

# 数据库实验二实验报告

尹小龙 201605301355

实验操作顺序模板如下（之后每步只列出 SQL 语句，不做重复说明）

- ①输入 SQL 语句
- ②点击“执行”按钮
- ③update dbtest set test=2
- ④点击“commit”按钮（一个蓝色边线的白色按钮）
- ⑤select \* from dbscore, 点击“执行”按钮  
查询所有实验的完成及出错情况
- ⑥select \* from dbrank, 点击“执行”按钮  
查询所有实验的得分情况以及班内排名

（注：写报告的时候发现，实验二的大部分实验都是在机房完成的，我的笔记本里没有保存实际执行的 sql 语句，所以暂时只放上第六步和第十步的 sql 语句，其他的步骤先打好框架，具体语句等下周三到机房统一补齐。实在抱歉…）

**第一步：找出没有选修任何课程的学生的学号、姓名(即没有选课记录的学生)**

sql 命令如下：

```
select sid, name
from pub.STUDENT
where sid not in ( select sid
                  from pub.STUDENT_COURSE)
```

**第二步：找出至少选修了学号为“200900130417”的学生所选修的一门课的学生的学号、姓名**

sql 命令如下：

```
create table test2_02 as
select c.SID, c.NAME
from pub.STUDENT_COURSE a, pub.STUDENT_COURSE b, pub.STUDENT c
where a.SID = '200900130417' and a.CID = b.CID and b.SID = c.SID
```

**第三步：找出至少选修了一门其先行课程号为“300002”号课程的学生的学号、姓名。**

sql 命令如下：

```
create table test2_03 as
select c.SID, c.NAME
from pub.COURSE a, pub.STUDENT_course b, pub.STUDENT c
where a.FCID = '300002' and b.SID = c.SID and a.CID = b.CID
```

第四步：找出选修了“操作系统”并且也选修了“数据结构”的学生的学号、姓名。

sql 命令如下：

```
create table test2_04 as select sid,name
from pub.student
where sid in(
    select sid
    from pub.student_course, pub.course
    where student_course.cid = course.cid
    and name = '操作系统')
and sid in(
    select sid
    from pub.student_course, pub.course
    where student_course.cid = course.cid
    and name = '数据结构')
```

第五步：查询 20 岁的所有有选课的学生学号、姓名、平均成绩(avg\_score, 此为列名,下同) (平均成绩四舍五入到个位)、总成绩(sum\_score)

sql 命令如下：

```
create table test2_05 as
select student.sid,name,cast(avg(score) as numeric(5,0)) avg_score,sum(score) sum_score
from pub.student, pub.student_course
where student.sid = student_course.sid and age = '20'
group by student.sid,name
```

第六步：查询所有课的最高成绩、次高成绩(次高成绩一定小于最高成绩)、最高成绩人数, test2\_06 有四个列: 课程号 cid、课程名称 name、最高成绩 max\_score、次高成绩 max\_score2、最高成绩人数 max\_score\_count (一个学生同一门课成绩都是第一, 只计一次)。如果没有学生选课, 则最高成绩为空值, 最高成绩人数为零。如果没有次高成绩, 则次高成绩为空值。

sql 命令如下: (未完成版本)

```
create table test2_06 as
with score_max(cid,max_score)
as( select cid,max(score)
    from pub.student_course
    group by cid),
temp(cid,sid,max_score_count)
as( select cid,sid,count(score)
    from pub.student_course sc
    where score=(select max_score
        from score_max
        where sc.cid=score_max.cid)
    group by cid,sid)

select    cid,name,max_score,
        (    select max(score)
```

```

        from pub.student_course sc1
        where sc1.score<( select max_score
                           from score_max
                           where sc1.cid=score_max.cid)
        group by cid ) max_score2,
    (   select count(sid)
        from temp
        group by cid) max_score_count
from pub.course natural left outer join score_max natural full outer join temp

```

简单解释一下我的思路。刚开始的时候我是打算直接在 **select** 里面定义三个函数，分别筛选出最大成绩、次大成绩和最大成绩人数。之后在写的时候发现后两个函数都需要选出最大成绩，所以这时候想到用 **with** 语句把最大成绩的筛选定义为一个临时函数，之后就可以直接调用函数，而不是大段的复制粘贴了，语句变得简洁了很多。

这样写完之后，点击执行，发现系统报错，错误为“单一查询返回多行结果”。我考虑可能是因为计算最大成绩人数的时候分组出了问题，于是先定义了一个临时函数 **temp** 计算筛选出每门课获得最大成绩的学生。然后再计算 **temp** 中每门课下面有几个学生。但是还是返回一样的错误。

最后跟同学交流的时候，参考别人的方法把所有的选择都定义为临时函数（用 **with/as** 语句定义了四个），得到了正确的结果。然而并没有解答为什么我的语句不行的问题。

所以我又把 **test2\_06** 删掉了，之后不断尝试新方法和排查语句错误，截止到书写报告的时候还是没有解决这个问题。

#### 第七步：查询所有不姓张、不姓李、也不姓王的学生的学号 **sid**、姓名 **name**

sql 命令如下：

```

create table test2_07 as
select sid,name
from pub.student
where name not in(
    select name
    from pub.student
    where name like '张%' or name like '李%' or name like '王%')

```

#### 第八步：查询学生表中每一个姓氏及其人数(不考虑复姓)，**test2\_08** 有两个列：**second\_name**、**p\_count**

sql 命令如下：

```

create table test2_08 as
select substr(name,1,1) second_name,count(*) p_count
from pub.student
group by substr(name,1,1)

```

**第九步：查询选修了 300003 号课程的学生的 sid、name、score**

sql 命令如下：

```
create table test2_09 as
select student.sid,student.name,score
from pub.student,pub.student_course
where student.sid = student_course.sid and cid = '300003'
```

**第十步：找出同一个同学同一门课程有两次或以上不及格的所有学生的学号、姓名（即一门课程需要补考两次或以上的学生的学号、姓名）。**

sql 命令如下：

```
create table test2_10 as
select sid,name
from pub.student
where sid in(
    select sid
    from pub.student_course
    where score<60
    group by sid,cid
    having count(cid)>=2)
```

**总结：**

二考察的是查询语句的综合运用，其中包括了条件判断、表的联结以及临时函数定义，是我对于查询语句的概念和运用有了更深入的理解。