

# 中山大学数据科学与计算机学院本科生实验报告

## (2020 学年秋季学期)

课程名称：数据库系统

任课教师：刘玉葆

助教：

年级&班级	18 级 计科一班	专业(方向)	计科 (人工智能与大数据方向)
学号	18340014	姓名	陈嘉宁
电话	18475934419	Email	734311072@qq.com
开始日期	2020/9/18	完成日期	2019/9/21

### 一、 实验题目

SQL 语句应用 (单表查询和连接查询)

### 二、 实验目的

1. 熟悉 SQL 的数据查询语言;
2. 能够使用 SQL 语句对数据库进行单表查询、连接查询。

### 三、 实验内容

1. 单表查询：
  - 1) 查询的目标表达式为所有列、指定列或指定列的运算;
  - 2) 用 DISTINCT 保留字消除重复行;
  - 3) 对查询结果排序和分组;
  - 4) 集合分组使用集函数进行各项统计。
2. 连接查询：
  - 1) 笛卡儿连接和等值连接;
  - 2) 自连接;
  - 3) 外连接;
  - 4) 复合条件连接;
  - 5) 多表连接。

## 四、实验过程及结果

### 1. 查询全部课程的详细记录:

查询语句:

```
select cid,cname,hour  
from COURSES
```

查询结果 (项数过多, 仅截取部分):

	cid	cname	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
7	10007	uml	30
8	10008	data structure	60
9	10009	cryptology	36
10	10010	software engineering	50
11	10011	distributed computing	36
12	10012	erp	40
13	10013	artificial intelligence	46
14	10014	computer network	60

### 2. 查询所有有选修课的学生的编号:

查询语句:

```
select sid  
from choices
```

查询结果 (项数过多, 仅截取部分):

	cid
1	10003
2	10004
3	10005
4	10006
5	10007
6	10008
7	10009
8	10010
9	10011
10	10012
11	10013
12	10014
13	10015
14	10016

### 3. 查询课时<88(小时)的课程的编号:

查询语句:

```
select cid  
from COURSES  
where hour<88
```

查询结果 (共有 48 项, 此处仅截取部分):

	cid
1	10003
2	10004
3	10005
4	10006
5	10007
6	10008
7	10009
8	10010
9	10011
10	10012
11	10013
12	10014
13	10015
14	10016

4. 请找出总分超过 400 分的学生:

查询语句:

```
select sid, SUM(score) as s
from CHOICES
group by sid
having SUM(score)>400
```

查询结果 (仅截取部分):

	sid	s
1	864460136	447
2	822101172	407
3	806671300	415
4	828217115	428
5	826220006	409
6	856862351	434
7	875312479	430
8	874259807	409
9	811119994	429
10	894478231	449
11	895997349	458
12	870772470	432
13	845388732	404
14	891230984	407

5. 查询课程的总数

查询语句:

```
select count(*) as number
from COURSES
```

查询结果:

	number
1	50

6. 查询所有课程和选修该课程的学生总数:

查询语句:

```
select cid, COUNT(*) as summary
from CHOICES
group by cid
order by cid
```

查询结果 (共 50 项, 仅截取部分):

	cid	summary
1	10001	5898
2	10002	6013
3	10003	5975
4	10004	6110
5	10005	6031
6	10006	6090
7	10007	5965
8	10008	5985
9	10009	5965
10	10010	6027
11	10011	6086
12	10012	5819
13	10013	5982
14	10014	5866

7. 查询选修成绩超过 60 的课程超过两门的学生编号:

查询语句:

```
select sid, COUNT(sid) courses
from CHOICES
where score > 60
group by sid
having COUNT(sid) > 2
```

查询结果 (仅截取部分):

	sid	courses
1	828667933	3
2	886690009	5
3	801855166	4
4	812917218	4
5	888277410	5
6	807064377	4
7	891770368	3
8	873055545	4
9	863857955	3
10	846080968	3
11	808999533	3
12	830850869	4
13	832380316	3
14	818423896	4

8. 统计各个学生的选修课程数目和平均成绩:

查询语句:

```
select sid, AVG(score) as avarage, COUNT(cid) as courses
from CHOICES
group by sid
```

查询结果:

	sid	avarage	courses
1	834984317	73	4
2	871395583	81	3
3	865375956	73	2
4	856542501	69	2
5	816362472	64	2
6	861315623	NULL	2
7	805265397	73	5
8	835921922	71	4
9	856698430	93	3
10	869826277	81	3
11	885936900	80	1
12	842972332	72	4
13	836949491	90	3
14	819154113	61	5

9. 查询选修Java的所有学生的编号及姓名:

在进行此项操作时, 我发现 SQL server 并不支持自然连接 (natural join) 的操作, 只好使用内连接 (inner join) 并通过指定 id 相同来完成自然连接的功能。

查询语句:

```
select s.sid, sname
from (STUDENTS as s inner join CHOICES as c on s.sid=c.sid) inner join COURSES as co on c.cid=co.cid
where cname='Java'
```

查询结果 (仅截取部分):

	sid	sname
1	808994056	bhonk
2	809122318	afstpkgar
3	809152852	yeuvs
4	809238137	kbbfenf
5	809374716	pcdfypql
6	809459051	oawyyj
7	809481258	vzqymzyz
8	809507502	mpxul
9	809536879	xetlezag
10	809536879	xetlezag
11	809555735	kprgtwzcx
12	809582865	htjinbaj
13	809794680	wsrcoryyd
14	809866907	htyatcvl

10. 查询姓名为 sssht 的学生所选的课程的编号和成绩：

同上题，使用 inner join 来实现 natural join 的功能。

查询语句：

```
select cid, score
from STUDENTS as s inner join CHOICES as c on s.sid=c.sid
where sname='sssht'
```

查询结果：

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

11. 查询其他课时比课程 C++ 多的课程的名称：

通过所有课时比课程 C++ 多的课程减去 C++ 课程来得到题目要求的元组。

查询语句：

```
select x.cname
from COURSES as x, COURSES as y
where x.hour>y.hour and y.cname = 'c++'
except
select x.cname
from COURSES as x, COURSES as y
where x.hour>y.hour and y.cname = 'c++' and x.cname = 'c++'
```

查询结果：

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

## 五、 实验感想

本次实验相较于上次有了一些难度上的增长，逻辑更加复杂且更加考究细节。另外，在实验过程中遇到一个较大的问题——SQL server 并不支持 natural join 的操作，因此只能通过 inner join 来实现 natural join 的功能。