PL/SQL

오라클에서 제공하는 프로그래밍 언어

일반 프로그래밍 요소 다 가지고 있고 DB업무 처리하기 위한 최적화된 언어;

DECLARE ; 선언부 변수, 상수, 커서 선언

BEGIN ; 실행부 SQL에 선택, 반복처리 등

END

set serveroutput on : PL/SQL의 표준 출력문. PLSQL안에서는 print로는 안됨

<반복처리문>

1. LOOP~END LOOP
2. FOR~LOOP
3. WHILE~LOOP

1.

EX>set serveroutput on

declare

v\_num number :=1;

begin

if v\_num =1 then dbms\_output.put\_line('DB');

else dbms\_output.put\_line('C');

end if;

end;

/

2.

set serveroutput on

DECLARE

V\_CNT NUMBER :=0;

V\_SUM NUMBER :=0;

BEGIN

FOR

V\_CNT IN 1..10 LOOP

V\_SUM := V\_SUM+V\_CNT;

DBMS\_OUTPUT.PUT\_LINE(V\_CNT||' '||V\_SUM);

END LOOP;

END;

/

SP(stored procedure)

create or replace procedure test

(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 v\_stu\_no=stu\_no;

end test;

/

SELECT

\*

FROM student;

variable d\_stu\_name varchar2(20);

execute test(20131025,:d\_stu\_name);

print d\_stu\_name;

ex2

create or REPLACE function prop

RETURN emp.ename%type

is

vename emp.ename%type;

begin

select ename

into vename from emp where sal = (select max(sal) from emp);

return vename;

end prop;

/

SELECT DISTINCT prop() from emp;

create or REPLACE function mola

(v\_dept student.stu\_dept%type)

return number

is

v\_studev number;

begin

SELECT STDDEV(enrol.enr\_grade)

into v\_studev from enrol natural join student where stu\_dept = v\_dept;

return v\_studev;

end mola;

/

SELECT \* from student natural join enrol;

SELECT DISTINCT mola('기계') from student;

create or REPLACE PROCEDURE mola2

(v\_empno emp.empno%type,

v\_job emp.job%type

)

is

begin

UPDATE emp set emp.job = v\_job where emp.empno = v\_empno;

end mola2;

/

SELECT

\*

FROM emp;

EXECUTE mola2(7839,'루팡');