郑州昇達經貿管理學院

ZhengZhou Shengda University Of Economics, Business & Management

《数据库原理与应用》

实

验

指

导

书

信息工程学院

2017 年 1 月

目　录

实验一 基本表定义 1

实验二 单表查询 6

实验三 连接查询 8

实验四 嵌套查询 9

实验五 更新、视图 11

实验六 存储过程 13

实验七 触发器 15

实验八 访问控制 16

实验一 基本表定义

实验类型：验证性

实验学时：4 学时

一、实验目的

掌握 sql 语句定义基本表，理解完整性约束定义方法。

二、实验内容

使用 select 语句完成以下操作，（1）或（2）二选一：

（1）建立教学数据库基本表,定义主码，姓名不能重复，成绩小于等于 100，并录入模拟数据。

student(sno,sname,ssex,sage,sdept)

course(cno,cname,cpno,ccredit)

sc(sno,cno,grade)

（2）建立供应关系数据库基本表,定义外码，录入模拟数据。

s(sno,sname,status,city)

p(pno,pname,color,weight)

j(jno,jname,city)

spj(sno,pno,jno,qty)

三、实验结果参考

本指导书的实验参考均以 ORACLE 为例。

（1）

```
create  table  student (
    sno  char(12)  not null ,
    sname  varchar2(20) unique,
    ssex  char(3) not null check(ssex='男' or ssex='女'),
    sage  number(3) not null,
    sdept  varchar2(20) ,
    primary  key(sno)
);
create  table course (
```

```
    cno varchar2(6) not null ,
    cname varchar2(20) not null ,
    cpno varchar2(6) ,
    ccredit number(2) ,
    primary  key (cno)
    );

create table sc (
    sno char(12)  not null ,
    cno varchar2(6) not null ,
    grade number(3) check(grade<=100) ,
    primary key (sno,cno)
) ;
insert into student(sno, sname, ssex, sage, sdept)
values('200215121', '李勇', '男',20,'cs');
insert into student(sno, sname, ssex, sage, sdept)
values('200215122', '刘晨', '女',19,'cs');
insert into student(sno, sname, ssex, sage, sdept)
values('200215123', '王敏', '女',18,'ma');
insert into student(sno, sname, ssex, sage, sdept)
values('200215125', '张立', '男',19,'is');

insert into course(cno, cname, cpno ,ccredit)
values('1', '数据库',5,4);
insert into course(cno, cname, cpno ,ccredit)
values('2', '数学',null,2);
insert into course(cno, cname, cpno ,ccredit)
values('3', '信息系统',1,4);
insert into course(cno, cname, cpno ,ccredit)
values('4', '操作系统',6,3);
insert into course(cno, cname, cpno ,ccredit)
values('5', '数据结构',7,4);
insert into course(cno, cname, cpno ,ccredit)
```

```
values('6', '数据处理',null,2);
insert into course(cno, cname, cpno ,ccredit)
values('7', 'pascal 语言',6,4);

insert into sc(sno, cno, grade)
values('200215121','1',92);
insert into sc(sno, cno, grade)
values('200215121','2',85);
insert into sc(sno, cno, grade)
values('200215121','3',88);
insert into sc(sno, cno, grade)
values('200215122','2',90);
insert into sc(sno, cno, grade)
values('200215122','3',80);
```

（2）

```
create table s (
    sno varchar2(10) not null ,
    sname varchar2(50)  ,
    status number(4) ,
    city varchar2(20)  ,
    primary key (sno)
) ;

create table p (
    pno varchar2(10)  not null ,
    pname varchar2(50),
    color varchar2(10),
    weight  number(4),
    primary key (pno)
) ;

create table j (
```

```
    jno varchar2(10)  not null ,
    jname varchar2(50) ,
    city varchar2(20)  ,
    primary key(jno)
) ;
create table spj (
    sno varchar2 (10)  not null ,
    pno varchar2 (10)  not null ,
    jno varchar2 (10)  not null ,
    qty  number(8) ,
    primary key (sno,pno,jno),
    foreign key (sno) references s(sno),
    foreign key (pno) references p(pno),
    foreign key (jno) references j(jno)
);

insert into s(sno, sname, status, city)
values('s1','精益',20,'天津');
insert into s(sno, sname, status, city)
values('s2','盛锡',10,'北京');
insert into s(sno, sname, status, city)
values('s3','东方红',30,'北京');
insert into s(sno, sname, status, city)
values('s4','丰泰盛',20,'天津');
insert into s(sno, sname, status, city)
values('s5','为民',30,'上海');
insert into p(pno, pname, color, weight)
values('p1','螺母','红',12);
insert into p(pno, pname, color, weight)
values('p2','螺栓','绿',17);
insert into p(pno, pname, color, weight)
values('p3','螺丝刀','蓝',14);
insert into p(pno, pname, color, weight)
```

```
values('p4','螺丝刀','红',14);
insert into p(pno, pname, color, weight)
values('p5','凸轮','蓝',40);
insert into p(pno, pname, color, weight)
values('p6','齿轮','红',30);
insert into j(jno, jname, city)
values('j1','三建','北京');
insert into j(jno, jname, city)
values('j2','一气','长春');
insert into j(jno, jname, city)
values('j3','弹簧厂','天津');
insert into j(jno, jname, city)
values('j4','造船厂','天津');
insert into j(jno, jname, city)
values('j5','机床厂','唐山');
insert into j(jno, jname, city)
values('j6','无线电厂','常州');
insert into j(jno, jname, city)
values('j7','半导体厂','南京');
insert into spj(sno, pno, jno, qty)
values('s1','p1','j1',200);
insert into spj(sno, pno, jno, qty)
values('s1','p1','j3',100);
insert into spj(sno, pno, jno, qty)
values('s1','p1','j4',700);
insert into spj(sno, pno, jno, qty)
values('s1','p2','j2',100);
insert into spj(sno, pno, jno, qty)
values('s2','p3','j1',400);
insert into spj(sno, pno, jno, qty)
values('s2','p3','j2',200);
insert into spj(sno, pno, jno, qty)
values('s2','p3','j4',500);
```

```
insert into spj(sno, pno, jno, qty)
values('s2','p3','j5',400);
insert into spj(sno, pno, jno, qty)
values('s2','p5','j1',400);
insert into spj(sno, pno, jno, qty)
values('s2','p5','j2',100);
insert into spj(sno, pno, jno, qty)
values('s3','p1','j1',200);
insert into spj(sno, pno, jno, qty)
values('s3','p3','j1',200);
insert into spj(sno, pno, jno, qty)
values('s4','p5','j1',100);
insert into spj(sno, pno, jno, qty)
values('s4','p6','j3',300);
insert into spj(sno, pno, jno, qty)
values('s4','p6','j4',200);
insert into spj(sno, pno, jno, qty)
values('s5','p2','j4',100);
insert into spj(sno, pno, jno, qty)
values('s5','p3','j1',200);
insert into spj(sno, pno, jno, qty)
values('s5','p6','j2',200);
insert into spj(sno, pno, jno, qty)
values('s5','p6','j4',500);
```

实验二 单表查询

实验类型：验证性

实验学时：4 学时

一、实验目的

学习 sql 连接查询，掌握单表查询方法。

二、实验内容

使用 select 语句完成以下操作：

（1）查询“cs”系，姓“刘”的学生学号和姓名；

（2）查询先行课（cpno）为空的课程名称；

（3）查询选修了“1”号课程的学生学号，按学号由小到大排列；

（4）统计选修了课程的学生人数；

（5）查询选修了“1”号课程，并且成绩 90 分以上的学生人数；

（6）统计各门课程的选修人数；

（7）统计每个学生的选课门数；

（8）查询选修的课程超过二门的学生学号及选课门数；

（9）查询每个系男女生的人数、平均年龄；

三、实验结果参考

（1）

```
select sno,sname from student where lower(sdept)='cs' and sname like'刘%';
```

（2）

```
select cname from course where cpno is null;
```

（3）

```
select sno from sc where cno='1' order by sno;
```

（4）

```
select count(distinct sno) from sc ;
```

（5）

```
select count(sno) from sc where cno='1' and grade>=90;
```

（6）

```
select cno,count(sno) from sc group by cno;
```

（7）

```
select sno,count(cno) from sc group by sno;
```

（8）

```
select sno,count(cno) from sc group by sno having count(cno)>2;
```

（9）

```
select sdept,ssex,count(sno),avg(sage)  from student group by sdept,ssex;
```

实验三 连接查询

实验类型：验证性

实验学时：4 学时

一、实验目的

学习 sql 连接查询，掌握连接查询方法。

二、实验内容

使用 select 语句完成以下操作：

（1）查询“cs”系男学的选课情况

（2）查询选修了“数据库”课程的学生姓名

（3）查询选修了“数据库”课程，并且成绩 90 分以上的学生人数、平均成绩

（4）统计各系选课的学生人数、课程门数

（5）统计各系男女生选课的学生人数、课程门数

（6）查询选修课超过二门的学生学号、姓名

（7）使用了“绿”或者“蓝”色零件的工程编号、名称，要求按工程编号从小到大排序

（8）使用了零件“凸轮”的工程数量

（9）统计各供应商供应工程的数量，只要求统计工程数量 3 个以上的，并显示供应商的编号和名称，同时按供应商编号排序

（10）查询各“北京”供应商供应的工程数量，要求显示供应商的编号和名称

三、实验结果参考

（1）

select sc.* from sc,student where sc.sno=student.sno and lower(sdept)='cs' and ssex='男';

（2）

select sname from sc,course,student where sc.cno=course.cno and sc.sno=student.sno and cname='数据库' ;

（3）

select count(sno),avg(grade) from sc,course where sc.cno=course.cno and cname='数据库' and grade>=90;

（4）

Select sdept,count(distinct student.sno),count(distinct cno) from sc,student where sc.sno=student.sno group by sdept;

（5）

Select sdept,ssex,count(distinct student.sno),count(distinct cno) from sc,student where sc.sno=student.sno group by sdept,ssex;

（6）

Select student.sno,sname,count(cno) from sc,student where sc.sno=student.sno group by student.sno,sname having count(cno)>2;

（7）

Select distinct j.jno,jname from spj,p,j where spj.pno=p.pno and spj.jno=j.jno and (color='绿' or color='蓝') order by j.jno;

（8）

Select count(distinct jno) from spj,p where spj.pno=p.pno and pname like '%凸轮%';

（9）

Select s.sno,sname,count(distinct jno) from spj,s where spj.sno=s.sno group by s.sno,sname having count(distinct jno)>=3;

（10）

Select s.sno,sname,count(distinct jno) from spj,s where spj.sno=s.sno and city='北京' group by s.sno,sname;

实验四 嵌套查询

实验类型：验证性

实验学时：4 学时

一、实验目的

学习 sql 嵌套查询，掌握嵌套查询方法。

二、实验内容

使用 in 或 exists 谓词完成以下操作：

（1）查询没有选择任何课程的学生

（2）查询没有选择 1 号课的学生

（3）查询没有选择“数据库”课程的学生

（4）查询“200215121”号学生没有选择的课程

（5）没有参与选课的男生的学生姓名

（6）没有使用“p2”零件的工程名称

（7）没有使用“s2”供应商零件的工程名称

（8）查询没有使用了红色零件的工程编号和工程名称

（9）没有人选修的课程

（10）查询选择了所有课程的学生学号和姓名

三、实验程序

（1）

Select * from student where sno not in (select sno from sc);

Select * from student where not exists (select * from sc where sno=student.sno);

（2）

Select * from student where sno not in (select sno from sc where cno='1');

Select * from student where not exists (select * from sc where cno='1' and sno=student.sno);

（3）

Select * from student where sno not in (select sno from sc,course where sc.cno=course.cno and cname='数据库');

Select * from student where not exists (select * from sc,course where sc.cno=course.cno and cname='数据库' and sno=student.sno);

（4）

Select * from course where cno not in (select cno from sc where sno='200215121');

Select * from course where not exists (select cno from sc where sno='200215121' and cno=course.cno);

（5）

Select sname from student where sno not in (select sno from sc) and ssex='男';

Select sname from student where not exists (select * from sc where sno=student.sno) and ssex='男';

（6）

Select jname from j where jno not in (select jno from spj where pno='p2');

Select jname from j where not exists (select * from spj where pno='p2' and jno=j.jno);

（7）

Select jname from j where jno not in (select jno from spj where sno='s2');

Select jname from j where not exists (select * from spj where sno='s2' and jno=j.jno);

（8）

Select jno,jname from j where jno not in (select jno from spj,p where spj.pno=p.pno and color='红');

Select jno,jname from j where not exists (select * from spj,p where spj.pno=p.pno and color='红' and jno=j.jno);

（9）

Select * from course where cno not in (select cno from sc);

Select * from course where not exists (select * from sc where cno=course.cno);

（10）

```
Select sno,sname from student
  Where not exists (select * from course
      Where not exists (select * from sc
         Where sc.cno=course.cno and sc.sno=student.sno));
```

补充运行下列语句后，再试试。

```
insert into sc(sno, cno, grade)
values('200215121','4',92);
insert into sc(sno, cno, grade)
values('200215121','5',85);
insert into sc(sno, cno, grade)
values('200215121','6',88);
insert into sc(sno, cno, grade)
values('200215121','7',88);
```

实验五 更新、视图

实验类型：验证性

实验学时：4 学时

一、实验目的

学习 sql 数据更新、视图定义和使用。

二、实验内容

（1）将‘1’号课名称改为“数据库概论”、学分为 3

（2）增加选课元组，（‘200215125’，‘1’，‘85’）、（‘200215125’，‘2’，‘88’）

（3）将‘is’系的选课成绩清零

（4）删除‘is’系的选课记录

（5）将系别改为小写字母(参见 lower 函数)

（6）将男生姓名前增加字符 a（参见字符串连接函数 concat（串 1，串 2））

（7）将零件的 color 增加一个‘色’字，如：“红色”、“蓝色”等

（8）删除“s5”供应商所有信息（涉及到 s，spj 两张表）

（9）创建反映 cs 系学生信息视图

（10）创建反映学生平均成绩的视图（包括学号、姓名、平均成绩）

三、实验结果参考

（1）

```
Update course set cname='数据库概论',ccredit=3 where cno='1';
```

（2）

```
insert into sc(sno, cno, grade)
values('200215125','1',85);
insert into sc(sno, cno, grade)
values('200215125','2',88);
```

（3）

```
Update sc set grade=0 where sno in (select sno from student where lower(sdept)='is');
```

或者

```
Update sc set grade=0 where exists (select * from student where lower(sdept)='is' and sno=sc.sno);
```

（4）

```
Delete from sc  where sno in (select sno from student where lower(sdept)='is');
```

或者

```
Delete from sc where exists (select * from student where lower(sdept)='is' and sno=sc.sno);
```

（5）

```
Update student set sdept=lower(sdept);
```

（6）

```
Update student set sname=concat('a',sname) where ssex='男';
```

（7）

```
Update p set color= concat(color,'色');
```

（8）

```
Delete from spj where sno='s5';
Delete from s where sno='s5';
/*注意两个语句执行顺序不能颠倒，否则违反参考完整性约束*/
```

（9）

```
create view cs_student
      as
      select sno,sname,sage
      from  student
      where  lower(sdept)= 'cs'
```

（10）

```
create view avg1 (学号,姓名,平均成绩)
      as
      select sc.sno,sname,avg(grade)
      from  sc,student
      where  sc.sno=student.sno
     group by sc.sno,sname;
```

实验六 存储过程

实验类型：验证性

实验学时：2 学时

一、实验目的

学习存储过程定义和使用。

二、实验内容

（1）定义插入 course 表元组存储过程，执行存储过程

（2）定义查询指定课程名称的选课人数的存储过程，执行存储过程

（3）定义删除 sc 表指定学号的选课元组存储过程，执行存储过程

三、实验结果参考

（1）

```
create or replace procedure insert_course (no in char, name in char,pno in char,credit in number)
as
begin
insert into course(cno,cname,cpno,ccredit)
  values(no,name,pno,credit);
end;
--执行存储过程
exec insert_course  ('8','程序设计','1',3);
```

（2）

```
create or replace procedure  s1(name in char, num out number)
as
begin
select count(sno) into num  from sc,course  where sc.cno=course.cno and cname=name;
end;
--执行存储过程
var num1 number;
exec s1 ('数据库',:num1);
print  num1;
```

（3）

```
create or replace procedure delete_sc (no in char)
as
begin
```

```
delete from sc where sno=no;
end;
--执行存储过程
exec delete_sc ( '200215122');
```

实验七 触发器

实验类型：验证性

实验学时：2 学时

一、实验目的

学习触发器定义和使用。

二、实验内容

建立一下两个关系模式：

书目 c（isbn，书名 bname，数量 qty）

图书 b（bid,isbn,状态 status）。

（1）创建图书 b 触发器，插入图书后，增加书目的数量

（2）创建图书 b 触发器，删除图书后，减少书目的数量

三、实验结果参考

```
Create table c(
      ISBN  varchar2(30)  primary key,
      Bname varchar2(50),
      QTY   number(4)
);
Create table b(
      BID  varchar2(30)  primary key,
      ISBN varchar2(30),
      STATUS   char(3),
  foreign key (ISBN) references c(ISBN)
);
Insert into c values ('nn001','数学',0);
Insert into c values ('nn002','语文',0);
```

（1）

```
create or replace trigger b_insert
after insert on b /* 修改后触发 */
for each row  /* 行级触发器*/
begin
   update c set qty=qty+1 where isbn=:new.isbn;
end;
//执行触发器
insert into b values ('1','nn001',0);
insert into b values ('2','nn001',0);
insert into b values ('3','nn002',0);
insert into b values ('4','nn002',0);
```

（2）

```
create or replace trigger b_delete
after delete on b /* 修改后触发 */
for each row  /* 行级触发器*/
begin
   update c set qty=qty-1 where isbn=:old.isbn;
end;
//执行触发器
delete from b where bid='2';
delete from b where bid='4';
```

实验八 访问控制

实验类型：验证性

实验学时：2 学时

一、实验目的

学习 dbms 访问控制。

二、实验内容

（1）新建用户 shengda，密码 123

（2）给 shengda 授权，授予 create session(访问权限) ,create table（建表权限）

（3）指定存储空间，10m 表空间

（4）用户 shengda 数据库连接

（5）建立教学数据库的表

（6）查看表空间有哪些表

（7）执行有关的操作语句

三、实验结果参考

（1）新建用户 shengda，密码 123。

```
sql>create user shengda identified by 123;
```

（2）给 shengda 授权，授予 create session(访问权限) ,create table（建表权限）。

```
sql>grant create session ,create table to shengda;
```

（3）指定存储空间，10m 表空间。

```
sql>alter user shengda
     default tablespace sysaux
     quota 10m on sysaux ;
```

（4）建立数据库连接

```
sql>connect shengda /123
```

（5）-（7）略