

Prosit 3.v2.txt

```
-- exercice 1
CREATE OR REPLACE FUNCTION FN_NBREDEPARTEMENT
RETURN NUMBER IS
  nbr number;
BEGIN
  SELECT count(*) INTO nbr FROM departments ;
  RETURN nbr;
end;
/
-- exercice 2
CREATE OR REPLACE FUNCTION FN_NOMDEPT(v_emp  IN
number) RETURN varchar2 IS
  v_nom departments.department_name%type;
BEGIN

  SELECT department_name
  INTO v_nom
  FROM departments
  WHERE department_id=(SELECT department_id
                        FROM employees
                        WHERE employee_id=v_emp);

  RETURN v_nom;
end;
/
-- exercice 3
CREATE OR REPLACE FUNCTION FN_SALAIRE(v_emp  IN
number) RETURN number IS
  v_sal number;
BEGIN

  SELECT salary
  INTO v_sal
  FROM employees
  WHERE employee_id=v_emp);
  RETURN v_sal;
end;
/
-- exercice 4
CREATE OR REPLACE FUNCTION FN_NBR_SALARIE(v_dept
IN number)  RETURN NUMBER IS
  nbr number;
```

Prosit 3.v2.txt

```
BEGIN
  SELECT count(*) INTO nbr FROM employees WHERE
department_id=v_dept ;
  RETURN nbr;
end;
/

CREATE OR REPLACE procedure
PROC_NBR_SALARIE(v_dept IN number,nbr OUT number)
IS

BEGIN
  SELECT count(*) INTO nbr FROM employees WHERE
department_id=v_dept ;

end;
/
-- exercice 5

CREATE OR REPLACE procedure PROC_TEST_NBR_SALARIE
IS
  CURSOR c IS SELECT department_id,department_name
                FROM departments;
BEGIN
  FOR rec IN c LOOP
    IF (rec.department_id BETWEEN 10 AND 110)
    AND  (FN_NBR_SALARIE(rec.department_id)>40)
THEN
      dbms_output.put_line(rec.department_name);
    END if;

  END loop;
end;

/
-- exercice 6
CREATE OR REPLACE PROCEDURE PROC_ANNEE_DEMBAUCHE
IS
  CURSOR c IS  SELECT count(*)
nbr,to_char(hire_date,'yyyy') annee
                FROM employees
```

```

        Prosit 3.v2.txt
        GROUP by to_char(hire_date,'yyyy')
        HAVING count(*)>10;
BEGIN
    FOR rec IN c loop
        dbms_output.put_line(rec.annee||'
'||rec.nbr);
    END loop;
end;
/
-- exercice 7
create or replace PROCEDURE PROC_JOBHISTORY IS
    CURSOR c IS SELECT employee_id
                  FROM employees
                  INTERSECT
                  SELECT employee_id
                  FROM job_history;
BEGIN
    FOR rec IN c loop
        dbms_output.put_line(rec.employee_id);
    END loop;
end;
/
-- exercice 9
create or replace procedure PROC_LISTE_EMP_DEPT IS

    CURSOR dept_cursor IS SELECT
department_id,department_name
                           FROM      departments
                           ORDER BY department_id;

    CURSOR emp_cursor(v_deptno NUMBER) IS SELECT
last_name,job_id,hire_date,salary
                                                FROM
employees
                                                WHERE
department_id = v_deptno;

BEGIN
    for cur in dept_cursor LOOP

```

```

                                Prosit 3.v2.txt
        DBMS_OUTPUT.PUT_LINE ('Department Number : '
|| cur.department_id || '    Department Name : '
|| cur.department_name);
        for cur_e in emp_cursor (cur.department_id)
LOOP
        DBMS_OUTPUT.PUT_LINE (cur_e.last_name|| '
' || cur_e.job_id || '    ' || cur_e.hire_date
|| '    ' || cur_e.salary);
        END LOOP;

DBMS_OUTPUT.PUT_LINE('-----
-----
-----');
        END LOOP;
END;
/

```

```

-- exercice 10
create or replace procedure
PROC_LISTE_EMP(v_manager number) IS
        CURSOR c IS SELECT last_name,first_name,salary
                        FROM employees
                        WHERE manager_id = v_manager;
BEGIN
        FOR rec IN c loop
                dbms_output.put_line(rec.last_name|| '
' || rec.first_name || '    ' || rec.salary);
        END loop;
end;
/

```

```

-- exercice 10 v1
create or replace procedure PROC_LISTE_EMP2 IS
        CURSOR c IS SELECT
last_name,to_char(hire_date,'yyyy') annee
                        FROM employees e

```

```

                                Prosit 3.v2.txt
                                WHERE to_char(hire_date,'yyyy')>
(SELECT to_char(hire_date,'yyyy')

FROM employees

WHERE employee_id=e.manager_id);
BEGIN
    FOR rec IN c loop
        dbms_output.put_line(rec.last_name||
' ||rec.annee);
    END loop;
end;
/
-- exercice 11
create or replace procedure FN_MOY_SALAIRE(v_emp
number) is
    moy    number;
BEGIN
    select avg(salary)
    into moy
    from employees
    group by department_id
    having department_id =(select department_id
from employees where employee_id=v_emp);

    dbms_output.put_line(moy);
end;
/
-- exercice 12
create or replace procedure PROC_SAL_SUP (v_emp
number) is
    CURSOR c IS select employee_id ,last_name
                from employees
                WHERE salary >(select salary from
employees where employee_id=v_emp);
BEGIN
    FOR rec IN c loop
        dbms_output.put_line(rec.employee_id||
' ||rec.last_name);
    END loop;
end;

```

Prosit 3.v2.txt

/

EXECUTE PROC_SAL_SUP(121);