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**

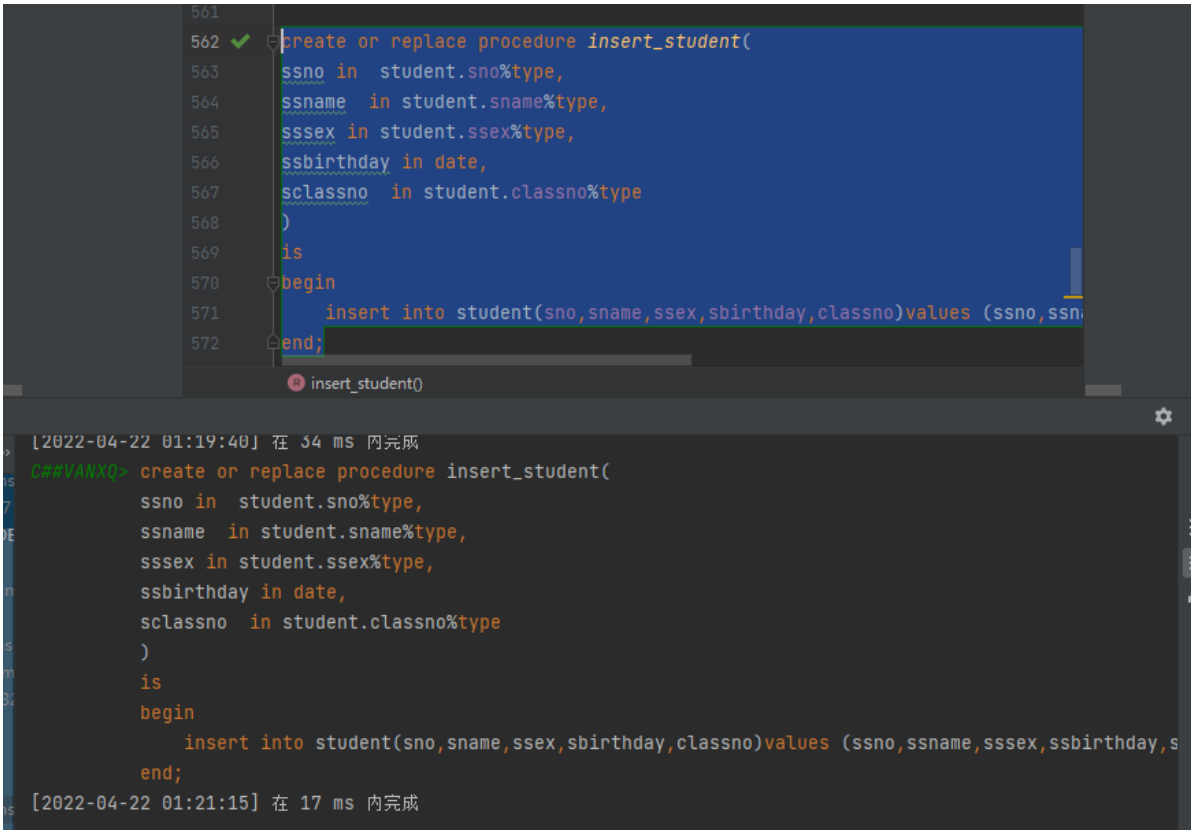
---

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。

```
create 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,sclassno);
end;
```



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
7	20300075	王娜娜	女	2003-09-23	RJ2003	
8	20300088	秦键	男	2004-03-01	RJ2003	
9	20300100	田邦仪	女	2003-02-26	RJ2004	
10	20300148	赵心砚	男	2002-04-25	RJ2005	
11	20300150	杨青	女	2004-11-15	RJ2005	
12	20300160	杨玲玲	女	2002-12-12	RJ2006	
13	20301168	rr	男	<null>	<null>	<nu
14	20301167	qfs	男	<null>	<null>	<nu
15	1	a	男	<null>	<null>	
16	2	b	男	<null>	<null>	
17	3	c	男	<null>	<null>	
18	4	d	男	<null>	<null>	
19	5	e	男	<null>	<null>	
20	6	f	男	<null>	<null>	
21	7	g	男	<null>	<null>	
22	8	h	男	<null>	<null>	
23	9	i	男	<null>	<null>	
24	10	j	男	<null>	<null>	
25	11	k	男	<null>	<null>	
26	12	l	男	<null>	<null>	
27	13	m	男	<null>	<null>	
28	14	n	男	<null>	<null>	
29	15	o	男	<null>	<null>	
30	16	p	男	<null>	<null>	

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

```
begin
    insert_student('20301174','万兴全','男',to_date('2001/3/20','yyyy/mm/dd
end;
select *from student;
```

31 行 | Tx 自动 | DDL | CSV

	SNO	SNAME	SSEX	SBIRTHDAY	CLASSNO
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。

```
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;
```

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

```
begin
    insert_course(000000,'体育课');
end;
begin
    insert_course(000001,'不上课',1);
end;
select * from course;
```

（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。

```
create or replace procedure query_student (
 ssno in student.sno%type
 )
```

```
as
ssname student.sname%type;
sssex  student.ssex%type;
ssbirthday student.sbirthday%type;
sclassno student.classno%type;
begin
select sname,ssex,sbirthday,classno
into ssname,sssex,ssbirthday,sclassno
from student
where sno=ssno;
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。

```
call INSERT_STUDENT('20301150','谁
啊','男',to_date('2000/1/1','yyyy/mm/dd'),'RJ2002');

begin   query_student('20301150');
END;
```

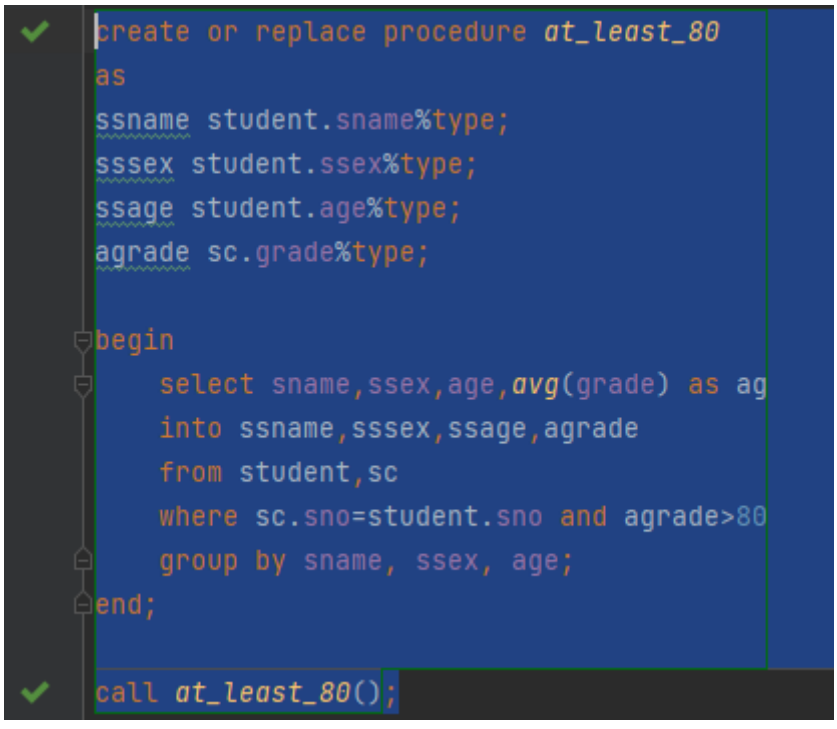
```
C##VANXQ> call INSERT_STUDENT('20301150','谁啊','男',to_date('2000/1/1','yyyy/mm/dd'),'RJ2002')
[2022-04-22 17:14:51] 在 17 ms 内完成
C##VANXQ> begin   query_student('20301150');
          END;
[2022-04-22 17:14:51] 在 1 ms 内完成
```

（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。

```
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();
```

调用该存储过程，并输出相应的结果。



5 行 CSV

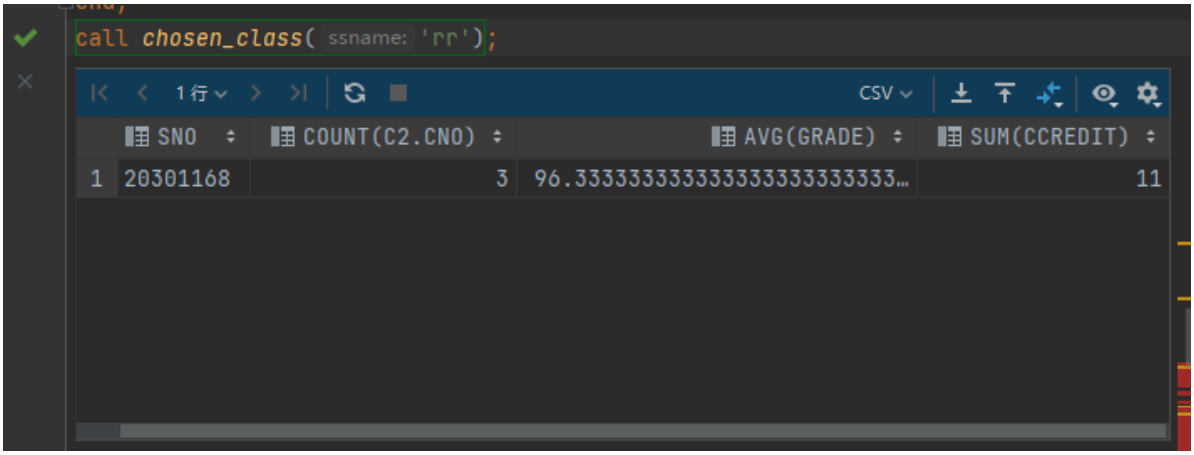
	SNAME	SSEX	AGE	AVG(GRADE)
1	杨敏	女	18	91.33333333333333333333333333333333…
2	赵鸿泽	男	18	100
3	杨青	女	18	92
4	rr	男	18	96.33333333333333333333333333333333…
5	a	男	18	100

（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的学号、姓名、选课数、平均成绩、总学分

```
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;
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
```



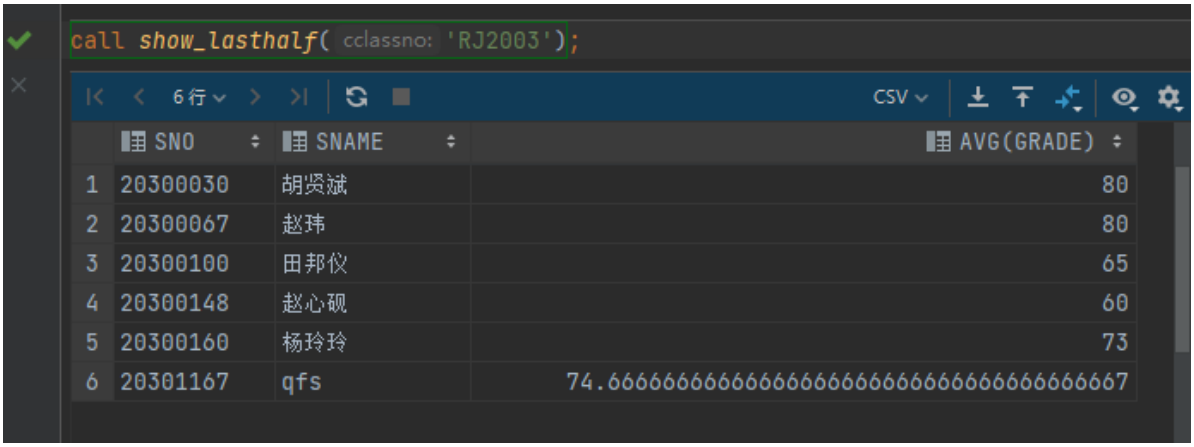
（6） 写存储过程显示所有平均成绩小于给定班级号的平均成绩的学生的学号、名字、平均成绩

```
create or replace procedure show_lasthalf(
cclassno class.classno%type
)as
    csno student.sno%type;
    csname student.sname%type;
    cag sc.grade%type;
begin
    select student.sno,student.sname,avg(grade)
    into csno,csname,cag
    from student join sc s on student.sno = s.sno
    group by student.sno, student.sname
    having avg(grade)< (select avg(grade) avg
                        from sc
                            join student s2 on
s2.sno = sc.sno
                            join class c3 on
c3.classno = s2.classno
                        where c3.classno=cclassno);

--游标（后知后觉），这个更加准确
create or replace procedure show_lasthalf_2
    (ssclassno student.classno%type)
is
    sssno student.sno%type;
    sssname student.sname%type;
```

```
    ssavggrade sc.grade%type;
begin
    declare
        cursor sscursor is
 select s.sno,sname,avg(grade)
 from student s,sc
 where s.sno=sc.sno
 group by s.sno,sname
 having avg(grade)<(select avg(grade)
                from student join sc using(sno)
                where classno=ssclassno
                group by classno);
        begin
 open sscursor;
 loop
     fetch sscursor into sssno,sssname,ssavggrade;
     exit when sscursor%notfound;

dbms_output.put_line('sno:'||sssno||',name:'||sssnam
e||',avg(grade):'||ssavggrade);
 end loop;
 close sscursor;
        end;
end;
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



```
call show_lasthalf( cclassno: 'RJ2003');
```

6行 CSV

	SNO	SNAME	AVG(GRADE)
1	20300030	胡贤斌	80
2	20300067	赵玮	80
3	20300100	田邦仪	65
4	20300148	赵心砚	60
5	20300160	杨玲玲	73
6	20301167	qfs	74.66666666666666666666666666666666666667

2. 触发器（40分，每题5分）

```
--做到这里才后知后觉要用到游标，之前的有的可以用的但是我没用。

--游标很像指针，更像迭代器。
```

在Oracle下用PL/SQL语句定义触发器

（1） 创建一个当向学生表student中插入一新同学时能自动列出全部同学信息的触发器display_trigger。执行存储过程insert_student，向学生表中插入一个新同学，看触发器display_trigger是否被触发。

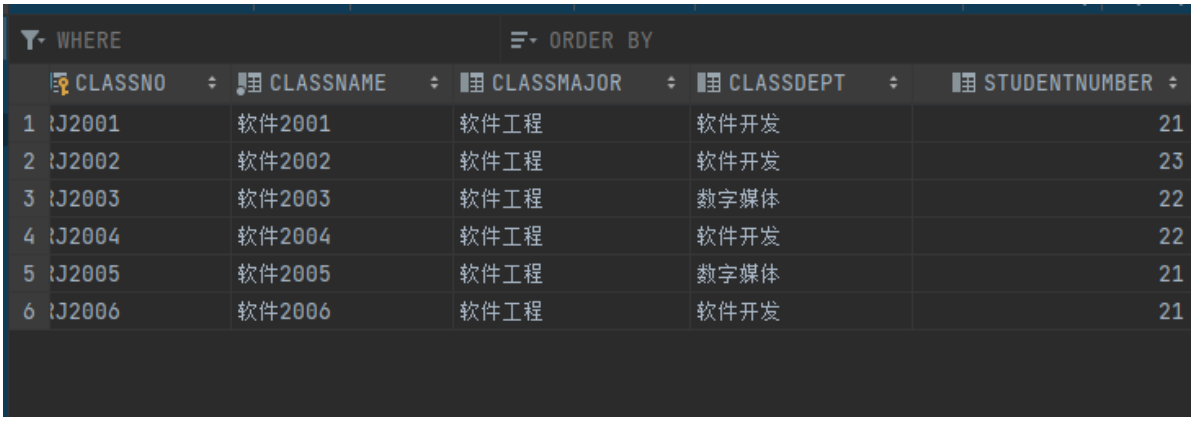
```
create or replace trigger display_trigger
    before insert on student
    for each row
    begin
    declare
        cursor c_1 is
        select * from student;
        c_2 student%rowtype;
        begin
        open c_1;
        loop
            fetch c_1 into c_2;
            exit when c_1 %notfound;
            DBMS_OUTPUT.PUT_LINE('sno:'||c_2.sno);
```

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

```
C##VANXQ> 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##VANXQ> 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	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	21

```
create or replace trigger aa
    after insert on student
    for each row
    begin
        update class
            set studentnumber=studentnumber+1
        where classno=:new.classno;
    end;
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	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

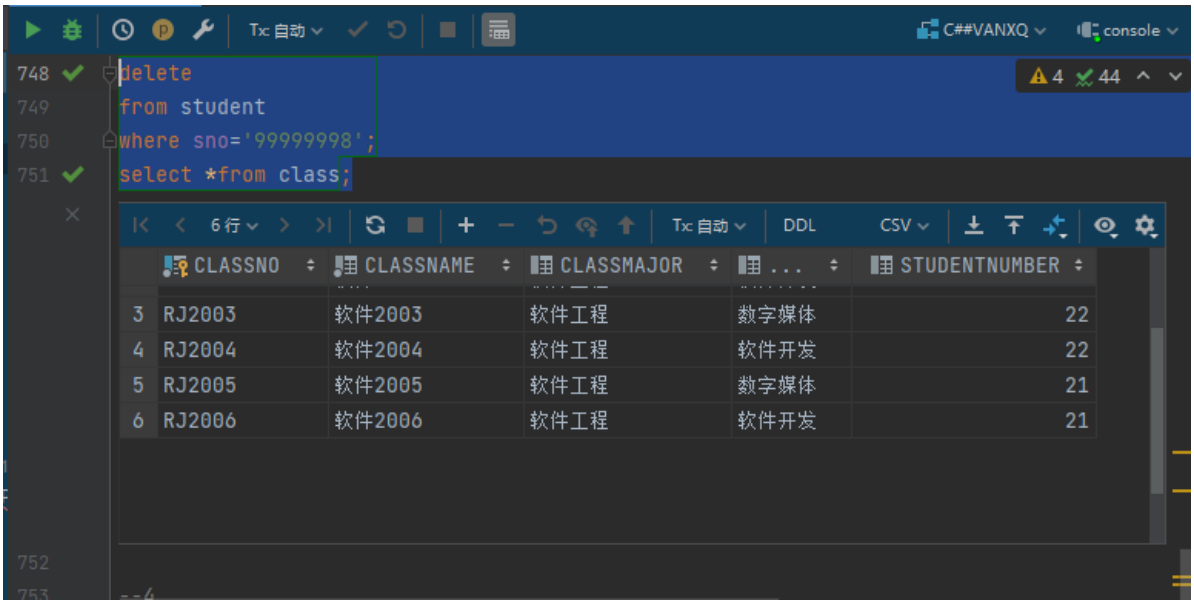
（3） 创建一个触发器，当从学生表student中删除一个同学时能自动更新（减1）class班级表中该生所在班级的总人数。

删前

WHERE ORDER BY

	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

```
create or replace trigger bb
    after delete on student
    for each row
    begin
        update class
            set studentnumber=studentnumber-1
        where classno=:old.classno;
    end;

delete
from student
where sno='99999998';
select *from class;
```



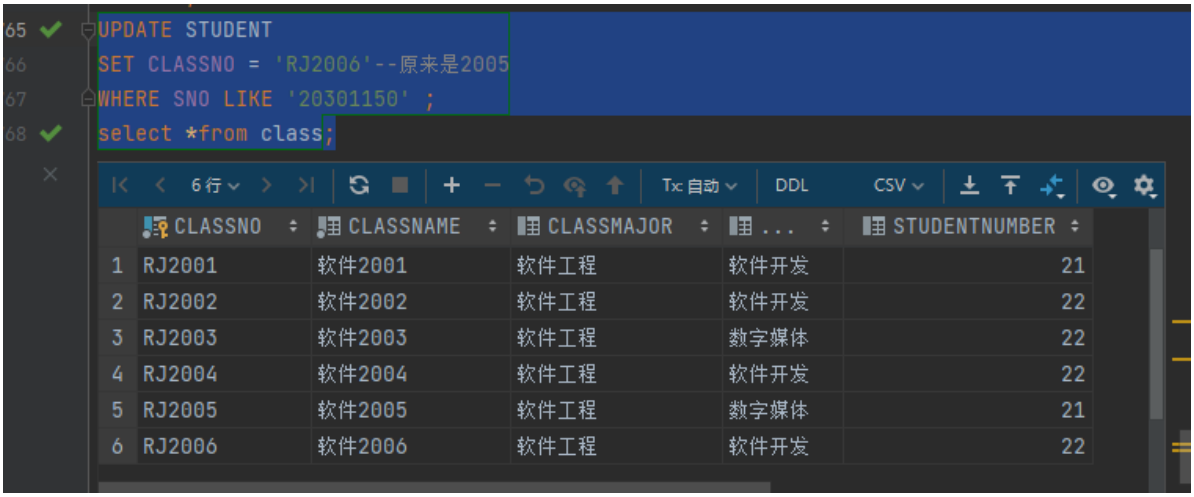
（4） 创建一个触发器，当将学生表student中某一个同学从一个班级改为另一个班级时，能自动更新class班级表中该生所在原来班级的总人数（减1）和新班级的总人数（增加1）。

变更前

WHERE ORDER BY

	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

```
create or replace trigger cc
    after update on student
    for each row
    begin
        update class
            set studentnumber=studentnumber-1
        where classno=:old.classno;
        update class
            set studentnumber=studentnumber+1
        where classno=:new.classno;
    end;
UPDATE STUDENT
SET CLASSNO = 'RJ2006'--原来是2005
WHERE SNO LIKE '20301150' ;
select *from class;
```



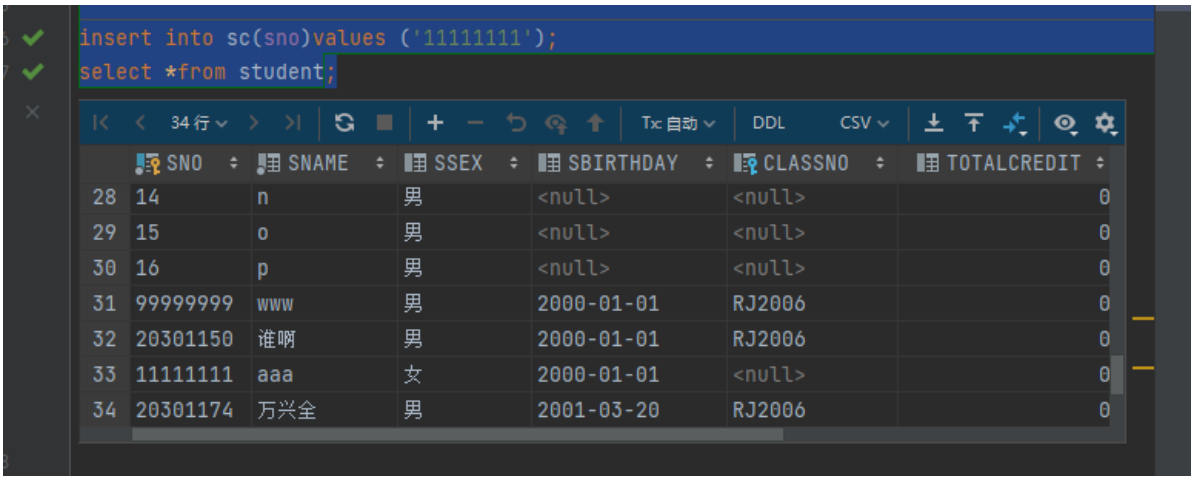
```
UPDATE STUDENT
SET CLASSNO = 'RJ2006'--原来是2005
WHERE SNO LIKE '20301150' ;
select *from class;
```

6 行 Tx 自动 DDL CSV

	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。

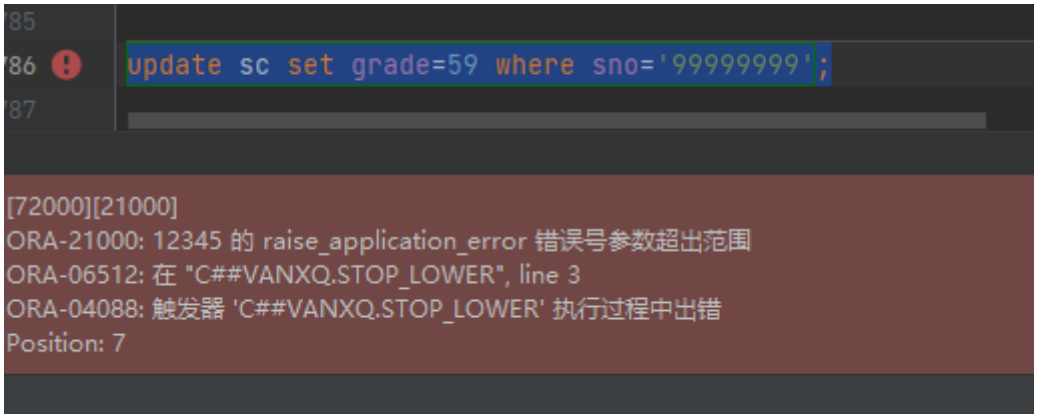
```
create or replace trigger trig_5
    after insert on sc
    for each row
    declare
        cs integer;
        begin
        select count(*) into cs from student where
:new.sno=student.sno;
        if cs=0 then
            insert into student  values
(:new.sno,'aaa','女',to_date('2000/1/1','yyyy/mm/dd'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



	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

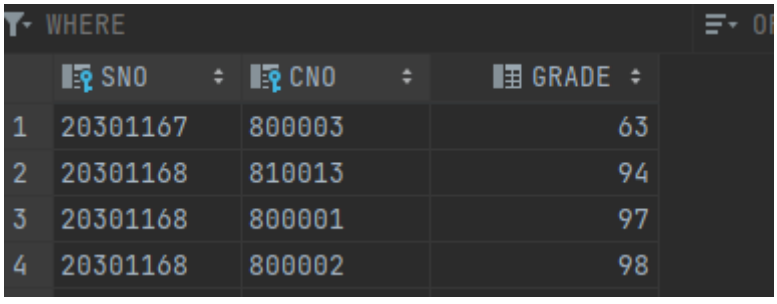
（6） 写一个触发器阻止将学生成绩降低

```
create or replace trigger stop_lower
    before update of grade on sc
    for each row
    begin
        if:new.grade<:old.grade
            then raise_application_error('12345','球球了，不能再低了');
        end if;
    end;

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




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

```
        ca int;
        begin
        open cc;
        loop fetch cc into ca;
        exit when cc%notfound;
        if ca>=3 then
            raise_application_error(54321,'太卷啦太卷
啦');
        end if;
    end loop;
        end;
insert into sc values (20301168,'800003',100);
```

（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。

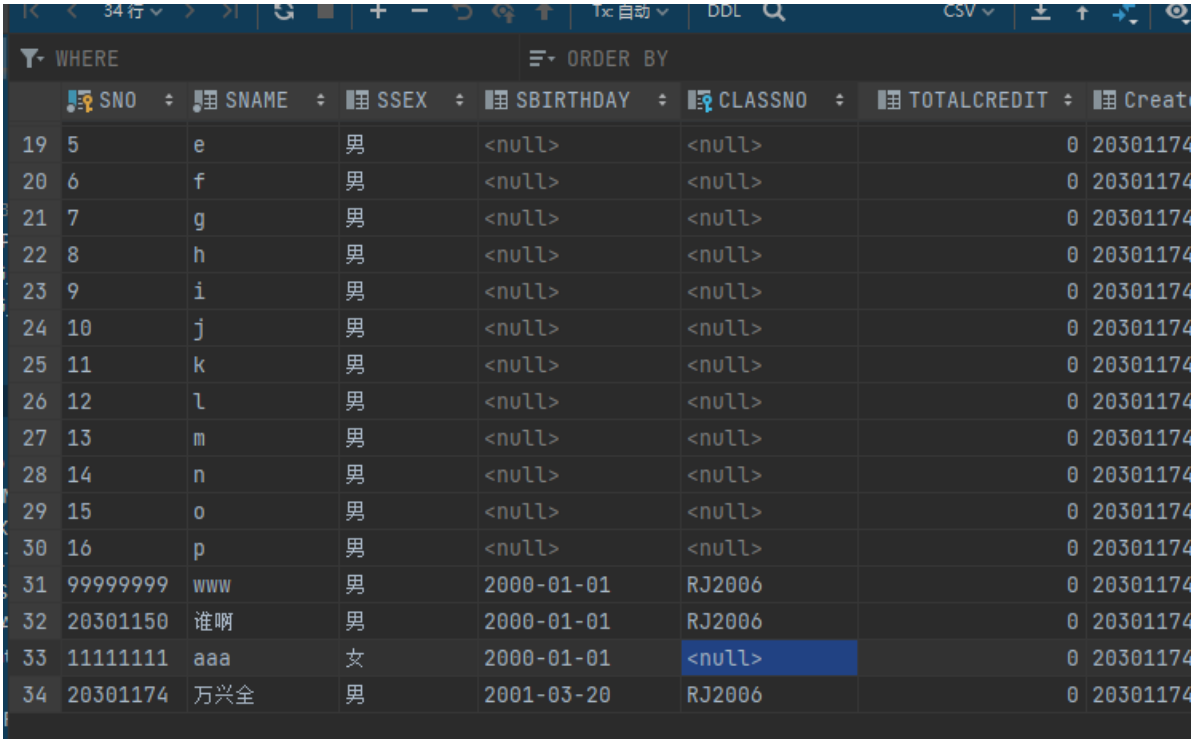
这里选一节没有人上过的好区分的课，名字叫“不上课”，一学分

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
7	1	不上课	1
8	0	体育课	4

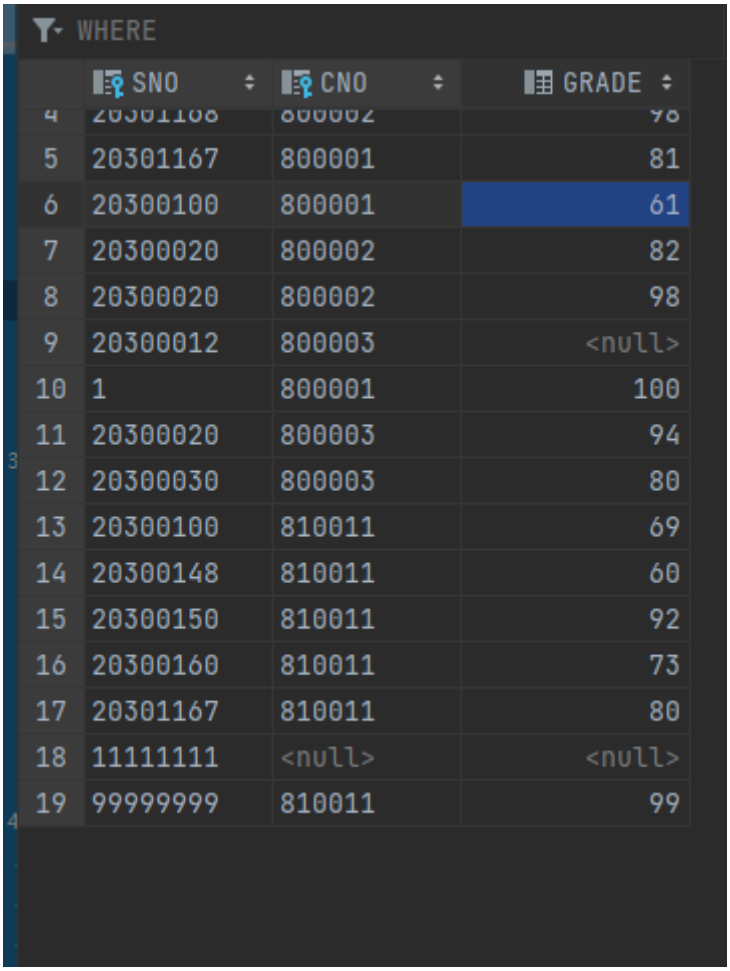
```
create or replace trigger trig_8
after insert or delete on sc
for each row
begin
    if inserting then
        update student
            set TOTALCREDIT=TOTALCREDIT+(select
ccredit
                                        from course
                                        where
course.cno=:new.cno)
            where sno=:new.sno;
    else
        update student
            set TOTALCREDIT=TOTALCREDIT-(select
ccredit
                                        from course
                                        where
course.cno=:old.cno)
            where sno=:old.sno;

    end if;
end;
```

查看当前学分



WHERE ORDER BY

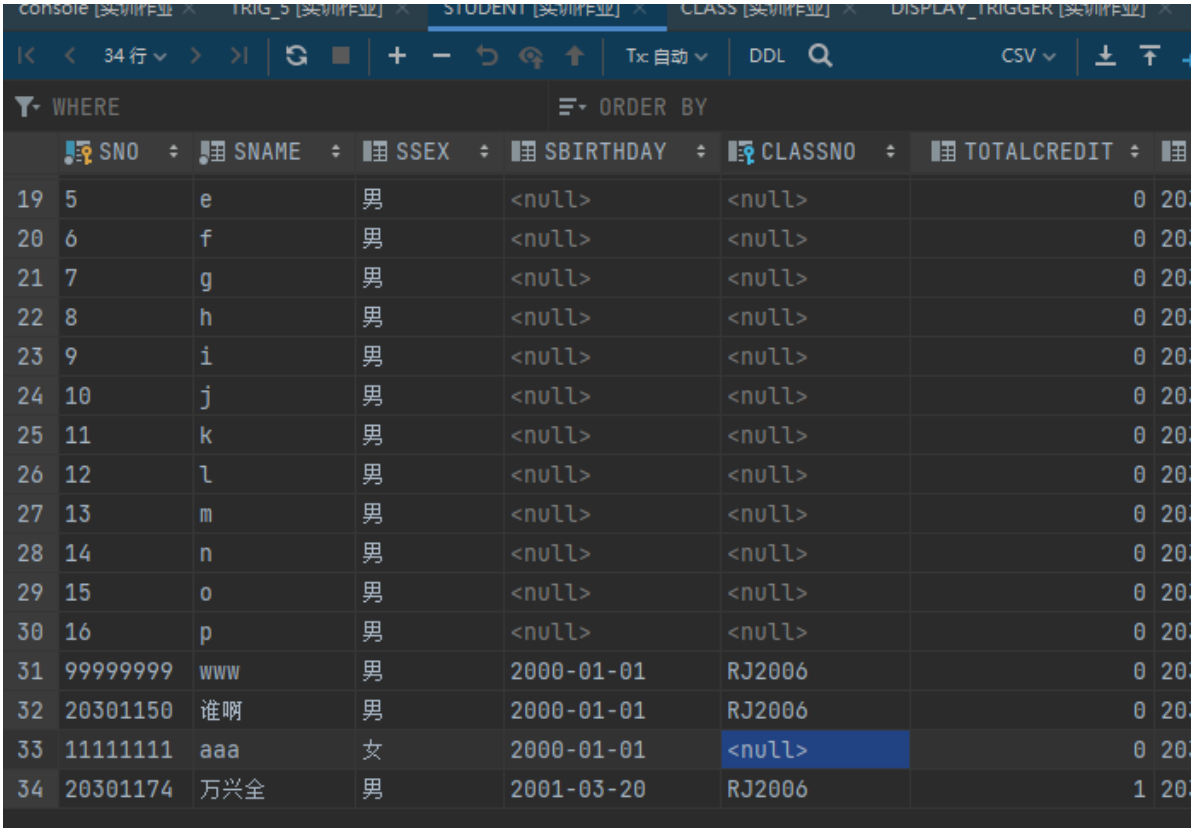
	SNO	SNAME	SSEX	SBIRTHDAY	CLASSNO	TOTALCREDIT	Creat
19	5	e	男	<null>	<null>	0	20301174
20	6	f	男	<null>	<null>	0	20301174
21	7	g	男	<null>	<null>	0	20301174
22	8	h	男	<null>	<null>	0	20301174
23	9	i	男	<null>	<null>	0	20301174
24	10	j	男	<null>	<null>	0	20301174
25	11	k	男	<null>	<null>	0	20301174
26	12	l	男	<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



WHERE

	SNO	CNO	GRADE
4	20301168	800002	98
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

增加选课后拿到一学分

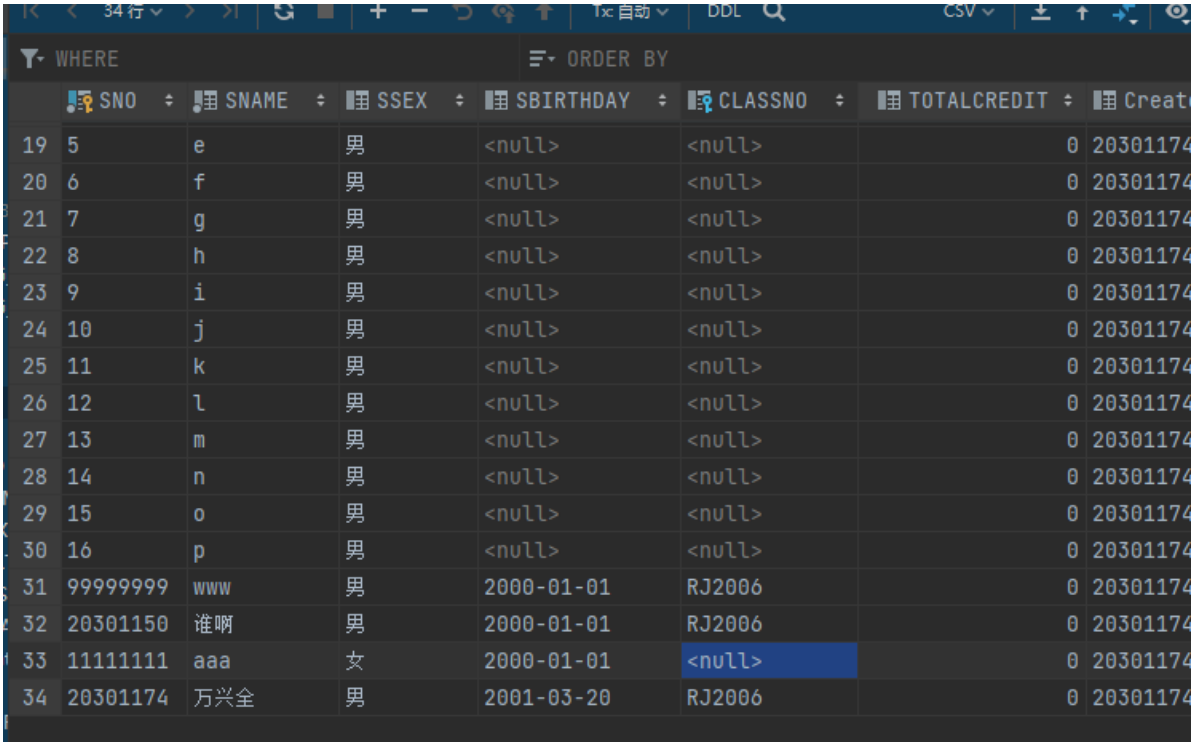


	SNO	SNAME	SSEX	SBIRTHDAY	CLASSNO	TOTALCREDIT	
19	5	e	男	<null>	<null>	0	20
20	6	f	男	<null>	<null>	0	20
21	7	g	男	<null>	<null>	0	20
22	8	h	男	<null>	<null>	0	20
23	9	i	男	<null>	<null>	0	20
24	10	j	男	<null>	<null>	0	20
25	11	k	男	<null>	<null>	0	20
26	12	l	男	<null>	<null>	0	20
27	13	m	男	<null>	<null>	0	20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

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	Creat
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

```
--第五次实验报告
--(一)
--1

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,sclassno);
end;

begin
    insert_student('20301174','万兴
全','男',to_date('2001/3/20','yyyy/mm/dd'),'RJ2006'
);
end;
select *from student;


--2
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;

begin
    insert_course(000000,'体育课');
end;
call
    insert_course(000001,'不上课',1);

select * from course;


--3
create or replace procedure query_student (
 ssno in student.sno%type
 )
as
ssname student.sname%type;
sssex  student.ssex%type;
ssbirthday student.sbirthday%type;
sclassno student.classno%type;
begin
select sname,ssex,sbirthday,classno
into ssname,sssex,ssbirthday,sclassno
from student
where sno=ssno;

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

begin   query_student('20301150');
END;
```

```
--4
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)
    into ssname,sssex,ssage,agrade
    from student,sc
    where sc.sno=student.sno
     having avg(grade)>80
    group by sname, ssex, age;
end;

call at_least_80();
    select sname,ssex,age,avg(grade)
    from student,sc
    where sc.sno=student.sno
    having avg(grade)>80
    group by sname, ssex, age;
--5
create or replace procedure cs(
asname student.sname%type,
ssno out student.sno%type,
ssname out student.sname%type,
ca out int,
ag out int,
ttc out student.totalcredit%type

)is
cursor csc is select
sc.sno,sname,count(cno),avg(grade),"Totalcredit"
into ssno,ssname,ca,ag,ttc
from student join sc s on student.sno = s.sno
and student.sno in(select sno from student
```

```
                                    where not
exists(select *
                                        from (select cno
                                              from sc
                                              where sno=
(select sno

from student

where sname=asname))t
                                        where not exists(
                                            select *from
sc where sc.cno=t.cno and sc.sno=student.sno
                                            )))
group by sc.cno,sname,"Totalcredit";
    begin open csc;
    loop fetch csc into ssno,ssname,ca,ttc;
    exit when csc%notfound;

        end loop;

DBMS_OUTPUT.PUT_LINE(ssno||','||ssname||','||ag);
    close csc;
    end cs;

        declare
ssno  student.sno%type;
ssname student.sname%type;
ca int;
ag int;
ttc  student.totalcredit%type;
    begin

        cs('万兴全',ssno,ssname,ca,ag,ttc);
            end;

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;
call chosen_class('rr');
    ;


--6
create or replace procedure show_lasthalf(
cclassno class.classno%type
)as
    csno student.sno%type;
    csname student.sname%type;
    cag sc.grade%type;
begin
    select student.sno,student.sname,avg(grade)
    into csno,csname,cag
    from student join sc s on student.sno = s.sno
    group by student.sno, student.sname
    having avg(grade)< (select avg(grade) avg
                        from sc
                            join student s2 on
s2.sno = sc.sno
                            join class c3 on
c3.classno = s2.classno
                        where c3.classno=cclassno);
```

```
    end;

    --游标（后知后觉）
create or replace procedure show_lasthalf
    (ssclassno student.classno%type)
is
    sssno student.sno%type;
    sssname student.sname%type;
    ssavggrade sc.grade%type;
begin
    declare
        cursor sscursor is
 select s.sno,sname,avg(grade)
 from student s,sc
 where s.sno=sc.sno
 group by s.sno,sname
 having avg(grade)<(select avg(grade)
                from student join sc using(sno)
                where classno=ssclassno
                group by classno);
        begin
 open sscursor;
 loop
     fetch sscursor into sssno,sssname,ssavggrade;
     exit when sscursor%notfound;

dbms_output.put_line('sno:'||sssno||',name:'||sssna
me||',avg(grade):'||ssavggrade);
 end loop;
 close sscursor;
        end;
end;

call show_lasthalf('RJ2003');
```

```
--(二)
--1

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;
 call
insert_student('99999999','www','男',to_date('2000/
1/1','yyyy/mm/dd'),'RJ2006');
```

```
23
24
20 --2
21 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
23 --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
20         where classno=:old.classno;
20     end;
26
20 delete
28 from student
26 where sno='99999998';
26 select *from class;
26
26 --4
28 create or replace trigger cc
29     after update on student
20     for each row
21     begin
22         update class
23             set studentnumber=studentnumber-1
```

```
        where classno=:old.classno;
        update class
            set studentnumber=studentnumber+1
        where classno=:new.classno;
    end;
UPDATE STUDENT
SET CLASSNO = 'RJ2006'--原来是2005
WHERE SNO LIKE '20301150' ;
select *from class;




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

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

--6
create or replace trigger stop_lower
    before update of grade on sc
    for each row
    begin
        if:new.grade<:old.grade
```

```
            then
raise_application_error('12345','球球了，不能再低了');
        end if;
    end;

update sc set grade=59 where sno='99999999';

--7
create or replace trigger trig_7
    after insert on sc
    for each row
    declare
        PRAGMA autonomous_transaction ;
        cursor cc is select count(cno)from sc group by sno;
        ca int;
        begin
        open cc;
        loop fetch cc into ca;
        exit when cc%notfound;
        if ca>=3 then
            raise_application_error(54321,'太卷啦太卷啦');
        end if;
    end loop;
        end;
insert into sc values (20301168,'800003',100);



--8
create or replace trigger TRIG_8
    after insert or delete
    on SC
    for each row
begin
    if inserting then
```

```
        update student
            set TOTALCREDIT=TOTALCREDIT+(select ccredit
                                         from course
                                         where course.cno=:new.cno)
            where sno=:new.sno;
    else
        update student
            set TOTALCREDIT=TOTALCREDIT-(select ccredit
                                         from course
                                         where course.cno=:old.cno)
            where sno=:old.sno;

    end if;
end;

select ccredit from course where cno=1;

insert into sc
values ('20301174',1,90);
select "Totalcredit" from student where sno=20301174;

delete from sc where cno=1;
--麻了，真的麻了
```