

## PL/SQL

Dans ce TD/TP, vous allez apprendre le langage PL/SQL proposé par Oracle. Voici un tableau récapitulatif des commandes PL/SQL principales.

Bloc PL/SQL	DECLARE -- Déclaration constantes/variables BEGIN -- Commandes/instructions EXCEPTION -- Traitement des erreurs à l'exé. END; / -- Exécution automatique lors du « start »
Déclaration de variable	Nom_Variable TYPE_VARIABLE;
Affectation	Nom_Variable:=valeur; SELECT attribut INTO Nom_Variable FROM table;
Tests	IF condition1 THEN -- Instructions ELSEIF condition2 THEN -- (Optionnel) -- Instructions ELSE -- (Optionnel) -- Instructions END IF;
Boucles	FOR compteur IN [REVERSE] min..max LOOP -- Instructions END LOOP; WHILE condition LOOP -- Instructions END LOOP;
Curseurs	- Déclaration CURSOR Nom_Curseur IS Requête_SQL; - Utilisation FOR tuple IN Nom_Curseur LOOP -- Instructions -- Ex. Nom_Variable:=tuple.attribut; END LOOP;-- NB : tuple est de type Nom_Curseur%ROWTYPE
Exceptions	- Déclarer Nom_Exception EXCEPTION; - Lever RAISE Nom_Exception; - Traiter WHEN Nom_Exception THEN -- Instruction;

### Partie 1

Le schéma de la BDD utilisée est composé de 2 tables EMP et DEPT.

1.

#### EMP

NUMEMP	NOM COMPLET	PROFESSION	CHEF	SALAIRE	NUMDEP
1	BARTH Florent	CHEF DE PROJET	3	13000	2
2	XAVIER Richard	CHERCHEUR	3	21000	1
3	NICOLLE Chris	CHERCHEUR		25000	1
4	BLAKE John	DEVELOPPEUR	6	8000	2
5	DUPONT Jean	DEVELOPPEUR	3	9000	2
6	MARTIN Alexandre	COMPTABLE	3	10000	3
7	RAY Benjamin	COMPTABLE	3	10000	3
8	MILLER Pascal	DEVELOPPEUR	3	9000	2
9	FORD John	DIRECTEUR	3	30000	4

## DEPT

NUMDEP	NOMDEP	NOMLOC
1	RECHERCHE	DIJON
2	DEVELOPPEMENT	NEW-YORK
3	FACTURATION	PARIS
4	DIRECTION	LONDRES

### A) Initiation PL/SQL

Soit le programme PL/SQL suivant :

```
DECLARE
    --
    n NUMBER(2);
    --
    CURSOR employes IS SELECT numemp, nomemp, salaire FROM EMP;
    --
    newsal emp.salaire%TYPE;
    --
    empv EXCEPTION;
BEGIN
    --
    SELECT COUNT(*) FROM EMP;
    --
    IF n=0 ALORS
        --
        RAISE empv;
    END IF;
    --
    FOR employe IN employes LOOP
        --
        newsal=employe.salaire+50;
        --
        UPDATE EMP SET SALAIRE=newsal where NUMEMP = employe.numemp;
    END LOOP;
    Commit;

    --
EXCEPTION
    --
    WHEN empv THEN dbms_output.put_line('Message d''erreur !');
END;
```

SQL> set serveroutput on

### B) Vos premiers pas avec PL/SQL

```
CREATE OR REPLACE PROCEDURE nom_proc (nom_param type_param,...) AS
-- Déclaration de variables.
BEGIN
-- Corps de la procédure
END;
```

```
DECLARE
BEGIN
    Nom_procedure;
END;
```

```
CREATE OR REPLACE FUNCTION nom_fonction (nom_param type_param)
RETURN Type_de_Retour IS
-- Déclaration de variables.
BEGIN
-- Corps de la fonction
RETURN Valeur_de_retour;
END;
```

Pour voir le résultat de votre fonction, vous pouvez utiliser la syntaxe suivante :

```
DECLARE
BEGIN
    dbms_output.PUT_LINE(Nom_de_Fonction(Parametres,...) );
END;
```

« NUMERO = *Valeur*, POSTE = *Valeur*, SALAIRE = *Valeur* »

## **C) Approfondissement**