



L'Ecole Nationale des Sciences Appliquées Kénitra  
Deuxième année cycle ingénieur - S7 - Génie informatique

Module : Administration et optimisation des bases de données

---

# Compte rendu de TP

## Utilisation des Variable

---

Année scolaire 2023-2024

Présenté par : BADRY ZAKARIA

Numéro Apogée : 22014301

Sous la direction de : Mme. CHAOUI Habiba

### Exercice 0 :

Création d'une procédure avec une variable v\_test de type VARCHAR2(10) :

```
CREATE OR REPLACE PROCEDURE check_type_error AS
v_test VARCHAR2(10);
BEGIN
NULL; -- Ne fait rien, juste pour vérifier la création de la
procédure
END;
```

```
1 v CREATE OR REPLACE PROCEDURE check_type_error AS
2   v_test VARCHAR2(10);
3 v BEGIN
4   NULL; -- Ne fait rien, juste pour vérifier la création de la procédure
5 END;
6 v /
7 EXECUTE check_type_error;
```

Procedure created.

Statement processed.

Affichage du nom de l'employé dont le numéro est 7369 :

```
SET SERVEROUTPUT ON
DECLARE
v_ename VARCHAR2(30);
BEGIN
SELECT ENAME INTO v_ename FROM EMP WHERE EMPNO = 7369;
DBMS_OUTPUT.PUT_LINE('Nom de l''employé : ' || v_ename);
END;
/
```

```

1 v SET SERVEROUTPUT ON
2 DECLARE
3     v_ename VARCHAR2(30);
4 BEGIN
5     SELECT ENAME INTO v_ename FROM EMP WHERE EMPNO = 7369;
6     DBMS_OUTPUT.PUT_LINE('Nom de l'employé : ' || v_ename);
7 END;
8 /

```

Unsupported Command

Statement processed.  
Nom de l'employé : SMITH

Modification du type de la colonne "ename" de VARCHAR2(10) à VARCHAR2(30) :

```
ALTER TABLE EMP MODIFY (ENAME VARCHAR2(30));
```

```
1 ALTER TABLE EMP MODIFY (ENAME VARCHAR2(30));
```

Table altered.

Vérification des types VARCHAR, VARCHAR2 et VARCHAR(10) :

```

1 v DECLARE
2     v_varchar VARCHAR(20);
3     v_varchar2 VARCHAR2(10);
4 v BEGIN
5     v_varchar := 'Test VARCHAR';
6     v_varchar2 := 'Test VARCHAR2';
7
8     DBMS_OUTPUT.PUT_LINE('v_varchar : ' || v_varchar);
9     DBMS_OUTPUT.PUT_LINE('v_varchar2 : ' || v_varchar2);
10
11 END;
12

```

```

DECLARE
v_varchar VARCHAR(20);
v_varchar2 VARCHAR2(10);
BEGIN
v_varchar := 'Test VARCHAR';
v_varchar2 := 'Test VARCHAR2';
DBMS_OUTPUT.PUT_LINE('v_varchar : ' || v_varchar);
DBMS_OUTPUT.PUT_LINE('v_varchar2 : ' || v_varchar2);

END;

```

### Exercice 1 :

```

SET SERVEROUTPUT ON

-- Création des variables avec le même type et la même
dimension que les colonnes de Dept
DECLARE
v_deptno Dept.DEPTNO%TYPE;
v_dname Dept.DNAME%TYPE;
v_loc Dept.LOC%TYPE;
BEGIN
-- Affectation des valeurs aux variables
v_deptno := 60;
v_dname := 'RHU';
v_loc := 'MONTRÉAL';
-- Affichage des valeurs des variables
DBMS_OUTPUT.PUT_LINE('v_deptno : ' || v_deptno);
DBMS_OUTPUT.PUT_LINE('v_dname : ' || v_dname);
DBMS_OUTPUT.PUT_LINE('v_loc : ' || v_loc);
END;
/

```

```

3  -- Création des variables avec le même type et la même dimension que les colonnes de De
4  DECLARE
5      v_deptno Dept.DEPTNO%TYPE;
6      v_dname Dept.DNAME%TYPE;
7      v_loc Dept.LOC%TYPE;
8  BEGIN
9      -- Affectation des valeurs aux variables
10     v_deptno := 60;
11     v_dname := 'RHU';
12     v_loc := 'MONTRÉAL';
13
14     -- Affichage des valeurs des variables
15     DBMS_OUTPUT.PUT_LINE('v_deptno : ' || v_deptno);
16     DBMS_OUTPUT.PUT_LINE('v_dname : ' || v_dname);
17     DBMS_OUTPUT.PUT_LINE('v_loc : ' || v_loc);
18 END;
19 /

```

Statement processed.  
v\_deptno : 60  
v\_dname : RHU  
v\_loc : MONTRÉAL

## **Exercice 2 :**

```

SET SERVEROUTPUT ON

-- Déclaration des variables correspondantes aux colonnes de
Dept
DECLARE
v_deptno Dept.DEPTNO%TYPE;
v_dname Dept.DNAME%TYPE;
v_loc Dept.LOC%TYPE;
BEGIN
-- Affectation des valeurs du département numéro 20 aux
variables
SELECT DEPTNO, DNAME, LOC INTO v_deptno, v_dname, v_loc FROM
Dept WHERE DEPTNO = 20;
-- Affichage des valeurs des variables
DBMS_OUTPUT.PUT_LINE('v_deptno : ' || v_deptno);
DBMS_OUTPUT.PUT_LINE('v_dname : ' || v_dname);
DBMS_OUTPUT.PUT_LINE('v_loc : ' || v_loc);
END;
/

```

```

1 v SET SERVEROUTPUT ON
2
3 -- Déclaration des variables correspondantes aux colonnes de Dept
4 DECLARE
5     v_deptno Dept.DEPTNO%TYPE;
6     v_dname Dept.DNAME%TYPE;
7     v_loc Dept.LOC%TYPE;
8 v BEGIN
9     -- Affectation des valeurs du département numéro 20 aux variables
10    SELECT DEPTNO, DNAME, LOC INTO v_deptno, v_dname, v_loc FROM Dept WHERE DEPTNO = 20;
11
12    -- Affichage des valeurs des variables
13    DBMS_OUTPUT.PUT_LINE('v_deptno : ' || v_deptno);
14    DBMS_OUTPUT.PUT_LINE('v_dname : ' || v_dname);
15    DBMS_OUTPUT.PUT_LINE('v_loc : ' || v_loc);
16 END;
17 /

```

Statement processed.  
v\_deptno : 20  
v\_dname : RESEARCH  
v\_loc : DALLAS

### **Exercice 3 :**

```

INSERT INTO DEPT (DEPTNO, DNAME, LOC)
VALUES (60, 'RHU', 'MONTRÉAL');

```

```
1 v INSERT INTO DEPT (DEPTNO, DNAME, LOC)
2 VALUES (60, 'RHU', 'MONTRÉAL');
```

1 row(s) inserted.

#### **Exercice 4 :**

Création des variables sous SQL\*Plus

```
VARIABLE v_deptno NUMBER;
VARIABLE v_dname VARCHAR2(20);
VARIABLE v_loc VARCHAR2(20);

-- Affectation des valeurs aux variables
EXEC :v_deptno := 70;
EXEC :v_dname := 'Finance';
EXEC :v_loc := 'Québec';
```

Création d'un bloc PL/SQL pour affecter les variables SQL\*Plus à d'autres variables

```
SET SERVEROUTPUT ON
DECLARE
v_deptno_plsql NUMBER;
v_dname_plsql VARCHAR2(20);
v_loc_plsql VARCHAR2(20);
BEGIN
-- Affectation des valeurs des variables SQL*Plus aux
variables PL/SQL
```

```

v_deptno_plsql := :v_deptno;
v_dname_plsql := :v_dname;
v_loc_plsql := :v_loc;
-- Affichage des valeurs des variables PL/SQL
DBMS_OUTPUT.PUT_LINE('v_deptno_plsql : ' || v_deptno_plsql);
DBMS_OUTPUT.PUT_LINE('v_dname_plsql : ' || v_dname_plsql);
DBMS_OUTPUT.PUT_LINE('v_loc_plsql : ' || v_loc_plsql);
END;
/

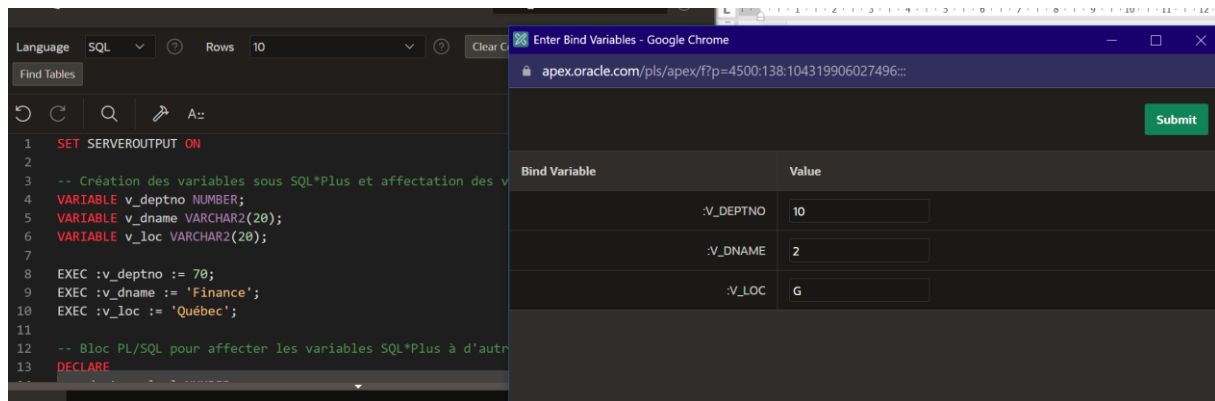
```

### Affichage des variables

```

-- Affichage des valeurs des variables SQL*Plus
PRINT v_deptno
PRINT v_dname
PRINT v_loc

```



The screenshot shows the Oracle APEX interface. On the left, the SQL editor contains the following code:

```

1 SET SERVEROUTPUT ON
2
3 -- Création des variables sous SQL*Plus et affectation des v
4 VARIABLE v_deptno NUMBER;
5 VARIABLE v_dname VARCHAR2(20);
6 VARIABLE v_loc VARCHAR2(20);
7
8 EXEC :v_deptno := 70;
9 EXEC :v_dname := 'Finance';
10 EXEC :v_loc := 'Québec';
11
12 -- Bloc PL/SQL pour affecter les variables SQL*Plus à d'autr
13 DECLARE

```

On the right, the 'Enter Bind Variables' dialog is open, displaying a table of bind variables and their values:

Bind Variable	Value
:V_DEPTNO	10
:V_DNAME	2
:V_LOC	G

### Exercice 5 :

```

DECLARE
v_deptno dept.deptno%TYPE;
v_dname dept.dname%TYPE;
v_loc dept.loc%TYPE;
BEGIN
-- Affectation des valeurs aux variables
v_deptno := 60;
v_dname := 'RHU';
v_loc := 'Montereal';
END;
/

```



```

4      v_loc dept.loc%TYPE;
5  BEGIN
6      -- Affectation des valeurs aux variables
7      v_deptno := 60;
8      v_dname := 'RHU';
9      v_loc := 'Montereal';

```

Statement processed.

### **Exercice 6 :**

```

-- Déclaration des variables
DECLARE
v_deptno dept.deptno%TYPE;
v_dname dept.dname%TYPE;
v_loc dept.loc%TYPE;
BEGIN
-- Affectation des valeurs des colonnes de la table DEPT
SELECT deptno, dname, loc INTO v_deptno, v_dname, v_loc
FROM dept
WHERE deptno = 30;

-- Affichage des valeurs des variables
DBMS_OUTPUT.PUT_LINE('v_deptno : ' || v_deptno);
DBMS_OUTPUT.PUT_LINE('v_dname : ' || v_dname);
DBMS_OUTPUT.PUT_LINE('v_loc : ' || v_loc);
END;
/

```

```

11
12  -- Affichage des valeurs des variables
13  DBMS_OUTPUT.PUT_LINE('v_deptno : ' || v_deptno);
14  DBMS_OUTPUT.PUT_LINE('v_dname : ' || v_dname);
15  DBMS_OUTPUT.PUT_LINE('v_loc : ' || v_loc);
16  END;
17  /

```

Statement processed.  
v\_deptno : 30  
v\_dname : SALES  
v\_loc : CHICAGO

### **Exercice 7 :**

```

-- Déclaration de la variable de type RECORD
DECLARE
v_dept dept%ROWTYPE;
BEGIN
-- Affectation de la ligne de la table DEPT où deptno=30 à la
variable v_dept
SELECT *
INTO v_dept
FROM dept
WHERE deptno = 30;

-- Affichage du contenu de la variable v_dept
DBMS_OUTPUT.PUT_LINE('v_dept.deptno : ' || v_dept.deptno);
DBMS_OUTPUT.PUT_LINE('v_dept.dname : ' || v_dept.dname);
DBMS_OUTPUT.PUT_LINE('v_dept.loc : ' || v_dept.loc);
END;
/

```

```
L0
L1  -- Affichage du contenu de la variable v_dept
L2  DBMS_OUTPUT.PUT_LINE('v_dept.deptno : ' || v_dept.deptno);
L3  DBMS_OUTPUT.PUT_LINE('v_dept.dname : ' || v_dept.dname);
L4  DBMS_OUTPUT.PUT_LINE('v_dept.loc : ' || v_dept.loc);
L5  END;
L6  /
```

Statement processed.  
v\_dept.deptno : 30  
v\_dept.dname : SALES  
v\_dept.loc : CHICAGO