PL/SQL 메소드 만들기

자주 사용하는 작업을 프로시저에 미리 만들어 메소드처럼 사용하는법

procedure()메소드 만들기

```
CREATE OR REPLACE PROCEDURE UPDATE_SAL
(P_EMP_ID IN NUMBER, P_AMOUNT IN NUMBER)
IS
--here you define variables
-- n number;
BEGIN

  UPDATE employees
  set salary=salary+P_AMOUNT
  where employee_id=P_EMP_ID;
 commit;

exception
```

```
WHEN OTHERS THEN
DBMS_OUTPUT.PUT_LINE (SQLCODE);
DBMS_OUTPUT.PUT_LINE (SQLERRM);

END;

-- Procedure UPDATE_SAL이(가) 컴파일되었습니다.

-- CREATE OR REPLACE PROCEDURE 사용법 1
execute UPDATE_SAL (100,50);

-- CREATE OR REPLACE PROCEDURE 사용법 2
begin
UPDATE_SAL (&emp_id,&amount); --UPDATE_SAL (100,50)
end;
```

procedure()메소드 제거하기

```
drop procedure UPDATE_SAL;
```

메소드 찾기 불러오기

```
select * from user_objects
where object_name='UPDATE_SAL';
------------------------
select * from user_source
where name='UPDATE_SAL'
order by line;
------------------------
select * from user_objects
```

메소드 인자값 변경에 대한 동작 증명 1

```
create or replace procedure add_products --##insert 메소드 존재
(p_prod_id number,p_prod_name varchar2,p_prod_type  varchar2)
is
begin

  insert into products values (p_prod_id,p_prod_name,p_prod_t
 commit;

exception
when others then
dbms_output.put_line ('error in insert ');
dbms_output.put_line (sqlcode);
dbms_output.put_line (sqlerrm);
end;

execute add_products (1,'Laptop','SW'); -- 동작
select * from products;
execute add_products (2,'PC');  -- 인자값이 부족한경우 미동작

execute add_products (p_prod_id=>2,p_prod_name=>'PC',p_prod_t
execute add_products (p_prod_name=>'Keyboard',p_prod_id=>3,p_
--위와같이 인자값 순서 틀려도 지정시 동작
execute add_products (4,p_prod_type=>'SW',p_prod_name=>'Windo



----------------------------인자값 지정
create or replace procedure add_products
(p_prod_id number,p_prod_name varchar2:='Ukowun',p_prod_type
is
begin

  insert into products values (p_prod_id,p_prod_name,p_prod_t
  dbms_output.put_line(p_prod_id||' '||p_prod_name||'  insert
 commit;
```

```
exception
when others then
dbms_output.put_line ('error in insert '||p_prod_id||' '||p_p
dbms_output.put_line (sqlcode);
dbms_output.put_line (sqlerrm);
end;



-
begin
add_products(10,'PC');
add_products(10,'Labtop');
add_products(20,'Keyboard');
end;
--출력  중복시 에러처리해서 멈추지 않음
                                                --- 만약 excption0
                                                --- 만약 excption0
                                                --- 만약 excption0
20  Keyboard    Ukowun
10  PC  Ukowun
```

메소드 인자값 변경에 대한 동작 증명 2

```
create or replace PROCEDURE test_plsql_records --메소드에 테이블
( rec in DEPARTMENTS%rowtype )
is
begin
insert into DEPARTMENTS values rec;

end;

-------
declare
v DEPARTMENTS%rowtype;   --테이블 레코드 규격을 가져와 채월넣는다
begin
v.DEPARTMENT_ID:=3;   --null아님으로 id,name 인자값은 지정해야 한다
v.DEPARTMENT_NAME:='v dept';
```

```
test_plsql_records (v);     --실행시 인자값인 레코드가 같아 정상 동작
end;
```

메소드 in,out,in out

[파라미터 IN, OUT, IN OUT]

IN : 읽기전용 파라미터 (내부에서 쓰일 변수) — 값넣는용도

OUT : 프로시저에서 값을 변경할 수 있음 (외부에서 쓰일 변수) —return 반환가

IN OUT : 읽고쓰는 작업을 동시에 할 수 있음 (내부에서 대입된 함수와 동시에 해당 변수를 리턴)

- out, inout 은 기본값 세팅 불가 //값지정 불