



Université Sidi Mohamed Ben Abdellah

Faculté des Sciences DHAR EL-Mehraz

-FÉS-

Département de l'informatique

Compte rendu du TP 1

PL/SQL

Réaliser
par :

EL KHOU Mohammed

Question 1 :

Sur le fichier tp2-q1.sql :

```
--conn mh / mh ;
declare
    oldNum budget.NUM_OPERATION%type;
    oldSolde budget.SOLDE%type;
begin
    select MAX(NUM_OPERATION) INTO oldNum from Budget;
    select SOLDE INTO oldSolde from Budget where
    NUM_OPERATION=oldNum;

    INSERT INTO Budget (NUM_OPERATION, NOM_OPERATION,
    CATEGORIE, DATE_OPE, MONTANT, SOLDE)
    VALUES (oldNum+1 , 'Courses', 'Débit', ' 14/01/2002 ', 500,
    oldSolde-500);
end;
/
```

Question 2 :

```
CREATE TABLE COURSES (
    NUM_OPERATION          NUMBER (4,2)          PRIMARY KEY,
    TYPE_OPERATION         CHAR (15),
    CATEGORIE              CHAR (10),
    DATE_OPERATION         DATE,
    MONTANT                NUMBER (6, 2)) ;
```

--1ère variante:

```
DECLARE
    lbudget budget%ROWTYPE;
```

```

        CURSOR req1 IS SELECT * FROM BUDGET WHERE
        nom_operation='Courses';
BEGIN
    OPEN req1;
    LOOP
        FETCH req1 INTO lbudget;
        EXIT WHEN req1%NOTFOUND;
        INSERT INTO COURSES VALUES
        (lbudget.num_operation, lbudget.nom_operation,
        lbudget.categorie, lbudget.date_operation, lbudget.montant);
    END LOOP;
    CLOSE req1;
END;
/

```

--2ème variante :

```

CREATE TABLE COURSES (
NUM_OPERATION          NUMBER (4,2)          PRIMARY KEY,
TYPE_OPERATION          CHAR (15),
CATEGORIE              CHAR (10),
dATE_OPERATION          DATE,
MONTANT                NUMBER (6, 2)) ;

```

```

DECLARE
    Num_op      BUDGET.Num_operation%TYPE;
    Type_op     BUDGET. TYPE_OPERATION %TYPE;
    Categorie_op      BUDGET.Categorie%TYPE;
    Date_op      BUDGET.Date_Operation%TYPE;
    Montant_op    BUDGET.Montant%TYPE;
    CURSOR Curseur_1 IS SELECT Num_operation, Nom_operation,
    Categorie, Date_operation, Montant FROM BUDGET;

```

```

BEGIN
    OPEN Curseur_1;
    LOOP
        FETCH Curseur_1 INTO Num_op, Type_op, Categorie_op,
        Date_op, Montant_op;

        IF Nom_op = 'Courses' THEN
            INSERT INTO COURSES VALUES (Num_op, Type_op,
            Categorie_op, Date_op, Montant_op);
        END IF ;
        EXIT WHEN Curseur_1%NOTFOUND ;
    END LOOP ;
    CLOSE Curseur_1;
END;
/

```

```

SELECT * FROM COURSES ;

```

```

DROP TABLE COURSES;

```

Question 3 :

```

CREATE TABLE BUDGET_EURO (
    NUM_OPERATION          NUMBER (4,2)          PRIMARY KEY,
    TYPE_OPERATION         varCHAR2 (15),
    CATEGORIE              varCHAR2 (10),
    DATE_OPERATION         DATE,
    MONTANT                NUMBER (6, 2),
    SOLDE                  NUMBER (7, 2)) ;

```

```

DECLARE

```

```
i NUMBER;
```

```
Num_op BUDGET.Num_operation%TYPE := 0 ;
```

```
Type_op BUDGET.Type_operation%TYPE;
```

```
Categorie_op BUDGET.Categorie%TYPE;
```

```
Date_op BUDGET.Date_Operation%TYPE;
```

```
Montant_op BUDGET.Montant%TYPE;
```

```
Solde_op BUDGET.Solde%TYPE;
```

```
CURSOR Curseur_2 IS SELECT Type_operation, Categorie,  
Date_operation, Montant, Solde FROM BUDGET WHERE  
Date_operation>'01/01/2002';
```

```
BEGIN
```

```
l:=0;
```

```
OPEN Curseur_2;
```

```
LOOP
```

```
l:= i+1;
```

```
FETCH Curseur_2 INTO Type_op, Categorie_op, Date_op,  
Montant_op, Solde_op;
```

```
INSERT INTO BUDGET_EURO VALUES (Num_op, Type_op,  
Categorie_op, Date_op, Montant_op/6.55957,  
Solde_op/6.55957);
```

```
Num_op := Num_op + 1 ;
```

```
EXIT WHEN Curseur_2%NOTFOUND ;
```

```
END LOOP ;
```

```
CLOSE Curseur_2;
```

```
END;
```

/

```
SELECT * FROM BUDGET_EURO ;  
DROP TABLE BUDGET_EURO;
```

Question 4 :

```
DECLARE  
    Num_op  BUDGET.Num_operation%TYPE;  
    Montant_op  BUDGET.Montant%TYPE;  
    Erreur_Remb  EXCEPTION ;  
    CURSOR Curseur_3 IS SELECT Num_operation, Montant FROM  
    BUDGET WHERE nom_Operation = 'Remboursement';  
  
BEGIN  
    OPEN Curseur_3;  
    LOOP  
        FETCH Curseur_3 INTO Num_op, Montant_op;  
        IF Montant_op>2000 THEN  
            RAISE Erreur_Remb ;  
        END IF ;  
        EXIT WHEN Curseur_3%NOTFOUND ;  
    END LOOP ;  
    CLOSE Curseur_3;  
  
    EXCEPTION  
        WHEN Erreur_Remb THEN  
            DELETE FROM BUDGET WHERE Num_operation = Num_op  
            ;
```

END;

/