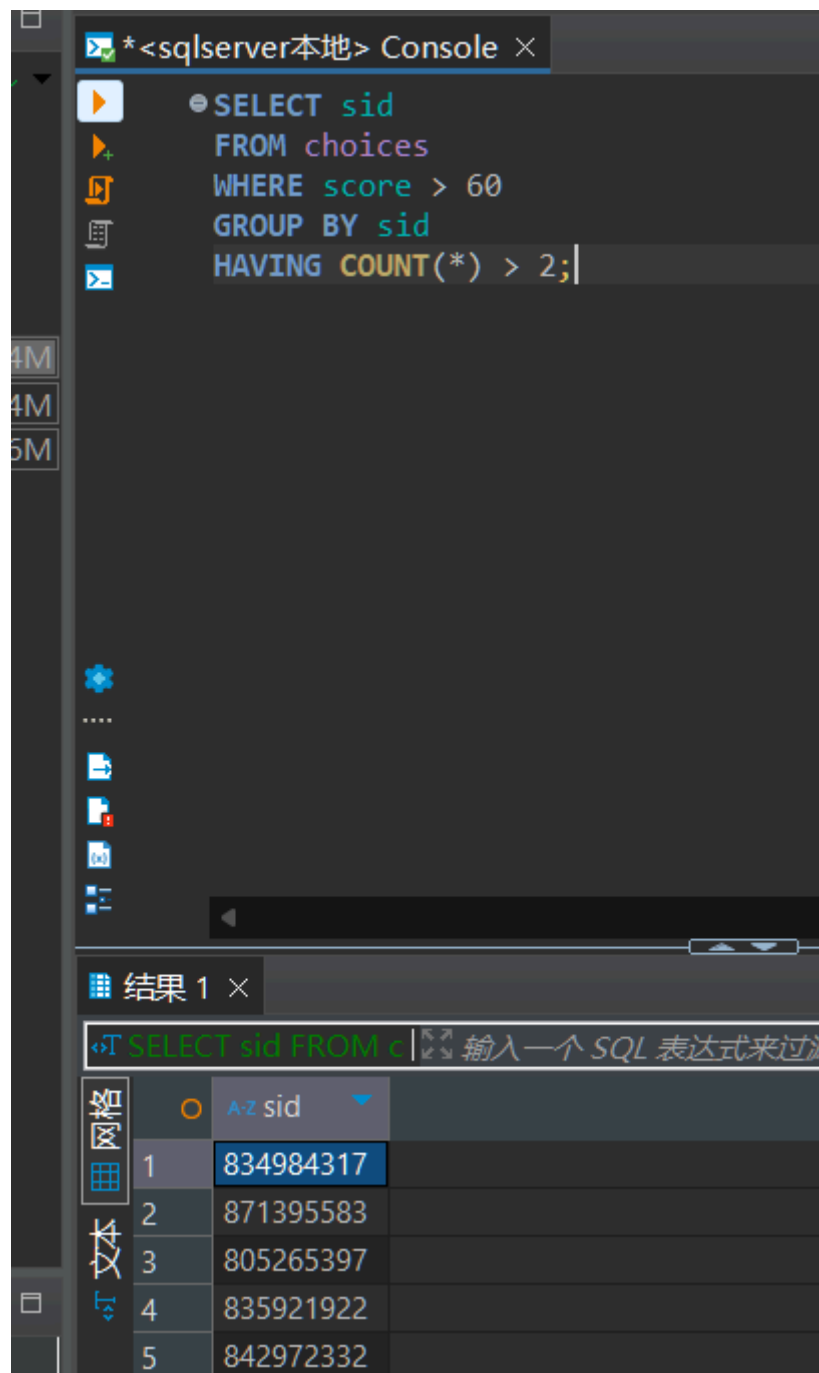
## (7) 查询选修成绩超过60的课程超过两门的学生编号

```

SELECT sid
FROM choices
WHERE score > 60
GROUP BY sid
HAVING COUNT(*) > 2;

```

查询并显示选修成绩超过60分的课程超过两门的学生编号，通过 `WHERE` 子句筛选出成绩超过60分的记录，再通过 `GROUP BY` 和 `HAVING` 子句筛选出选修课程数超过两门的学生。

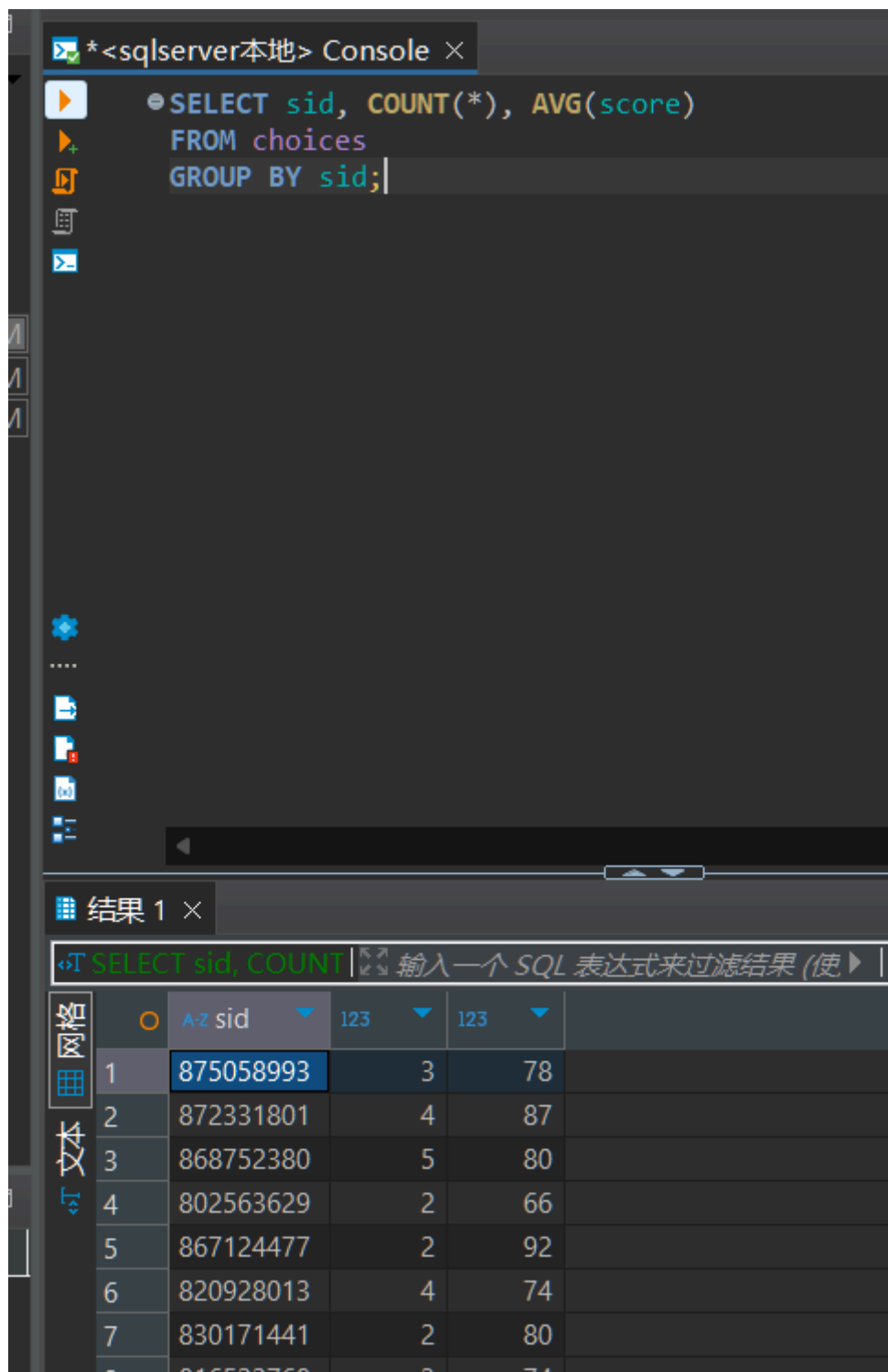


## (8) 统计各个学生的选修课程数目和平均成绩

```
SELECT sid, COUNT(*), AVG(score)
FROM choices
GROUP BY sid;
```

查询并显示每个学生的选修课程数目和平均成绩，通过 `GROUP BY` 对学生进行分组，并使用 `COUNT(*)` 和 `AVG(score)` 函数分别统计选课数和平均成绩。





### (9) 查询选修Java的所有学生的编号及姓名

```
SELECT students.sid, students.sname  
FROM students, choices, courses  
WHERE students.sid = choices.sid AND choices.cid = courses.cid AND courses.cname  
= 'Java';
```

查询并显示选修Java课程的所有学生的编号和姓名，通过多表连接 (students, choices, courses) 并使用 WHERE 子句筛选出选修Java课程的学生。

The screenshot shows a SQL query in the 'Console' window of SQL Server Enterprise Manager. The query is:

```
SELECT students.sid, students.sname
FROM students, choices, courses
WHERE students.sid = choices.sid AND choices.cid = courses.cid AND courses.cname = 'Java';
```

The results are displayed in a table with 7 rows and 2 columns: 'sid' and 'sname'.

	sid	sname
1	833961570	bzbogqep
2	856259316	qnsmnk
3	840205068	vtzpxh
4	870755799	gjikyychh
5	811353704	wcgatkwjn
6	881617726	ikmsab
7	806836750	rrveair

## (10) 查询姓名为sssh的学生所选的课程的编号和成绩

```
SELECT choices.cid, choices.score
FROM students, choices
WHERE students.sid = choices.sid AND students.sname = 'sssh';
```

查询并显示姓名为sssh的学生所选课程的编号和成绩，通过多表连接 (students, choices) 并使用 WHERE 子句筛选出指定学生的选课记录。

The screenshot shows a SQL query in the 'Console' window of SQL Server Enterprise Manager. The query is:

```
SELECT choices.cid, choices.score
FROM students, choices
WHERE students.sid = choices.sid AND students.sname = 'sssh';
```

The results are displayed in a table with 5 rows and 2 columns: 'cid' and 'score'.

	cid	score
1	10004	76
2	10037	84
3	10037	54
4	10030	53
5	10019	79

## (11) 查询其他课时比课程C++多的课程的名称

```
SELECT cname
FROM courses
WHERE hour > (SELECT hour
               FROM courses
               WHERE cname = 'C++');
```

查询并显示课时比C++课程多的其他课程的名称，通过子查询获取C++课程的课时，并在主查询中筛选出课时大于该值的课程。

\*<sqlserver本地> Console × COURSES STUDENTS

```
SELECT cname
FROM courses
WHERE hour > (SELECT hour
               FROM courses
               WHERE cname = 'C++');
```

结果 1 ×

SELECT cname FROM courses WHERE hour > (SELECT h

	A-Z cname	
1	database	
2	operating system	
3	tcp/ip protocol	
4	algorithm	
5	compiling principle	