

Chapitre V : Les Fonctions

1- Introduction aux fonctions :

Les fonctions sont de plusieurs types :

↳ Les fonctions scalaires qui retournent des valeurs scalaires

↳ Les fonctions tables en ligne qui retournent des tables

- Les fonctions scalaires retournent, grâce au mot clé RETURN, une valeur scalaire.

- Les fonctions table ne retournent comme résultat, qu'une table, qui est le résultat d'une instruction SELECT.

2- La création d'une fonction :

La création d'une fonction se fait quasiment de la même manière pour les deux types.

Voici les deux syntaxes principales de création de fonction (scalaire et table).

- Fonction table :

```
CREATE FUNCTION Nom_Fonction (Liste de Paramètres) Returns Table  
AS
```

```
Return (Requête Select)
```

Dans ce cas-là, après l'instruction de création de fonction, **CREATE FUNCTION**, on indique simplement que la fonction retourne une donnée de type table avec la clause **RETURNS TABLE**. Par la suite, après la clause **AS**, on précise quelle instruction doit être retournée dans la valeur de retour de type table de la fonction.

Exemple :

```
CREATE FUNCTION Commande (@Id int)  
RETURNS TABLE  
AS  
RETURN (SELECT * FROM Stock WHERE Id_Stock = @Id)
```

- Fonction scalaire :

```
CREATE FUNCTION Nom_Fonction (Liste de Paramètres) Returns Type  
As Begin  
--Instructions SQL
```

Return Valeur

End

Dans ce cas-là, après l'instruction de création de fonction, **CREATE FUNCTION**, suivie du nom de la fonction à utiliser et des paramètres à prendre en compte entre parenthèses. On indique simplement que la fonction retourne une valeur de scalaire avec la clause **RETURNS Type**. Les clauses AS BEGIN et END déclarent dans l'ordre, le début et la fin de la fonction. Par la suite, après la clause **AS begin**, on précise quelle valeur doit être retournée la fonction.

Exemple :

```
CREATE FUNCTION QTstock(@Id int) RETURNS int
```

```
AS BEGIN
```

```
    DECLARE @qt int
```

```
    Set @qt=(select quantité from article where ID=@id)
```

```
    RETURN @qt
```

```
END
```

Cette fonction retourne une valeur qui désigne la quantité en stock, pour un article passé en paramètre.

3- Appel :

Pour appeler une fonction, il faut utiliser son nom à deux composantes (dbo.name), il suffit de l'utiliser tel quel :

➤ **Pour fonction table :**

```
select * from tp1(3)
```

➤ **pour fonction scalaire**

```
PRINT dbo.tp(3)
```