

### Exercice 1

Parmi les déclarations de variables suivantes, déterminer celles qui sont incorrectes en justifiant votre réponse :

A - DECLARE v_id NUMBER(4);	<u>Correcte</u>
B - DECLARE v_x,v_y,v_z VARCHAR2(10);	<u>Incorrecte : un seul identifiant par ligne</u>
C - DECLARE v_date_naissance DATE NOT NULL;	<u>Incorrecte : une valeur NOT NULL doit être initialisée</u>
D - DECLARE v_en_stock BOOLEAN := 1;	<u>Incorrecte : 1 n'est pas une valeur booléenne</u>
E - DECLARE emp_record emp_record_type;	<u>Incorrecte : EMP_RECORD TYPE doit être déclaré</u>
F - DECLARE TYPE type_table_nom IS TABLE OF VARCHAR2(20) INDEX BY BINARY_INTEGER; dept_table_nom type_table_nom;	<u>Correcte</u>

### Exercice 2

**Réponse :** Non. La procédure ProcedureA appelle la procédure ProcedureB, alors ProcedureB doit être déclarée avant ProcedureA de sorte que la référence à ProcedureB puisse être résolue.

### Exercice 2

DECLARE

```
TYPE TabReels IS VARRAY(10) OF REAL;
notes TabReels := TabReels(10.5, 15, 13);
i INTEGER;
m REAL := 0;
```

BEGIN

```
notes.EXTEND(7);
notes(4) := 8.5;
notes(5) := 18.5;
notes(6) := 16;
notes(7) := 4;
notes(8) := 9.5;
notes(9) := 12;
notes(10) := 11;
FOR i IN 1..notes.COUNT LOOP
    m := m + notes(i);
END LOOP;
m := m / notes.COUNT;
DBMS_OUTPUT.PUT_LINE('Moyenne = '||TO_CHAR(m));
DBMS_OUTPUT.PUT_LINE('Taille actuelle / Taille limite = '
    ||notes.COUNT||' / '||notes.LIMIT);

notes.TRIM;
DBMS_OUTPUT.PUT_LINE('Taille actuelle / Taille limite = '
    ||notes.COUNT||' / '||notes.LIMIT);

notes.DELETE;
DBMS_OUTPUT.PUT_LINE('Taille actuelle / Taille limite = '
    ||notes.COUNT||' / '||notes.LIMIT);
END;
```

### Exercise 3

CREATE OR REPLACE FUNCTION F\_MTT (E\_NUM NUMBER) RETURN NUMBER IS

CURSOR C IS

SELECT (SAL\*12 + NVL(COMM,0) ) FROM EMP WHERE EMPNO=E\_NUM;

W\_MT NUMBER:=0;

BEGIN

OPEN C;

FETCH C INTO W\_MT;

CLOSE C;

RETURN W\_MT;

END F\_MTT;

CREATE OR REPLACE PROCEDURE P\_INFOS(P\_Num in number) is

P\_Nom employe.ename%type;

P\_fonction emp.fonction%type;

CURSOR C IS

SELECT ename, fonction FROM emp WHERE EMPNO =P\_NUM;

BEGIN

OPEN C;

FETCH C INTO P\_Nom, P\_fonction;

CLOSE C;

DBMS\_OUTPUT.PUT\_LINE( 'Nom : ' || P\_Nom );

DBMS\_OUTPUT.PUT\_LINE( 'Fonction : ' || P\_fonction );

END P\_INFOS;

declare

n integer;

begin

update employe set salaire=salaire+200

where n\_dept =(select n\_dept from departement where nom='Commercial');

n:=Sql%ROWCOUNT;

DBMS\_OUTPUT.PUT\_LINE(n);

end;