

数据库实验四实验报告

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=4
 - 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

第一步：将 pub 用户下表 student_41 及数据复制到主用户的表 test4_01 中，使用 alter table 语句为表增加列：“总成绩:sum_score”。使用 update 语句,利用 pub.student_course、pub.course，统计“总成绩”。

sql 命令如下：

```
create table test4_01 as
(select * from pub.STUDENT_41)

alter table test4_01
add (sum_score numeric(6, 1))

update test4_01
set sum_score = (
    select sum(score)
    from pub.STUDENT_COURSE
    where pub.STUDENT_COURSE.SID = test4_01.SID
    group by pub.STUDENT_COURSE.SID);
```

第二步：将 pub 用户下表 student_41 及数据复制到主用户的表 test4_02 中，使用 alter table 语句为表增加列“平均成绩:avg_score”（小数点后保留 1 位）。利用 pub.student_course、pub.course，统计“平均成绩”，四舍五入到小数点后 1 位。

sql 命令如下：

```
create table test4_02 as (select * from pub.STUDENT_41)
alter table test4_02 add (avg_score numeric(4, 1))
update test4_02
set avg_score = (select round(avg(score),1)
    from pub.STUDENT_COURSE
    where pub.STUDENT_COURSE.SID = test4_02.SID);
```

第三步：将 pub 用户下表 student_41 及数据复制到主用户的表 test4_03 中，使用 alter table 语句为表增加列：“总学分:sum_credit”。使用 update 语句,利用 pub.student_course、pub.course，统计“总学分”。这是需要注意：成绩及格才能够计算所得学分,一门课多个成绩都及格只计一次学分。

sql 命令如下：

```
create table test4_03 as
(select * from pub.STUDENT_41)

alter table test4_03 add
(sum_credit numeric(8, 1))

update test4_03
set sum_credit = (
    select sum(credit)
    from pub.COURSE , pub.STUDENT_COURSE
    where    pub.STUDENT_COURSE.SID = test4_03.SID
            and pub.STUDENT_COURSE.SCORE >= 60
            and pub.STUDENT_COURSE.CID = pub.COURSE.CID
    group by sid)
```

第四步：将 pub 用户下表 student_41 及数据复制到主用户的表 test4_04 中。根据列院系名称 dname 到 pub.department 找到对应院系编号 did，将对应的院系编号回填到院系名称列 dname 中，如果表中没有对应的院系名称，则列 dname 中内容不变仍然是原来的内容。

sql 命令如下：

```
create table test4_04 as
(select * from pub.STUDENT_41)

update test4_04
set dname = (
    select did
    from pub.DEPARTMENT
    where test4_04.DNAME = pub.DEPARTMENT.DNAME)
where exists(
    select *
    from pub.DEPARTMENT
    where test4_04.DNAME = pub.DEPARTMENT.DNAME)
```

第五步：将 pub 用户下表 student_41 及数据复制到主用户的表 test4_05 中，使用 alter table 语句为表增加 4 个列：“总成绩:sum_score”、“平均成绩:avg_score”、“总学分:sum_credit”、“院系编号:did varchar(2)”。(1) 利用 pub.student_course、pub.course，统计“总成绩”；(2) 利用 pub.student_course、pub.course，统计“平均成绩”，四舍五入到小数点后 1 位；(3) 利用 pub.student_course、pub.course，统计“总学分”；(4) 根据院系名称到 pub.department 或者 pub.department_41 中，找到对应编号，填写到院系编号中，如果都没有对应的院系，则填写为 00。

sql 命令如下：

```
create table test4_05 as select * from pub.STUDENT_41

alter table test4_05 add (sum_score numeric(6, 1))
update test4_05
set sum_score = (select sum(score)
                  from pub.STUDENT_COURSE
                  where pub.STUDENT_COURSE.SID = test4_05.SID);

alter table test4_05 add (avg_score numeric(4, 1))
update test4_05
set avg_score = (select round(avg(score),1)
                  from pub.STUDENT_COURSE
                  where pub.STUDENT_COURSE.SID = test4_05.SID);

alter table test4_05 add (sum_credit numeric(4, 1))
update test4_05
set sum_credit = (select sum(credit)
                  from pub.COURSE, pub.STUDENT_COURSE
                  where pub.STUDENT_COURSE.SID = test4_05.SID
                  and pub.STUDENT_COURSE.SCORE >= 60
                  and pub.STUDENT_COURSE.CID = pub.COURSE.CID);

alter table test4_05 add (did varchar(2))
update test4_05
set did= (select did
           from pub.DEPARTMENT
           where test4_05.DNAME = pub.DEPARTMENT.DNAME)
where exists(select *
             from pub.DEPARTMENT
             where test4_05.DNAME = pub.DEPARTMENT.DNAME)

update test4_05
set did = (select did
           from pub.DEPARTMENT_41
           where test4_05.DNAME = pub.DEPARTMENT_41.DNAME)
where exists(select *
```

```
from pub.DEPARTMENT_41
where test4_05.DNAME = pub.DEPARTMENT_41.DNAME)

update test4_05
set did = '00'
where did is null
```

第六步：将 pub 用户下的 Student_42 及数据复制到主用户的表 test4_06 中,对表中的数据进行整理，修复那些不规范的数据：

剔除姓名列中的所有空格；。

sql 命令如下：

```
create table test4_06 as
(select * from pub.STUDENT_42)

update test4_06
set name = (replace(name, ' ', ''))
```

第七步：将 pub 用户下的 Student_42 及数据复制到主用户的表 test4_07 中,对表中的数据进行整理，修复那些不规范的数据：对性别列进行规范（需要先确定哪些性别数据不规范，也就是那些和大多数不一样的就是不规范的）。

sql 命令如下：

```
create table test4_07 as
(select * from pub.STUDENT_42)

update test4_07
set sex = (replace(sex, ' ', ''))

update test4_07
set sex = (replace(sex, '性', ''))
```

第八步：将 pub 用户下的 Student_42 及数据复制到主用户的表 test4_08 中,对表中的数据进行整理，修复那些不规范的数据：对班级列进行规范（需要先确定哪些班级不规范）。

sql 命令如下：

```
create table test4_08 as
(select * from pub.STUDENT_42)

update test4_08
set class = (replace(class, '级', ''))
```

第九步：将 pub 用户下的 Student_42 及数据复制到主用户的表 test4_09 中,对表中的数据进行整理,修复那些不规范的数据: 年龄为空值的根据出生日期设置学生年龄 (截止到 2012 年的年龄, 即年龄=2012-出生年份), 年龄不为空值的不要改变。

sql 命令如下:

```
create table test4_09 as
(select * from pub.STUDENT_42)

update test4_09
set age = (2012 - extract(year from birthday))
where age is NULL
```

第十步：将 pub 用户下的 Student_42 及数据复制到主用户的表 test4_10 中,对表中的数据进行整理,修复那些不规范的数据: (1) 剔除姓名列中的所有空格; (2) 剔除院系名称列中的所有空格; (3) 对性别列进行规范 (需要先确定哪些性别数据不规范, 也就是那些和大多数不一样的就是不规范的); (4) 对班级列进行规范 (需要先确定哪些班级不规范)。 (5) 年龄为空值的根据出生日期设置学生年龄 (截止到 2012 年的年龄, 即年龄=2012-出生年份), 年龄不为空值的不要改变。

sql 命令如下:

```
create table test4_10 as
(select * from pub.STUDENT_42)

update test4_10
set name = (replace(name, ' ', ''))

update test4_10
set dname = (replace(dname, ' ', ''))

update test4_10
set sex = (replace(sex, ' ', ''))

update test4_10
set sex = (replace(sex, '性', ''))

update test4_10
set class = (replace(class, '级', ''))

update test4_10
set age = (2012 - extract(year from birthday))
where age is NULL
```