

实验报告5-20301174-万兴全

存储过程源代码 + 测试执行*过程结果**输出**的截图或者文字*;

触发器源代码 + 测试执行过程结果*输出*的截图或者文字。

检查: **开发环境下运行存储过程和触发器**

1. 存储过程（60分，每题10分）

在Oracle下用PL/SQL语句定义存储过程(太痛苦了)

(1) 创建一个能向学生表student中插入一条记录的存储过程insert_student, 该过程需要5个参数, 分别用来传递学号、姓名、性别、出生日期、班级号。

```
1  create procedure insert_student(  
2  ssno in student.sno%type,  
3  ssname in student.sname%type,  
4  sssex in student.ssex%type,  
5  ssbirthday in date,  
6  sclassno in student.classno%type  
7  )  
8  is  
9  begin  
10     insert into  
11     student(sno,sname,ssex,sbirthday,classno)values  
12     (ssno,ssname,sssex,ssbirthday,sclassno);  
13 end;
```

```
561
562 ✓ create or replace procedure insert_student(
563     ssno in student.sno%type,
564     ssname in student.sname%type,
565     sssex in student.ssex%type,
566     ssbirthday in date,
567     sclassno in student.classno%type
568 )
569 is
570 begin
571     insert into student(sno,sname,ssex,sbirthday,classno)values (ssno,ssn
572 end;
```

insert_student()

[2022-04-22 01:19:40] 在 34 ms 内完成

```
C#RVANXQ> create or replace procedure insert_student(
    ssno in student.sno%type,
    ssname in student.sname%type,
    sssex in student.ssex%type,
    ssbirthday in date,
    sclassno in student.classno%type
)
is
begin
    insert into student(sno,sname,ssex,sbirthday,classno)values (ssno,ssname,sssex,ssbirthday,s
end;
```

[2022-04-22 01:21:15] 在 17 ms 内完成

写出调用存储过程insert_student的SQL语句，向数据表student中插入一个新同学，并提供相应的实参值。

调用前student表：

SNO	SNAME	SSEX	SBIRTHDAY	CLASSNO	TOTALCREDIT
20300007	赵坤	男	2002-08-21	RJ2003	
20300075	王娜娜	女	2003-09-23	RJ2003	
20300088	秦健	男	2004-03-01	RJ2003	
20300100	田邦仪	女	2003-02-26	RJ2004	
20300148	赵心砚	男	2002-04-25	RJ2005	
20300150	杨青	女	2004-11-15	RJ2005	
20300160	杨玲玲	女	2002-12-12	RJ2006	
20301168	rr	男	<null>	<null>	<null>
20301167	qfs	男	<null>	<null>	<null>
1	a	男	<null>	<null>	
2	b	男	<null>	<null>	
3	c	男	<null>	<null>	
4	d	男	<null>	<null>	
5	e	男	<null>	<null>	
6	f	男	<null>	<null>	
7	g	男	<null>	<null>	
8	h	男	<null>	<null>	
9	i	男	<null>	<null>	
10	j	男	<null>	<null>	
11	k	男	<null>	<null>	
12	l	男	<null>	<null>	
13	m	男	<null>	<null>	
14	n	男	<null>	<null>	
15	o	男	<null>	<null>	
16	p	男	<null>	<null>	

```

1 begin
2     insert_student('20301174','万兴
   全','男',to_date('2001/3/20','yyyy/mm/dd'),'RJ2006');
3 end;

```

574	✓	begin
575		insert_student('20301174','万兴全','男',to_date('2001/3/20','yyyy/mm/dd'));
576		end;
577	✓	select *from student;

SNO	SNAME	SSEX	SBIRTHDAY	CLASSNO	TOTALCREDIT
27	13	m	男	<null>	<null>
28	14	n	男	<null>	<null>
29	15	o	男	<null>	<null>
30	16	p	男	<null>	<null>
31	20301174	万兴全	男	2001-03-20	RJ2006

(2) 创建一个向课程表course中插入一门新课程的存储过程insert_course, 该存储过程需要三个参数, 分别用来传递课程号、课程名、学分, 但允许参数“学分”的默认值为4, 即当调用存储过程insert_course时, 未给第三个参数“学分”提供实参值时, 存储过程将按默认值4进行运算。

```
1 create or replace procedure insert_course(  
2   ccno course.cno%type,  
3   ccname course.cname%type,  
4   cccredit int:=4  
5 )  
6 is  
7     begin  
8         insert into course (cno, cname, ccredit)  
9     values (ccno,ccname,cccredit);  
10    end;
```

```
end;  
[2022-04-22 01:43:22] 在 11 ms 内完成  
C##VANXQ> create or replace procedure insert_course(  
    ccno course.cno%type,  
    ccname course.cname%type,  
    cccredit int:=4  
    )  
    is  
        begin  
            insert into course (cno, cname, ccredit) values (ccno,ccname,cccredit);  
        end;  
[2022-04-22 01:43:40] 在 2 ms 内完成
```

调用存储过程insert_course, 向课程表course中插入一门新课程。分两种情况 (给出第三个参数和未给出第三个参数) 写出相应的SQL命令, 并比较结果。

插入前的course表

WHERE		ORDER BY	
	CNO	CNAME	CCREDIT
1	800001	计算机基础	4
2	800002	程序设计语言	4
3	800003	数据结构	4
4	810011	数据库系统	4
5	810013	计算机网络	3
6	810015	微机原理与应用	4

```

1 begin
2     insert_course(000000,'体育课');
3 end;
4 begin
5     insert_course(000001,'不上课',1);
6 end;
7 select * from course;

```

1	✓	begin
2		insert_course(ccno: 000000, ccname: '体育课');
3		end;
4	✓	begin
5		insert_course(ccno: 000001, ccname: '不上课', cccredit: 1);
6		end;
7	✓	select * from course;

	CNO	CNAME	CCREDIT
3	800003	数据结构	4
4	810011	数据库系统	4
5	810013	计算机网络	3
6	810015	微机原理与应用	4
7	0	体育课	4
8	1	不上课	1

(3) 创建一个名称为query_student的存储过程，该存储过程的功能是从数据表student中根据学号查询某一同学的姓名、性别、出生日期、班级号。

```

1 create or replace procedure query_student (
2     ssno in student.sno%type
3 )

```

```

4  as
5  sname student.sname%type;
6  ssex student.ssex%type;
7  sbirthday student.sbirthday%type;
8  sclassno student.classno%type;
9  begin
10 select sname,ssex,sbirthday,classno
11 into sname,sssex,ssbirthday,sclassno
12 from student
13 where sno=ssno;
14 END;

```

调用存储过程query_student，查询学号为“20300150”的姓名、性别、出生日期、班级号，并写出完成此功能的SQL命令。

```

1  call INSERT_STUDENT('20301150','谁
   啊','男',to_date('2000/1/1','yyyy/mm/dd'),'RJ2002');
2
3  begin query_student('20301150');
4  END;

```

```

C##VANKQ> call INSERT_STUDENT('20301150','谁啊','男',to_date('2000/1/1','yyyy/mm/dd'),'RJ2002')
[2022-04-22 17:14:51] 在 17 ms 内完成
C##VANKQ> begin query_student('20301150');
           END;
[2022-04-22 17:14:51] 在 1 ms 内完成

```

(4) 建立存储过程，输出平均成绩大于80分的学生的姓名、性别、年龄和平均成绩。

```

1  create or replace procedure at_least_80
2  as
3  sname student.sname%type;
4  ssex student.ssex%type;

```

```

5  ssage student.age%type;
6  agrade sc.grade%type;
7
8  begin
9      select sname,ssex,age,avg(grade) as ag
10     into ssname,sssex,ssage,agrade
11     from student,sc
12     where sc.sno=student.sno and agrade>80
13     group by sname, ssex, age;
14 end;
15
16 call at_least_80();

```

调用该存储过程，并输出相应的结果。

```

✓ create or replace procedure at_least_80
as
  ssname student.sname%type;
  sssex student.ssex%type;
  ssage student.age%type;
  agrade sc.grade%type;
begin
  select sname,ssex,age,avg(grade) as ag
  into ssname,sssex,ssage,agrade
  from student,sc
  where sc.sno=student.sno and agrade>80
  group by sname, ssex, age;
end;
✓ call at_least_80();

```

	SNAME	SSEX	AGE	AVG(GRADE)
1	杨敏	女	18	91.3333333333333333333333333333...
2	赵鸿泽	男	18	100
3	杨青	女	18	92
4	rr	男	18	96.3333333333333333333333333333...
5	a	男	18	100

(5) 写存储过程显示所有选择了与给定学生姓名选择的全部课程的学生学号、姓名、选课数、平均成绩、总学分

```

1 create or replace procedure chosen_class(
2   ssname in student.sname%type
3 )as
4   ssno student.sno%type;
5   count_class int;
6   avg_grade int;
7   ttc int;
8 begin
9   select
10    student.sno,count(c2.cno),avg(grade),sum(ccredit)
11    into ssno,count_class,avg_grade,ttc
12    from student join sc s on student.sno = s.sno
13    join course c2 on c2.cno = s.cno
14    where student.sname=ssname
15    group by student.sno
16    ;
17 end;
18 call chosen_class('rr');

```

调用该存储过程，并根据指定的学生姓名输出与其相应的结果。

```

C##VANXQ> create or replace procedure chosen_class(
  ssname in student.sname%type
)as
  ssno student.sno%type;
  count_class int;
  avg_grade int;
  ttc int;
begin
  select student.sno,count(c2.cno),avg(grade),sum(ccredit)
  into ssno,count_class,avg_grade,ttc
  from student join sc s on student.sno = s.sno join course c2 on c2.cno = s.cno
  where student.sname=ssname
  group by student.sno
  ;
end;
[2022-04-22 17:48:25] 在 2 ms 内完成
C##VANXQ> call chosen_class('rr')
[2022-04-22 17:48:25] 在 1 ms 内完成

```


call chosen_class(ssname: 'rr');

	SNO	COUNT(C2.CNO)	AVG(GRADE)	SUM(CCREDIT)
1	20301168	3	96.333333333333333333333333333333...	11

(6) 写存储过程显示所有平均成绩小于给定班级号的平均成绩的学生
的学号、名字、平均成绩

```

1 create or replace procedure show_half(
2   cclassno class.classno%type
3 )as
4   csno student.sno%type;
5   csname student.sname%type;
6   cag sc.grade%type;
7 begin
8   select student.sno, student.sname, avg(grade)
9   into csno, csname, cag
10  from student join sc s on student.sno = s.sno
11  group by student.sno, student.sname
12  having avg(grade) < (select avg(grade) avg
13                      from sc
14                      join student s2 on
15                      s2.sno = sc.sno
16                      join class c3 on
17                      c3.classno = s2.classno
18                      where c3.classno = cclassno);
19
20  --游标（后知后觉），这个更加准确
21  create or replace procedure show_half_2
22    (ssclassno student.classno%type)
23  is
24    sssno student.sno%type;
25    sssname student.sname%type;

```

```

24      ssavggrade sc.grade%type;
25  begin
26      declare
27          cursor sscursor is
28      select s.sno,sname,avg(grade)
29      from student s,sc
30      where s.sno=sc.sno
31      group by s.sno,sname
32      having avg(grade)<(select avg(grade)
33                          from student join sc using(sno)
34                          where classno=ssclassno
35                          group by classno);
36      begin
37      open sscursor;
38      loop
39          fetch sscursor into sssno,sssname,ssavggrade;
40          exit when sscursor%notfound;
41
42          dbms_output.put_line('sno:' || sssno || ',name:' || sssname || ',avg(grade):' || ssavggrade);
43      end loop;
44      close sscursor;
45      end;
46  call show_lasthalf('RJ2003');

```

调用该存储过程，并根据指定的班级号输出与其相应的结果。

调用成功，但是这里显示的是整个年级的分数低于给定班级号所对应均分的同学的四列信息，不是没有限制在给定班级内，与题意相符


```

15 DBMS_OUTPUT.PUT_LINE('sname:' || c_2.sname);
16 DBMS_OUTPUT.PUT_LINE('ssex:' || c_2.ssex);
17
18 DBMS_OUTPUT.PUT_LINE('sbirthday:' || c_2.sbirthday);
19
20 DBMS_OUTPUT.PUT_LINE('classno:' || c_2.classno);
21 DBMS_OUTPUT.PUT_LINE('age:' || c_2.age);
22 end loop;
23 close c_1;END;END;
24 call
25 insert_student('99999999','www','男',to_date('2000/1
26 /1','yyyy/mm/dd'),'RJ2006');

```

```

C#VAXQ> create or replace trigger display_trigger
before insert on student
for each row
begin
declare
cursor c_1 is
select * from student;
c_2 student%rowtype;
begin
open c_1;
loop
fetch c_1 into c_2;
exit when c_1 %notfound;
DBMS_OUTPUT.PUT_LINE('sno:' || c_2.sno);
DBMS_OUTPUT.PUT_LINE('sname:' || c_2.sname);
DBMS_OUTPUT.PUT_LINE('ssex:' || c_2.ssex);
DBMS_OUTPUT.PUT_LINE('sbirthday:' || c_2.sbirthday);
DBMS_OUTPUT.PUT_LINE('classno:' || c_2.classno);
DBMS_OUTPUT.PUT_LINE('age:' || c_2.age);
end loop;
close c_1;END;END;
[2022-04-22 21:09:27] 在 24 ms 内完成
C#VAXQ> call insert_student('99999999','www','男',to_date('2000/1/1','yyyy/mm/dd'),'RJ2006')
[2022-04-22 21:09:36] 在 11 ms 内完成

```

(2) 创建一个触发器，当向学生表student中插入一新同学时能自动更新（增加1）class班级表中该生所在班级的总人数。

插入前

	CLASSNO	CLASSNAME	CLASSMAJOR	CLASSEPT	STUDENTNUMBER
1	RJ2001	软件2001	软件工程	软件开发	21
2	RJ2002	软件2002	软件工程	软件开发	23
3	RJ2003	软件2003	软件工程	数字媒体	22
4	RJ2004	软件2004	软件工程	软件开发	22
5	RJ2005	软件2005	软件工程	数字媒体	21
6	RJ2006	软件2006	软件工程	软件开发	21

```

1 create or replace trigger aa
2   after insert on student
3   for each row
4   begin
5       update class
6         set studentnumber=studentnumber+1
7         where classno=:new.classno;
8   end;
9 call
  insert_student('99999998','xxx','男',to_date('2000/1/
1', 'yyyy/mm/dd'),'RJ2006');

```

插入一个六班同学后，发现六班的人数加了一

call insert_student('99999998','xxx','男',to_date('2000/1/1','yyyy/mm/dd'),'RJ2006');					
select *from class;					
	CLASSNO	CLASSNAME	CLASSMAJOR	CLASSEPT	STUDENTNUMBER
1	RJ2001	软件2001	软件工程	软件开发	21
2	RJ2002	软件2002	软件工程	软件开发	23
3	RJ2003	软件2003	软件工程	数字媒体	22
4	RJ2004	软件2004	软件工程	软件开发	22
5	RJ2005	软件2005	软件工程	数字媒体	21
6	RJ2006	软件2006	软件工程	软件开发	22

(3) 创建一个触发器，当从学生表student中删除一个同学时能自动更新（减1）class班级表中该生所在班级的总人数。

删前

	CLASSNO	CLASSNAME	CLASSMAJOR	CLASSDEPT	STUDENTNUMBER
1	RJ2001	软件2001	软件工程	软件开发	21
2	RJ2002	软件2002	软件工程	软件开发	23
3	RJ2003	软件2003	软件工程	数字媒体	22
4	RJ2004	软件2004	软件工程	软件开发	22
5	RJ2005	软件2005	软件工程	数字媒体	21
6	RJ2006	软件2006	软件工程	软件开发	22

```

1  create or replace trigger bb
2      after delete on student
3      for each row
4      begin
5          update class
6              set studentnumber=studentnumber-1
7              where classno=:old.classno;
8      end;
9
10 delete
11 from student
12 where sno='99999998';
13 select *from class;

```

Txc 自动

✓

↺

■

☰

C##VANXQ

console

748

✓

delete

749

from student

750

where sno='99999998';

751

✓

select *from class;

×

1<

<

6 行

>

>1

↺

■

+

-

↺

↻

↑

Txc 自动

DDL

CSV

↓

↕

🔍

⚙️

CLASSNO

CLASSNAME

CLASSMAJOR

...

STUDENTNUMBER

3

RJ2003

软件2003

软件工程

数字媒体

22

4

RJ2004

软件2004

软件工程

软件开发

22

5

RJ2005

软件2005

软件工程

数字媒体

21

6

RJ2006

软件2006

软件工程

软件开发

21

752

753

(4) 创建一个触发器，当将学生表student中某一个同学从一个班级改为另一个班级时，能自动更新class班级表中该生所在原来班级的总人数（减1）和新班级的总人数（增加1）。

变更前

	CLASSNO	CLASSNAME	CLASSMAJOR	CLASSDEPT	STUDENTNUMBER
1	RJ2001	软件2001	软件工程	软件开发	21
2	RJ2002	软件2002	软件工程	软件开发	22
3	RJ2003	软件2003	软件工程	数字媒体	22
4	RJ2004	软件2004	软件工程	软件开发	22
5	RJ2005	软件2005	软件工程	数字媒体	22
6	RJ2006	软件2006	软件工程	软件开发	21

```
1 create or replace trigger cc
2   after update on student
3   for each row
4   begin
5       update class
6         set studentnumber=studentnumber-1
7       where classno=:old.classno;
8       update class
9         set studentnumber=studentnumber+1
10      where classno=:new.classno;
11   end;
12 UPDATE STUDENT
13 SET CLASSNO = 'RJ2006'--原来是2005
14 WHERE SNO LIKE '20301150' ;
15 select *from class;
```

65	✓	UPDATE STUDENT
66		SET CLASSNO = 'RJ2006'--原来是2005
67		WHERE SNO LIKE '20301150' ;
68	✓	select *from class;

	CLASSNO	CLASSNAME	CLASSMAJOR	...	STUDENTNUMBER
1	RJ2001	软件2001	软件工程	软件开发	21
2	RJ2002	软件2002	软件工程	软件开发	22
3	RJ2003	软件2003	软件工程	数字媒体	22
4	RJ2004	软件2004	软件工程	软件开发	22
5	RJ2005	软件2005	软件工程	数字媒体	21
6	RJ2006	软件2006	软件工程	软件开发	22

(5) 建一个触发器，当往SC表中插入一个在STUDENT 表中不存在的学号SNO时，就往STUDENT表中插入该学号的记录，对于有约束的字段如姓名、性别、班级号等随机产生有效值。

```

1 create or replace trigger trig_5
2   after insert on sc
3   for each row
4   declare
5       cs integer;
6   begin
7       select count(*) into cs from student where
:new.sno=student.sno;
8       if cs=0 then
9           insert into student values
(:new.sno,'aaa','女',to_date('2000/1/1','yyyy/mm/dd'
),default,default,20301174,default,default);
10      end if;
11  end;

```

insert into sc(sno)values ('11111111');
select *from student;

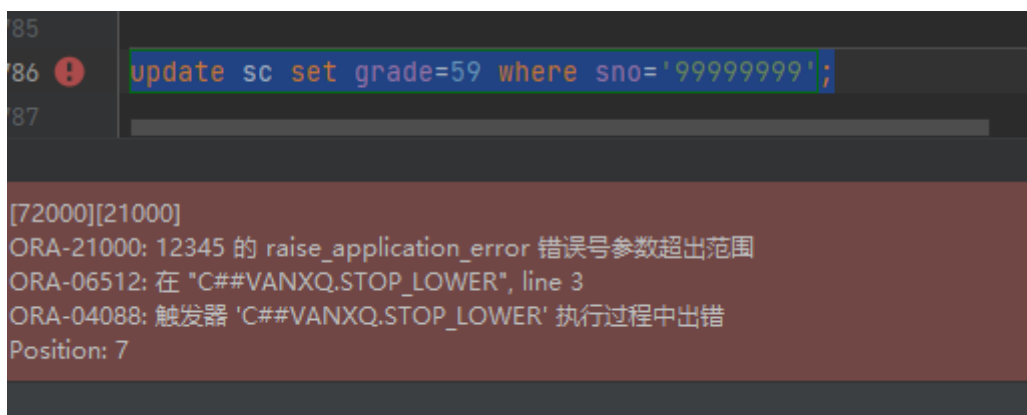
	SNO	SNAME	SSEX	SBIRTHDAY	CLASSNO	TOTALCREDIT
28	14	n	男	<null>	<null>	0
29	15	o	男	<null>	<null>	0
30	16	p	男	<null>	<null>	0
31	99999999	www	男	2000-01-01	RJ2006	0
32	20301150	谁啊	男	2000-01-01	RJ2006	0
33	11111111	aaa	女	2000-01-01	<null>	0
34	20301174	万兴全	男	2001-03-20	RJ2006	0

(6) 写一个触发器阻止将学生成绩降低


```

1 create or replace trigger stop_lower
2   before update of grade on sc
3   for each row
4   begin
5       if:new.grade<:old.grade
6       then raise_application_error('12345','球
球了，不能再低了');
7       end if;
8   end;
9
10 update sc set grade=59 where sno='99999999';

```



(7) 在sc表上创建触发器，只要有人选修的课程超过3门，就中断操作并提示警告

20301168已选三门，执行代码

	SNO	CNO	GRADE
1	20301167	800003	63
2	20301168	810013	94
3	20301168	800001	97
4	20301168	800002	98

```

1 create or replace trigger trig_7
2   after insert on sc
3   for each row
4   declare
5       PRAGMA autonomous_transaction ;
6       cursor cc is select count(cno)from sc group
by sno;

```

```

7      ca int;
8      begin
9      open cc;
10     loop fetch cc into ca;
11     exit when cc%notfound;
12     if ca>=3 then
13         raise_application_error(54321,'太卷啦太卷
14     end if;
15 end loop;
16 end;
17 insert into sc values (20301168,'800003',100);

```

```

813     if ca>=3 then
814         raise_application_error(54321,'太卷啦太卷啦');
815     end if;
816 end loop;
817 end;
818 ! insert into sc values (20301168,'800003',100);
819
820
821

```

[72000][21000]
ORA-21000: 54321 的 raise_application_error 错误号参数超出范围
ORA-06512: 在 "C##VANXQ.TRIG_7", line 10
ORA-04088: 触发器 'C##VANXQ.TRIG_7' 执行过程中出错
Position: 12

(8) 创建一个触发器，当往SC表插入选修的课程时自动将该课程的学分累加到STUDENT表的该生的总学分（TOTALCREDIT）中，当从SC表中退课时自动从STUDENT表的该生的总学分中减去该课程的学分。

这里选一节没有人上过的好区分的课，名字叫“不上课”，一学分

	CNO	CNAME	CCREDIT
1	800001	计算机基础	4
2	800002	程序设计语言	4
3	800003	数据结构	4
4	810011	数据库系统	4
5	810013	计算机网络	3
6	810015	微机原理与应用	4
7	1	不上课	1
8	0	体育课	4

```

1  create or replace trigger trig_8
2  after insert or delete on sc
3  for each row
4  begin
5      if inserting then
6          update student
7              set TOTALCREDIT=TOTALCREDIT+(select
ccredit
8                                      from course
9                                      where
course.cno=:new.cno)
10             where sno=:new.sno;
11     else
12         update student
13             set TOTALCREDIT=TOTALCREDIT-(select
ccredit
14                                     from course
15                                     where
course.cno=:old.cno)
16             where sno=:old.sno;
17
18     end if;
19 end;
```

查看当前学分

	SNO	SNAME	SSEX	SBIRTHDAY	CLASSNO	TOTALCREDIT	CreateTime
19	5	e	男	<null>	<null>	0	20301174
20	6	f	男	<null>	<null>	0	20301174
21	7	g	男	<null>	<null>	0	20301174
22	8	h	男	<null>	<null>	0	20301174
23	9	i	男	<null>	<null>	0	20301174
24	10	j	男	<null>	<null>	0	20301174
25	11	k	男	<null>	<null>	0	20301174
26	12	l	男	<null>	<null>	0	20301174
27	13	m	男	<null>	<null>	0	20301174
28	14	n	男	<null>	<null>	0	20301174
29	15	o	男	<null>	<null>	0	20301174
30	16	p	男	<null>	<null>	0	20301174
31	99999999	www	男	2000-01-01	RJ2006	0	20301174
32	20301150	谁啊	男	2000-01-01	RJ2006	0	20301174
33	11111111	aaa	女	2000-01-01	<null>	0	20301174
34	20301174	万兴全	男	2001-03-20	RJ2006	0	20301174

	SNO	CNO	GRADE
4	20301100	800002	70
5	20301167	800001	81
6	20300100	800001	61
7	20300020	800002	82
8	20300020	800002	98
9	20300012	800003	<null>
10	1	800001	100
11	20300020	800003	94
12	20300030	800003	80
13	20300100	810011	69
14	20300148	810011	60
15	20300150	810011	92
16	20300160	810011	73
17	20301167	810011	80
18	11111111	<null>	<null>
19	99999999	810011	99

增加选课拿到一学分

	SNO	SNAME	SSEX	SBIRTHDAY	CLASSNO	TOTALCREDIT	
19	5	e	男	<null>	<null>	0	20301174
20	6	f	男	<null>	<null>	0	20301174
21	7	g	男	<null>	<null>	0	20301174
22	8	h	男	<null>	<null>	0	20301174
23	9	i	男	<null>	<null>	0	20301174
24	10	j	男	<null>	<null>	0	20301174
25	11	k	男	<null>	<null>	0	20301174
26	12	l	男	<null>	<null>	0	20301174
27	13	m	男	<null>	<null>	0	20301174
28	14	n	男	<null>	<null>	0	20301174
29	15	o	男	<null>	<null>	0	20301174
30	16	p	男	<null>	<null>	0	20301174
31	99999999	www	男	2000-01-01	RJ2006	0	20301174
32	20301150	谁啊	男	2000-01-01	RJ2006	0	20301174
33	11111111	aaa	女	2000-01-01	<null>	0	20301174
34	20301174	万兴全	男	2001-03-20	RJ2006	1	20301174

撤销这门课

```
49
50 ✓ delete from sc where cno=1;
```

学分又变回0

	SNO	SNAME	SSEX	SBIRTHDAY	CLASSNO	TOTALCREDIT	CreateTime
19	5	e	男	<null>	<null>	0	20301174
20	6	f	男	<null>	<null>	0	20301174
21	7	g	男	<null>	<null>	0	20301174
22	8	h	男	<null>	<null>	0	20301174
23	9	i	男	<null>	<null>	0	20301174
24	10	j	男	<null>	<null>	0	20301174
25	11	k	男	<null>	<null>	0	20301174
26	12	l	男	<null>	<null>	0	20301174
27	13	m	男	<null>	<null>	0	20301174
28	14	n	男	<null>	<null>	0	20301174
29	15	o	男	<null>	<null>	0	20301174
30	16	p	男	<null>	<null>	0	20301174
31	99999999	www	男	2000-01-01	RJ2006	0	20301174
32	20301150	谁啊	男	2000-01-01	RJ2006	0	20301174
33	11111111	aaa	女	2000-01-01	<null>	0	20301174
34	20301174	万兴全	男	2001-03-20	RJ2006	0	20301174

源码

```
1  --第五次实验报告
2  --(一)
3  --1
4
5  create or replace procedure insert_student(
6  ssno in student.sno%type,
7  ssname in student.sname%type,
8  sssex in student.ssex%type,
9  ssbirthday in date,
10 sclassno in student.classno%type
11 )
12 is
13 begin
14     insert into
15     student(sno,sname,ssex,sbirthday,classno)values
16     (ssno,ssname,sssex,ssbirthday,sclassno);
17 end;
18
19 begin
20     insert_student('20301174','万兴
21     全','男',to_date('2001/3/20','yyyy/mm/dd'),'RJ2006'
22     );
23 end;
24
25 select *from student;
26
27
28 --2
29
30 create or replace procedure insert_course(
31 ccno course.cno%type,
32 ccname course.cname%type,
33 cccredit int:=4
34 )
35 is
36
37     begin
38
39         insert into course (cno, cname, ccredit)
40         values (ccno,ccname,cccredit);
```

```
32     end;
33
34 begin
35     insert_course(000000,'体育课');
36 end;
37 call
38     insert_course(000001,'不上课',1);
39
40 select * from course;
41
42
43 --3
44 create or replace procedure query_student (
45     ssno in student.sno%type
46 )
47 as
48 ssname student.sname%type;
49 sssex   student.ssex%type;
50 ssbirthday student.sbirthday%type;
51 sclassno student.classno%type;
52 begin
53     select sname,sssex,sbirthday,classno
54     into ssname,sssex,ssbirthday,sclassno
55     from student
56     where sno=ssno;
57
58 END;
59 call INSERT_STUDENT('20301150','谁
60 啊','男',to_date('2000/1/1','yyyy/mm/dd'),'RJ2002')
61 ;
62
63 begin    query_student('20301150');
64 END;
65
66
67
```

```

68  --4
69  create or replace procedure at_least_80
70  as
71  ssname student.sname%type;
72  sssex student.ssex%type;
73  ssage student.age%type;
74  agrade sc.grade%type;
75
76  begin
77      select sname,sssex,age,avg(grade)
78      into ssname,sssex,ssage,agrade
79      from student,sc
80      where sc.sno=student.sno
81      having avg(grade)>80
82      group by sname, ssex, age;
83  end;
84
85  call at_least_80();
86      select sname,sssex,age,avg(grade)
87      from student,sc
88      where sc.sno=student.sno
89      having avg(grade)>80
90      group by sname, ssex, age;
91  --5
92  create or replace procedure cs(
93  asname student.sname%type,
94  ssno out student.sno%type,
95  ssname out student.sname%type,
96  ca out int,
97  ag out int,
98  ttc out student.totalcredit%type
99
100 )is
101 cursor csc is select
102 1 sc.sno,sname,count(cno),avg(grade),"Totalcredit"
103 into ssno,ssname,ca,ag,ttc
104 from student join sc s on student.sno = s.sno
105 and student.sno in(select sno from student

```



```

10         where not
5 exists(select *
10             from (select cno
10                 from sc
10                 where sno=
8 (select sno
10
9 from student
11
0 where sname=asname)))t
11             where not exists(
11                 select *from
2 sc where sc.cno=t.cno and sc.sno=student.sno
11                 )))
13 group by sc.cno,sname,"Totalcredit";
14     begin open csc;
15     loop fetch csc into ssno,ssname,ca,ttc;
16     exit when csc%notfound;
17
18     end loop;
19
0 DBMS_OUTPUT.PUT_LINE(ssno||','||ssname||','||ag);
12     close csc;
12     end cs;
12
13     declare
14 ssno student.sno%type;
18 ssname student.sname%type;
10 ca int;
12 ag int;
18 ttc student.totalcredit%type;
19     begin
10
13         cs('万兴全',ssno,ssname,ca,ag,ttc);
13         end;
13
14 create or replace procedure chosen_class(
15 ssname in student.sname%type

```

```

18 )as
17     ssno student.sno%type;
18     count_class int;
19     avg_grade int;
10     ttc int;
14 begin
12     select
3 student.sno,count(c2.cno),avg(grade),sum(ccredit)
14     into ssno,count_class,avg_grade,ttc
14     from student join sc s on student.sno = s.sno
5 join course c2 on c2.cno = s.cno
14     where student.sname=ssname
16     group by student.sno
17     ;
18 end;
19 call chosen_class('rr');
10     ;
11
12
13 --6
14 create or replace procedure show_lasthalf(
15 cclassno class.classno%type
16 )as
17     csno student.sno%type;
18     csname student.sname%type;
10     cag sc.grade%type;
10 begin
16     select student.sno,student.sname,avg(grade)
10     into csno,csname,cag
18     from student join sc s on student.sno = s.sno
16     group by student.sno, student.sname
16     having avg(grade)< (select avg(grade) avg
16                             from sc
10                             join student s2 on
8 s2.sno = sc.sno
16                             join class c3 on
9 c3.classno = s2.classno
17     where c3.classno=cclassno);

```

```

10
11     end;
12
13     --游标（后知后觉）
14 create or replace procedure show_lasthalf
15     (ssclassno student.classno%type)
16 is
17     sssno student.sno%type;
18     sssname student.sname%type;
19     ssavggrade sc.grade%type;
20 begin
21     declare
22         cursor sscursor is
23         select s.sno,sname,avg(grade)
24         from student s,sc
25         where s.sno=sc.sno
26         group by s.sno,sname
27         having avg(grade)<(select avg(grade)
28                             from student join sc using(sno)
29                             where classno=ssclassno
30                             group by classno);
31
32         begin
33         open sscursor;
34         loop
35             fetch sscursor into sssno,sssname,ssavggrade;
36             exit when sscursor%notfound;
37
38             7 dbms_output.put_line('sno:'||sssno||',name:'||sssna
39             me||',avg(grade):'||ssavggrade);
40         end loop;
41         close sscursor;
42         end;
43 end;
44
45 call show_lasthalf('RJ2003');
46
47
48

```

```
20
20
20
20
21
22
23 --(二)
24 --1
25
26 create or replace trigger display_trigger
27     before insert on student
28     for each row
29     begin
30     declare
31         cursor c_1 is
32             select * from student;
33             c_2 student%rowtype;
34     begin
35         open c_1;
36         loop
37             fetch c_1 into c_2;
38             exit when c_1 %notfound;
39             DBMS_OUTPUT.PUT_LINE('sno: ' || c_2.sno);
40
41 1 DBMS_OUTPUT.PUT_LINE('sname: ' || c_2.sname);
42
43 2 DBMS_OUTPUT.PUT_LINE('ssex: ' || c_2.ssex);
44
45 3 DBMS_OUTPUT.PUT_LINE('sbirthday: ' || c_2.sbirthday);
46
47 4 DBMS_OUTPUT.PUT_LINE('classno: ' || c_2.classno);
48     DBMS_OUTPUT.PUT_LINE('age: ' || c_2.age);
49     end loop;
50     close c_1;END;END;
51
52 call
53
54 8 insert_student('99999999','www','男',to_date('2000/
1/1','yyyy/mm/dd'),'RJ2006');
```

```

23
29
20 --2
24 create or replace trigger aa
22     after insert on student
23     for each row
24     begin
25         update class
26             set studentnumber=studentnumber+1
27             where classno=:new.classno;
28     end;
29 call
0 insert_student('99999998','xxx','男',to_date('2000/
1/1','yyyy/mm/dd'),'RJ2006');
25 select *from class;
21
28 --3
23 create or replace trigger bb
24     after delete on student
25     for each row
26     begin
27         update class
28             set studentnumber=studentnumber-1
29             where classno=:old.classno;
30     end;
26
28 delete
28 from student
29 where sno='99999998';
26 select *from class;
26
28 --4
28 create or replace trigger cc
29     after update on student
30     for each row
31     begin
32         update class
33             set studentnumber=studentnumber-1

```

```

24         where classno=:old.classno;
25         update class
26             set studentnumber=studentnumber+1
27             where classno=:new.classno;
28     end;
29 UPDATE STUDENT
30 SET CLASSNO = 'RJ2006'--原来是2005
31 WHERE SNO LIKE '20301150' ;
32 select *from class;
33
34
35
36
37 --5
38 create or replace trigger trig_5
39     after insert on sc
40     for each row
41     declare
42         cs integer;
43     begin
44         select count(*) into cs from student where
45             :new.sno=student.sno;
46         if cs=0 then
47             insert into student values
48                 (:new.sno,'aaa','女',to_date('2000/1/1','yyyy/mm/dd
49                 '),default,default,20301174,default,default);
50         end if;
51     end;
52
53
54 insert into sc(sno)values ('11111111');
55 select *from student;
56
57
58 --6
59 create or replace trigger stop_lower
60     before update of grade on sc
61     for each row
62     begin
63         if:new.grade<:old.grade

```

```

30         then
0    raise_application_error('12345','球球了，不能再低
    了');
31         end if;
31     end;
32
33 update sc set grade=59 where sno='99999999';
34
35 --7
36 create or replace trigger trig_7
37     after insert on sc
38     for each row
39     declare
40         PRAGMA autonomous_transaction ;
32         cursor cc is select count(cno)from sc group
2    by sno;
32         ca int;
33         begin
34             open cc;
38             loop fetch cc into ca;
38             exit when cc%notfound;
32             if ca>=3 then
38                 raise_application_error(54321,'太卷啦太
9    卷啦');
33             end if;
38         end loop;
31         end;
32 insert into sc values (20301168,'800003',100);
33
34
35
36 --8
37 create or replace trigger TRIG_8
38     after insert or delete
39     on SC
40     for each row
34 begin
34     if inserting then

```

```

33         update student
34             set TOTALCREDIT=TOTALCREDIT+(select
5 ccredit
34                                     from
6 course
34                                     where
7 course.cno=:new.cno)
34             where sno=:new.sno;
38     else
39         update student
40             set TOTALCREDIT=TOTALCREDIT-(select
1 ccredit
35                                     from
2 course
35                                     where
3 course.cno=:old.cno)
35             where sno=:old.sno;
34
35     end if;
36 end;
37
38 select ccredit from course where cno=1;
39
40 insert into sc
41 values ('20301174',1,90);
42 select "Totalcredit" from student where
3 sno=20301174;
36
43 delete from sc where cno=1;
44 --麻了，真的麻了

```