

《数据库系统原理》

课内实验报告

易凯

2017 年 4 月 16 日

班 级 软件 53 班

学 号 2151601053

邮 箱 williamyi96@gmail.com

联系电话 13772103675

个人网站 <https://williamyi.tech>

提交日期 2017 年 5 月 12 日

| | |
|-----|---|
| 目 录 | 2 |
|-----|---|

目录

| | |
|---|----|
| 1 实验报告声明 | 4 |
| 2 实验目的 | 4 |
| 3 实验内容 | 4 |
| 3.1 建表 | 4 |
| 3.2 查询 | 5 |
| 4 收获总结 | 6 |
| 5 致谢 | 6 |
| 6 参考文献 | 7 |
| 7 附录-实验结果 | 7 |
| 7.1 Appendix I: Createtab.sql and Insert.sql | 7 |
| 7.2 Appendix II: Created DB and Created Table | 11 |
| 7.3 Appendix III: Query Results | 13 |

插图

| | | |
|---|-------------------------------------|----|
| 1 | A.use database of student | 11 |
| 2 | B. table s | 11 |
| 3 | C. table c | 12 |
| 4 | D. table sc | 12 |
| 5 | A. 查询 1 和 2 | 13 |
| 6 | B. 查询 3 和 4 | 13 |
| 7 | C. 查询 5 和 6 | 14 |
| 8 | D. 查询 7 和 8 | 14 |

1 实验报告声明

此报告为西安交通大学本科生易凯在《数据库基本原理》的基础之上，加上自己的理解以及个人感悟完成的数据库综合实验报告。未经允许，禁止挪作他用。由于笔者能力有限，报告中不乏有许多缺漏指出，因此望读者谅解的同时对于其中的不足指出能够批评指正。

2 实验目的

《数据库系统原理》课内实验的实验目的主要有如下的几个方面：

1. 熟悉 SQL 基本建库建表语句
2. 掌握 SQL 语句查询基本方法
3. 建立数据库设计的基本意识
4. 熟练运用 MySQL 进行 SQL 语句实现
5. 培养分析问题解决问题的能力

3 实验内容

3.1 建表

该部分主要是利用数据库建表等语言进行的数据库建立以及数据填充，在本次试验中，建立了名为 student 的数据库，并且创建了三张不同的表。

数据库的使用参见附录 II.A 所示所示。

接下来图 2 是 s 表的基本内容，其中 s 表中的属性有 SNO 表示学生学号，SNAME 表示学生的名字，SEX 表示学生的性别，BDATE 表示出生日期，HEIGHT 表示身高，DORM 表示宿舍地址。SNO 为主键，设置为非空，而且由于其是定长的，因而使用 char 来声明。其他较为明显，具体内容参见附录 II.B 所示。

接下来图 3 呈现的是 c 表的基本内容，也就是课程表的基本内容。其中 c 表中的属性有 CNO 表示课程号，LHOUR 表示课时数，CREDIT 表

示学分, SEMESTER 表示季度。CNO 为主键, 设置为非空, 由于其是定长的, 因此使用 char 来声明。其他较为明显, 具体内容参见附录 II.C 所示。

最后呈现的图 4 是 sc 表的内容, 也即是选课表的基本内容。其中 sc 表中的属性有 SNO 表示学生号, CNO 表示课程号, GRADE 表示课程号。其中 SNO 与 CNO 均为外键, 分别引用自 S 表和 C 表。具体声明以及数据存储情况如附录 II.D 所示。

由上面的内容可以直观的看到, 数据库的三张表已经被较好地建立起来。

3.2 查询

query one 查询选修课程“CS102”的学生学号、成绩, 结果参见附录 III.A。

query two 查询选修课程“EE101”的女同学姓名, 结果参见附录 III.A。

query three 查询不选修课程“CS102”的学生姓名, 结果参见附录 III.B。

query four 查询所有比“李涛”年龄大的学生姓名、性别、年龄, 结果参见附录 III.B。

query five 查询选修课程“CS101”的学生中成绩最高的学生的学号, 结果参见附录 III.C。

query six 查询学生姓名及其所选修课程的课程号、学分和成绩, 结果参见附录 III.C。

query seven 查询平均成绩超过 80 分的学生姓名和平均成绩, 结果参见附录 III.D。

query eight 查询选修三门以上课程 (包括三门) 的学生总成绩 (不统计不及格的课程), 并要求按总成绩的降序排列出来, 结果参见附录 III.D。

4 收获总结

《数据库系统原理》的学习是一段十分令人难忘的历程，而今，随着数据库综合实验实验报告的落笔，至此，本科阶段进行数据库系统的系统性学习的场合就此远去了，这一路走来，感慨良多。

课程内容本身而言，在学习的过程中，《数据库系统原理》中有很多相关的概念相对而言不是那么直观，容易理解，同时知识点相对而言较多，虽然成系统，但是实际上真正把握还有一定的难度。因此，整个数据库系统原理的学习过程中，侯老师在课堂上发挥了很强的名师风采，但是于己而言有些知识点没有很好的消化吸收。因此，在私底下花了较多的时间去完成作业，去把握理解核心的概念与知识点。后来渐渐发现，只要沉下心来，有些东西也不是那么难以理解。渐渐地也掌握了学习的相关方法。

这次的数据库系统原理的综合实验实际上就是考察的 sql 语句的基本用法，熟悉数据库设计实现的基本流程，为之后的学习打下坚实的基础。总体而言，在完成实验报告的过程中，出于实验题目基础性的限制，因此没有遇到很多的障碍，但是一路走来对于数据库的相关概念有了一个更深的理解。同时也对数据库建库、建表以及增删改查等的相关操作更加熟练了，此外还对于这些内容有了一个更深的把握。因此，完成实验报告的这一段路同样也是受益匪浅。

作为一名软件工程的本科生，同时数据库又作为软件项目的一个极为重要的环节，因此之后会多次进行交互。希望带着这一段过程中学习到的思路和方法，能够在未来的学习生活中，发挥作用。为未来数据库的学习与实践打下坚实的基础。

5 致谢

Give my sincere thanks to my database teacher, Hou, for his excellent teaching skills and serious altitude for our homework. Give my sincere thanks to some students who have helped me and who have inspired me when discussing with them. Give my sincere thanks to myself because I've overcome all difficulties and have successfully finished my database comprehensive training paper. Give my sincere thanks to those pioneers who have devoted themselves to writing immortal books. Give my sincere thanks to all the people sharing their ideas and harvests without pay in QA commu-

nities.

衷心感谢《数据库系统原理》侯老师出色的教学风范以及严谨的治学态度，衷心感谢那些在讨论中帮助和启发我的同学们，衷心感谢克服了种种困难最终完成了数据库综合训练论文的自己，衷心感谢那些写了不朽著作作为后人指明道路的先驱们，衷心感谢那些在问答社区无私奉献自己智慧成果的所有同仁！

6 参考文献

[1] 沈钧毅，侯迪，冯中慧，何亮. 数据库系统原理 [M]. 西安：西安交通大学出版社. 2014-2.

[2] 侯迪. 数据库系统原理本科生教学课件. 西安：西安交通大学. 2017年春.

[3] Abraham Silberschatz, Henry F.Korth, S.Sudarshan. Database System Concepts[M] Mc-Graw Hill. Connect Learn Succeed. 2009-3.

[4] 肖海蓉. 数据库原理与应用 [M] 北京：清华大学出版社. 2016-1.

[5] Thomas M.Connolly, Carolyn E.Begg. DATABASE SYSTEMS – A Practical Approach to Design, Implementation, and Management.[M]. six edition. Pearson. 2016-6.

7 附录—实验结果

7.1 Appendix I: Createtab.sql and Insert.sql

```
1  — MySQL dump 10.13  Distrib 5.5.27, for Win64 (x86)
2  —
3  — Host: localhost    Database: student
4  —
5  — Server version      5.5.27
6
7  /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
8  /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
9  /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
10 /*!40101 SET NAMES utf8 */;
11 /*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
```

```

12  /*!40103 SET TIME_ZONE='+00:00' */;
13  /*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
14  /*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
15  /*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
16  /*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
17
18  —
19  — Table structure for table 'c'
20  —
21
22  DROP TABLE IF EXISTS 'c';
23  /*!40101 SET @saved_cs_client      = @@character_set_client */;
24  /*!40101 SET character_set_client = utf8 */;
25  CREATE TABLE 'c' (
26    'CNO' char(5) NOT NULL,
27    'LHOUR' double(5,0) DEFAULT NULL,
28    'CREDIT' char(1) DEFAULT NULL,
29    'SEMESTER' varchar(4) DEFAULT NULL,
30    PRIMARY KEY ('CNO')
31  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
32  /*!40101 SET character_set_client = @saved_cs_client */;
33
34  —
35  — Dumping data for table 'c'
36  —
37
38  LOCK TABLES 'c' WRITE;
39  /*!40000 ALTER TABLE 'c' DISABLE KEYS */;
40  INSERT INTO 'c' VALUES ('CS101',60,'3',''), ('CS102',80,'4',''),
41  ('CS103',40,'2','Çi'), ('EE101',60,'3',''), ('EE102',100,'5','Çi');
42  /*!40000 ALTER TABLE 'c' ENABLE KEYS */;
43  UNLOCK TABLES;
44
45  —
46  — Table structure for table 's'
47  —
48
49  DROP TABLE IF EXISTS 's';
50  /*!40101 SET @saved_cs_client      = @@character_set_client */;

```



```

51  /*!40101 SET character_set_client = utf8 */;
52  CREATE TABLE 's' (
53      'SNO' char(8) NOT NULL,
54      'SNAME' varchar(20) DEFAULT NULL,
55      'SEX' varchar(4) DEFAULT NULL,
56      'BDATE' varchar(10) NOT NULL,
57      'HEIGHT' double(3,2) DEFAULT NULL,
58      'DORM' varchar(20) DEFAULT NULL,
59      PRIMARY KEY ('SNO')
60  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
61  /*!40101 SET character_set_client = @saved_cs_client */;
62
63  —
64  — Dumping data for table 's'
65  —
66
67  LOCK TABLES 's' WRITE;
68  /*!40000 ALTER TABLE 's' DISABLE KEYS */;
69  INSERT INTO 's' VALUES ( '01032001' , 'ÖÄpÄ • ' , 'Ä@' , '1983-11-17' , 1.58 , 'Î÷1Éá312' ) ,
70  ( '01032005' , 'Äö¾²' , 'Ä@' , '1982-1-10' , 1.63 , 'Î÷1Éá312' ) ,
71  ( '01032010' , 'ÄîÎ' , 'ÄÐ' , '1982-4-5' , 1.72 , 'Î÷14Éá221' ) ,
72  ( '01032023' , 'ËîÎÄ' , 'ÄÐ' , '1983-6-10' , 1.80 , 'Î÷14Éá221' ) ,
73  ( '01032112' , '¶ÎÄ • æ' , 'ÄÐ' , '1982-2-20' , 1.71 , 'Î÷14Éá221' ) ,
74  ( '03031009' , 'ÎæÄæÄ' , 'Ä@' , '1982-8-11' , 1.60 , 'Î÷2Éá104' ) ,
75  ( '03031011' , 'ÍõÜ»' , 'Ä@' , '1983-12-20' , 1.66 , 'Î÷2Éá104' ) ,
76  ( '03031014' , 'ÖÔË¼Ñ' , 'ÄÐ' , '1981-6-6' , 1.85 , 'Î÷18Éá421' ) ,
77  ( '03031033' , '²Î÷ÁÁ' , 'ÄÐ' , '1982-3-12' , 1.75 , 'Î÷18Éá423' ) ,
78  ( '03031051' , 'ÖÜ½£' , 'ÄÐ' , '1981-5-8' , 1.68 , 'Î÷18Éá422' ) ;
79  /*!40000 ALTER TABLE 's' ENABLE KEYS */;
80  UNLOCK TABLES;
81
82  —
83  — Table structure for table 'sc'
84  —
85
86  DROP TABLE IF EXISTS 'sc';
87  /*!40101 SET @saved_cs_client      = @@character_set_client */;
88  /*!40101 SET character_set_client = utf8 */;
89  CREATE TABLE 'sc' (

```

```

90 'SNO' char(8) NOT NULL,
91 'CNO' char(5) NOT NULL,
92 'GRADE' double(5,1) DEFAULT NULL,
93 KEY 'SNO' ('SNO'),
94 KEY 'CNO' ('CNO'),
95 CONSTRAINT 'sc_ibfk_1' FOREIGN KEY ('SNO') REFERENCES 's' ('SNO'),
96 CONSTRAINT 'sc_ibfk_2' FOREIGN KEY ('CNO') REFERENCES 'c' ('CNO')
97 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
98 /*!40101 SET character_set_client = @saved_cs_client */;
99
100 —
101 — Dumping data for table 'sc'
102 —
103
104 LOCK TABLES 'sc' WRITE;
105 /*!40000 ALTER TABLE 'sc' DISABLE KEYS */;
106 INSERT INTO 'sc' VALUES ('01032010', 'CS101', 82.0),
107 ('01032010', 'CS102', 91.0), ('01032010', 'CS103', 83.5),
108 ('01032001', 'CS101', 77.5), ('01032001', 'CS102', 85.0),
109 ('01032001', 'CS103', 83.0), ('01032005', 'CS101', 62.0),
110 ('01032005', 'CS102', 77.0), ('01032005', 'CS103', 82.0),
111 ('01032023', 'CS101', 55.0), ('01032023', 'CS102', 81.0),
112 ('01032023', 'CS103', 76.0), ('01032112', 'CS101', 88.0),
113 ('01032112', 'CS102', 91.5), ('01032112', 'CS103', 86.0),
114 ('03031033', 'EE101', 93.0), ('03031033', 'EE102', 89.0),
115 ('03031009', 'EE101', 88.0), ('03031009', 'EE102', 78.5),
116 ('03031011', 'EE101', 91.0), ('03031011', 'EE102', 86.0),
117 ('03031051', 'EE101', 78.0), ('03031051', 'EE102', 58.0),
118 ('03031014', 'EE102', 71.0), ('03031014', 'EE101', 79.0);
119 /*!40000 ALTER TABLE 'sc' ENABLE KEYS */;
120 UNLOCK TABLES;
121 /*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;
122
123 /*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
124 /*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
125 /*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
126 /*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
127 /*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
128 /*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```

```

129  /*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;
130
131  — Dump completed on 2017-04-20 13:50:54

```

7.2 Appendix II: Created DB and Created Table

```

mysql> use student;
Database changed
mysql> show tables;
+-----+
| Tables_in_student |
+-----+
| c                  |
| s                  |
| sc                 |
+-----+
3 rows in set (0.00 sec)

```

图 1: A.use database of student

```

mysql> desc s;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| SNO   | char(8)       | NO   | PRI | NULL    |       |
| SNAME | varchar(20)   | YES  |     | NULL    |       |
| SEX   | varchar(4)    | YES  |     | NULL    |       |
| BDATE | varchar(10)   | NO   |     | NULL    |       |
| HEIGHT | double(3,2)  | YES  |     | NULL    |       |
| DORM  | varchar(20)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.01 sec)

mysql> select * from s;
+-----+-----+-----+-----+-----+-----+
| SNO   | SNAME | SEX | BDATE   | HEIGHT | DORM   |
+-----+-----+-----+-----+-----+-----+
| 01032001 | 张晓梅 | 女 | 1983-11-17 | 1.58 | 西1舍312 |
| 01032005 | 刘静 | 女 | 1982-1-10 | 1.63 | 西1舍312 |
| 01032010 | 李涛 | 男 | 1982-4-5 | 1.72 | 西14舍221 |
| 01032023 | 孙文 | 男 | 1983-6-10 | 1.80 | 西14舍221 |
| 01032112 | 董卫锋 | 男 | 1982-2-20 | 1.71 | 西14舍221 |
| 03031009 | 田婷婷 | 女 | 1982-8-11 | 1.60 | 西2舍104 |
| 03031011 | 王倩 | 女 | 1983-12-20 | 1.66 | 西2舍104 |
| 03031014 | 赵思扬 | 男 | 1981-6-6 | 1.85 | 西18舍421 |
| 03031033 | 蔡明亮 | 男 | 1982-3-12 | 1.75 | 西18舍423 |
| 03031051 | 周剑 | 男 | 1981-5-8 | 1.68 | 西18舍422 |
+-----+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)

```

图 2: B. table s

```
mysql> desc c;
```

| Field | Type | Null | Key | Default | Extra |
|----------|-------------|------|-----|---------|-------|
| CNO | char(5) | NO | PRI | NULL | |
| LHOUR | double(5,0) | YES | | NULL | |
| CREDIT | char(1) | YES | | NULL | |
| SEMESTER | varchar(4) | YES | | NULL | |

4 rows in set (0.03 sec)

```
mysql> select * from c;
```

| CNO | LHOUR | CREDIT | SEMESTER |
|-------|-------|--------|----------|
| CS101 | 60 | 3 | 春 |
| CS102 | 80 | 4 | 春 |
| CS103 | 40 | 2 | 秋 |
| EE101 | 60 | 3 | 春 |
| EE102 | 100 | 5 | 秋 |

5 rows in set (0.00 sec)

图 3: C. table c

```
mysql> desc sc;
```

| Field | Type | Null | Key | Default | Extra |
|-------|-------------|------|-----|---------|-------|
| SNO | char(8) | NO | MUL | NULL | |
| CNO | char(5) | NO | MUL | NULL | |
| GRADE | double(5,1) | YES | | NULL | |

3 rows in set (0.01 sec)

```
mysql> select * from sc;
```

| SNO | CNO | GRADE |
|----------|-------|-------|
| 01032010 | CS101 | 82.0 |
| 01032010 | CS102 | 91.0 |
| 01032010 | CS103 | 83.5 |
| 01032001 | CS101 | 77.5 |
| 01032001 | CS102 | 85.0 |
| 01032001 | CS103 | 83.0 |
| 01032005 | CS101 | 62.0 |
| 01032005 | CS102 | 77.0 |
| 01032005 | CS103 | 82.0 |
| 01032023 | CS101 | 55.0 |
| 01032023 | CS102 | 81.0 |
| 01032023 | CS103 | 76.0 |
| 01032112 | CS101 | 88.0 |
| 01032112 | CS102 | 91.5 |
| 01032112 | CS103 | 86.0 |
| 03031033 | EE101 | 93.0 |
| 03031033 | EE102 | 89.0 |
| 03031009 | EE101 | 88.0 |
| 03031009 | EE102 | 78.5 |
| 03031011 | EE101 | 91.0 |
| 03031011 | EE102 | 86.0 |
| 03031051 | EE101 | 78.0 |
| 03031051 | EE102 | 58.0 |
| 03031014 | EE102 | 71.0 |
| 03031014 | EE101 | 79.0 |

25 rows in set (0.00 sec)

图 4: D. table sc

7.3 Appendix III: Query Results

```
mysql> SELECT SNO, GRADE
-> FROM SC
-> WHERE CNO = 'CS102';
```

| SNO | GRADE |
|----------|-------|
| 01032010 | 91.0 |
| 01032001 | 85.0 |
| 01032005 | 77.0 |
| 01032023 | 81.0 |
| 01032112 | 91.5 |

5 rows in set (0.05 sec)

```
mysql> SELECT SNAME
-> FROM S, SC
-> WHERE S.SNO = SC.SNO AND SEX = '女' AND CNO = 'EE101';
```

| SNAME |
|-------|
| 田婷婷 |
| 王倩 |

2 rows in set (0.00 sec)

图 5: A. 查询 1 和 2

```
mysql> SELECT DISTINCT SNAME
-> FROM S, SC
-> WHERE S.SNO = SC.SNO AND CNO NOT LIKE 'CS102';
```

| SNAME |
|-------|
| 张晓明 |
| 刘静 |
| 李涛 |
| 孙文 |
| 董卫锋 |
| 田婷婷 |
| 王倩 |
| 赵思扬 |
| 蔡明亮 |
| 周剑 |

10 rows in set (0.03 sec)

```
mysql> SELECT SNAME, SEX, BDATE
-> FROM S
-> WHERE BDATE < (SELECT BDATE
-> FROM S S2
-> WHERE S2.SNAME = '李涛');
```

| SNAME | SEX | BDATE |
|-------|-----|-----------|
| 刘静 | 女 | 1982-1-10 |
| 董卫锋 | 男 | 1982-2-20 |
| 赵思扬 | 男 | 1981-6-6 |
| 蔡明亮 | 男 | 1982-3-12 |
| 周剑 | 男 | 1981-5-8 |

5 rows in set (0.00 sec)

图 6: B. 查询 3 和 4

```
mysql> SELECT SNO
-> FROM SC
-> WHERE SC.GRADE = (SELECT MAX(GRADE)
-> FROM SC SC1);
```

| SNO |
|----------|
| 03031033 |

1 row in set (0.00 sec)

```
mysql> SELECT SNAME, C.CNO, CREDIT, GRADE
-> FROM S, C, SC
-> WHERE S.SNO = SC.SNO AND C.CNO = SC.CNO;
```

| SNAME | CNO | CREDIT | GRADE |
|-------|-------|--------|-------|
| 张晓梅 | CS101 | 3 | 77.5 |
| 刘静 | CS101 | 3 | 62.0 |
| 李涛 | CS101 | 3 | 82.0 |
| 孙文 | CS101 | 3 | 55.0 |
| 董卫锋 | CS101 | 3 | 88.0 |
| 张晓梅 | CS102 | 4 | 85.0 |
| 刘静 | CS102 | 4 | 77.0 |
| 李涛 | CS102 | 4 | 91.0 |
| 孙文 | CS102 | 4 | 81.0 |
| 董卫锋 | CS102 | 4 | 91.5 |
| 张晓梅 | CS103 | 2 | 83.0 |
| 刘静 | CS103 | 2 | 82.0 |
| 李涛 | CS103 | 2 | 83.5 |
| 孙文 | CS103 | 2 | 76.0 |
| 董卫锋 | CS103 | 2 | 86.0 |
| 田婷婷 | EE101 | 3 | 88.0 |
| 王倩倩 | EE101 | 3 | 91.0 |
| 赵思明 | EE101 | 3 | 79.0 |
| 蔡明亮 | EE101 | 3 | 93.0 |
| 周剑 | EE101 | 3 | 78.0 |
| 田婷婷 | EE102 | 5 | 78.5 |
| 王倩倩 | EE102 | 5 | 86.0 |
| 赵思明 | EE102 | 5 | 71.0 |
| 蔡明亮 | EE102 | 5 | 89.0 |
| 周剑 | EE102 | 5 | 58.0 |

25 rows in set (0.00 sec)

图 7: C. 查询 5 和 6

mysql> SELECT SNAME, AVG(GRADE)
-> FROM S, SC
-> WHERE S.SNO = SC.SNO AND GRADE > '80';

| SNAME | AVG(GRADE) |
|-------|------------|
| 张晓梅 | 86.6667 |

1 row in set (0.00 sec)

mysql> SELECT SUM(GRADE)
-> FROM SC
-> WHERE SNO IN (SELECT SC2.SNO
-> FROM SC SC2
-> WHERE SC2.GRADE > 60
-> GROUP BY SNO
-> HAVING COUNT(CNO) >= 3
->)
-> GROUP BY SNO;

| SUM(GRADE) |
|------------|
| 245.5 |
| 221.0 |
| 256.5 |
| 265.5 |

4 rows in set (0.10 sec)

图 8: D. 查询 7 和 8