

(문제 1) 학번, 학년을 입력으로 학생의 학년을 수정

<실행>

```
SQL> create or replace procedure test2
  (v_stu_no in student.stu_no%type,
   v_stu_grade in student.stu_grade%type)
is
begin
  update student
    set stu_grade = v_stu_grade
    where stu_no = v_stu_no;
end test2;

exec test2(20213075,2);
```

STU_NO	STU_NAME	STU_DEPT	STU_GRADE	STU_CLASS	STU_GENDER	STU_HEIGHT	STU_WEIGHT
20213075	육한빛	기계	2	C	M	177	80

(문제 2) 학번을 입력으로 학생의 이름을 검색

```
SQL> create or replace procedure test3
  (v_stu_no in student.stu_no%type,
   v_stu_name out student.stu_name%type)
is
begin
  select stu_name
  into v_stu_name
  from student
  where stu_no = v_stu_no;
end test3;

exec test3(20213075,:d_stu_name);
```

Name	Type	IN/OUT	Value
d_stu_name	String	IN	육한빛

(문제 3) 학생의 점수를 임의 점수만큼 올려주는 프로시저

```
SQL> create or replace procedure test4
(v_sub_no in enrol.sub_no%type,
v_stu_no in enrol.stu_no%type,
v_enr_grade in out enrol.enr_grade%type)
is
begin
    update enrol
    set enr_grade = enr_grade + v_enr_grade
    where stu_no = v_stu_no
    and sub_no = v_sub_no;
    select enr_grade
    into v_enr_grade
    from enrol
    where stu_no = v_stu_no
    and sub_no = v_sub_no;
end test4;
```

```
exec test4(101,20191001,:d_enr_grade);
```

d_enr_grade	Integer	IN/OUT	90
-------------	---------	--------	----

(문제 4) 과목 추가, 과목번호는 과목이 입력되는 순서대로 부여

● 시퀀스 seq1, 201부터 999까지 한 번에 1씩 증가

```
SQL> create sequence seq1
    increment by 1
    start with 201
    maxvalue 999;
```

```
SQL> create procedure test5
(v_sub_name in subject.sub_name%type,
v_sub_prof in subject.sub_prof%type,
v_sub_grade in subject.sub_grade%type,
v_sub_dept in subject.sub_dept%type)
is
begin
    insert into subject
    values(seq1.nextval, v_sub_name,
    v_sub_prof, v_sub_grade, v_sub_dept);
    commit;
end test5;
```

```
exec test5('앱', '홍길동', 3, '컴퓨터정보');
```

SUB_NO	SUB_NAME	SUB_PROF	SUB_GRADE	SUB_DEPT
101	컴퓨터개론	강종영	3	컴퓨터정보
102	기계공학법	김태영	1	기계
103	기초전자실습	김유석	1	전기전자
104	시스템분석설계	강석현	3	컴퓨터정보
105	기계요소설계	김명성	1	기계
106	전자회로실습	최영민	3	전기전자
107	CAD응용실습	구룡규	2	기계
108	소프트웨어공학	권민성	1	컴퓨터정보
109	자동화설계	박민영	3	기계
110	자동제어	정순정	2	전기전자
111	데이터베이스	이재영	2	컴퓨터정보
112	컴퓨터구조	강주봉	2	컴퓨터정보
201	앱	홍길동	3	컴퓨터정보

(문제 5) 성적 환산 함수

```
SQL> create or replace function test6
(v_enr_grade in number)
return char
is
    enr_score char;
begin
    if v_enr_grade >= 90 then enr_score := 'A';
    elsif v_enr_grade >= 80 then enr_score := 'B';
    elsif v_enr_grade >= 70 then enr_score := 'C';
    elsif v_enr_grade >= 60 then enr_score := 'D';
    else enr_score := 'F';
    end if;
    return (enr_score);
end test6;
```

ENR_GRADE	SCORE
66	D
56	F

(문제 6) 101번 과목 수강하는 학생의 학번과 성적을 확인

```
SQL> create or replace procedure test7
is
    v_stu_no enrol.stu_no%type;
    v_sub_no enrol.sub_no%type;
    v_enr_grade enrol.enr_grade%type;
    cursor t_cursor is
        select stu_no, sub_no, enr_grade
        from enrol
```

```

        where sub_no = 101;
begin
    open t_cursor;
    loop
        fetch t_cursor into v_stu_no, v_sub_no, v_enr_grade;
        exit when t_cursor%NOTFOUND;
        dbms_output.put_line(v_stu_no||' ='||v_sub_no||' ='||v_enr_grade);
    end loop;
    close t_cursor;
end test7;

```

execute test7;

Line	Position	Message
1		20191001 =101 =90
2		20191025 =101 =65

(문제 7) test7.sql을 FOR문 사용하여 test8.sql 작성

SQL> create or replace procedure test8

is

```

    v_enrol enrol%rowtype;
    cursor t_cursor is
        select stu_no, sub_no, enr_grade
        from enrol
        where sub_no = 101;

```

begin

```

    for v_enrol in t_cursor loop
        exit when t_cursor%NOTFOUND;
        dbms_output.put_line(v_enrol.stu_no||'=='||v_enrol.sub_no||'=='||v_enrol.enr_grade);
    end loop;
end test8;

```

Line	Position	Message
1		20191001==101==90
2		20191025==101==65

(문제 8) 과목 수강자(학번) 과 성적 검색

```
SQL> create or replace procedure test9
(v_sub_no in enrol.sub_no%type)
is
v_stu_no enrol.stu_no%type;
v_enr_grade enrol.enr_grade%type;
cursor t_cursor(v_sub_no number) is
select student.stu_no, enr_grade
from student, enrol
where student.stu_no = enrol.stu_no
and enrol.sub_no = v_sub_no;
begin
open t_cursor(v_sub_no);
loop
fetch t_cursor into v_stu_no, v_enr_grade;
exit when t_cursor%NOTFOUND;
dbms_output.put_line(v_stu_no||'=='||v_enr_grade);
end loop;
close t_cursor;
end test9
```

Line	Position	Message
1		20191001==90
2		20191025==65

(문제 9) create or replace procedure test10

```
(v_stu_no in student.stu_no%type)
is
v_stu_name student.stu_name%type;
begin
select stu_name
Into v_stu_name
from student
where stu_no = v_stu_no;
dbms_output.put_line(v_stu_name);
exception
when no_data_found then
dbms_output.put_line('해당 데이터가 없습니다. ');
end test10;
```

Line	Position	Message
1		이태연

(문제 10) 과목 수강자 수 검색

```
SQL> create or replace procedure test11
  (v_sub_no in enrol.sub_no%type)
is
  v_cnt number;
  cnt_error exception;
begin
  select count(stu_no) into v_cnt from enrol where sub_no = v_sub_no;
  dbms_output.put_line(v_sub_no||'과목 수강자는'||v_cnt||' 명입니다.' );
  if v_cnt = 0 then
    raise cnt_error;
  end if;
exception
  when cnt_error then
    dbms_output.put_line('수강자가 없습니다.');
```

end test11;

Line	Position	Message
1		120과목 수강자는0 명입니다.
2		수강자가 없습니다.

(문제 11) 상품코드를 입력받아 재고량을 반환하는 함수를 작성하여라.
함수 이름 : f_stock_return

<실행>

```
SQL> variable y NUMBER
```

```
SQL> execute :y := f_stock_return(501);
```

상품코드가 없습니다.

```
SQL> execute :y := f_stock_return(501);
```

y
20