



**ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR**

Membre de **HONORIS UNITED UNIVERSITIES**

École Marocaine des Sciences de l'Ingénieur

Filière: IIR

Les Bases de Données

Prof. Zakaria KHATAR

Procédures et Fonctions

1) Définitions :

- Procédure en PL/SQL :

- Une procédure en PL/SQL est un bloc de code structuré qui permet d'effectuer une tâche spécifique dans une base de données Oracle.
- Elle peut prendre des paramètres d'entrée pour traiter des données spécifiques et peut modifier ces données ou l'état de la base de données, mais elle ne retourne pas directement de valeur.
- Les procédures sont utiles pour regrouper et organiser des opérations complexes, comme les mises à jour en masse ou les traitements conditionnels, facilitant ainsi la maintenance et la lisibilité du code.

1) Définitions :

- Fonction en PL/SQL :

- Une fonction en PL/SQL est également un bloc de code, mais contrairement à une procédure, elle est conçue pour retourner une valeur unique.
- Elle peut accepter des paramètres d'entrée et est souvent utilisée pour effectuer des calculs, traiter des données et retourner un résultat qui peut être utilisé directement dans des requêtes SQL.
- Les fonctions sont idéales pour des opérations qui nécessitent une valeur de retour, comme le calcul d'un total, la transformation de données, ou la récupération d'informations spécifiques de la base de données.

1) Procédure en PL/SQL:

Syntaxe de la procédure :

```
CREATE [OR REPLACE] PROCEDURE nom_Procédure ( paramètre1 IN
type1, paramètre2 IN type2, ... ) IS  -- on utilise IS ou AS

    -- déclarations de variables locales (facultatif)
    nom_variable type_données;
    ...

BEGIN

    -- Instructions SQL et PL/SQL --

END nom_Procédure;
```

1) Les Procédures en PL/SQL:

Exemple :

Cette procédure prend en entrée l'**ID d'un employé** et un **nouveau salaire**, et met à jour le salaire de l'employé correspondant dans la table **employees**.

```
CREATE OR REPLACE PROCEDURE update_employee_salary (p_employee_id IN  
NUMBER, p_new_salary IN NUMBER) IS  
  BEGIN  
    UPDATE employees SET salary = p_new_salary  
    WHERE employee_id = p_employee_id;  
  COMMIT;  
END update_employee_salary;
```

1) Procédure en PL/SQL:

Syntaxe d'appel d'une procédure :

Après avoir créé une procédure en PL/SQL, vous pouvez l'appeler en utilisant la syntaxe suivante, en spécifiant les paramètres d'entrée, Il est important de respecter l'ordre et le type de ces paramètres tels qu'ils sont définis dans la déclaration de la procédure.

```
BEGIN
```

```
    procedure_name(parameter1, parameter2,...);
```

```
END;
```

2) Les Fonctions en PL/SQL:

Syntaxe de la fonction:

```
CREATE [OR REPLACE] FUNCTION nom_Fonction ( paramètre1 IN type1,  
paramètre2 IN type2, ... )  
RETURN type_données_retour IS  
    -- déclarations de variables locales (facultatif)  
    nom_variable type_données;  
    ...  
BEGIN  
    -- Instructions de la fonction --  
  
    RETURN valeur_retour;  
  
END nom_Fonction;
```


2) Les Fonctions en PL/SQL:

Exemple : Cette fonction prend en entrée l'**ID d'un employé**, et retourne le **nom complet** de l'employé correspondant à partir de la table **employees**.

```
CREATE OR REPLACE FUNCTION get_employee_name (f_employee_id IN  
NUMBER)  
RETURN VARCHAR2 IS  
    v_employee_name VARCHAR2(50);  
BEGIN  
    SELECT first_name || ' ' || last_name INTO v_employee_name  
        FROM employees  
        WHERE employee_id = f_employee_id;  
    RETURN v_employee_name;  
END get_employee_name;
```

2) Les Fonctions en PL/SQL:

Syntaxe d'appel d'une fonction :

Après avoir créé une fonction en PL/SQL, vous pouvez l'appeler en utilisant la syntaxe suivante, en spécifiant les paramètres d'entrée.

Les fonctions retournent une valeur, et vous pouvez les utiliser dans vos requêtes SQL ou les assigner à des variables dans un bloc PL/SQL. Il est important de respecter l'ordre et le type de ces paramètres tels qu'ils sont définis dans la déclaration de la fonction.

```
DECLARE
```

```
    result datatype;
```

```
BEGIN
```

```
    result := function_name(parameter1, parameter2);
```

```
END;
```

3) Les Fonctions en PL/SQL:

En PL/SQL, pour supprimer une fonction ou une procédure, on utilise les commandes DROP comme suit

Action	Commande PL/SQL
Supprimer une fonction	DROP FUNCTION nom_de_la_fonction;
Supprimer une procédure	DROP PROCEDURE nom_de_la_procedure;