CORRECTION TP 1

Activité 2 :

DECLARE

n NUMBER(2);

CURSOR employes IS SELECT empno, ename, job, mgr, hiredate, sal, comm, deptno FROM scott.emp;

--employe employes%ROWTYPE;

newsal scott.emp.sal%TYPE;

newcomm scott.emp.comm%TYPE;

empvide EXCEPTION;

BEGIN

--Test table vide

SELECT COUNT(\*) INTO n FROM scott.emp;

IF n=0 THEN

RAISE empvide;

END IF;

--Remplissage emp\_tn

FOR employe IN employes LOOP

--Calculs

newsal:=employe.sal\*3.1;

newsal:=newsal\*1.25;

IF employe.comm IS NOT NULL THEN

newcomm:=employe.comm\*3.1;

ELSE

newcomm:=NULL;

END IF;

--Insertion de données

INSERT INTO emp\_tn VALUES(employe.empno, employe.ename, employe.job, employe.mgr, employe.hiredate, newsal, newcomm, employe.deptno);

END LOOP;

EXCEPTION

WHEN empvide THEN RAISE\_APPLICATION\_ERROR (-20501, 'Pas d’employé !');

END;

--select \* from emp\_tn;

Activité 3 :

declare

cNum cleaner.cno%type:= &test;

ccName cleaner.cname%type;

ccSalary cleaner.csalary%type;

status boolean;

procedure getCleanerDetails(CleanerNo in cleaner.cno%type, CleanerName out cleaner.cname%type,CleanerSalary out cleaner.csalary%type,status out Boolean)

is

begin

select cname, csalary

into CleanerName, CleanerSalary

from cleaner

where cno= CleanerNo;

status :=true;

exception

when NO\_DATA\_FOUND THEN status:=false;

end;

begin

--cnum:='117';

getCleanerDetails(cNum,ccName,ccSalary,status);

if (status) then

dbms\_output.put\_line(cnum||' '||ccname||' '||ccsalary\*1.1);

else

--dbms\_output.put\_line('test');

dbms\_output.put\_line ('Cleaner '||cNum||' introuvable');

end if;

end;

set serveroutput on

declare

cursor cleanerCursor is select cNo, cName, cSalary from Cleaner;

cleanerCursor\_rec cleanerCursor%rowtype;

status boolean := true;

-- définition de la procédure

procedure getAllCleanerDetails is

begin

open cleanerCursor;

fetch cleanerCursor into cleanerCursor\_rec;

while cleanerCursor%found loop

dbms\_output.put\_line(cleanerCursor\_rec.cNo||' '||cleanerCursor\_rec.cName||' '||cleanerCursor\_rec.cSalary \*1.1);

fetch cleanerCursor into cleanerCursor\_rec;

end loop;

-- Déclaration de l'exception de table cleaner vide

exception

when NO\_DATA\_FOUND THEN status:=false;

close cleanerCursor;

end;

begin

-- Traitement de l'exception et appel de la procédure

if (status) then

dbms\_output.put\_line('All Cleaners details :');

getAllCleanerDetails;

else

dbms\_output.put\_line ('The cleaner table is empty');

end if;

end;

Create or replace function getCleanersLocation(cleanerNum in Cleaner.cNo%type)

return Depot.dAddress %type as

dLocation depot.dAddress %type;

begin

select dAddress into dLocation from Cleaner c, Depot d

where cNo= cleanerNum and d.dNo=c.dNo;

return (dLocation);

end;

-- test de la fonction

select cName, getCleanersLocation(cNo) "Address" from Cleaner

where cNo='110';