

# Les Fonctions de Domaine dans Access

par Philippe JOCHMANS ([home page de Starec](#))

Date de publication : 12/10/2007

Dernière mise à jour : 12/10/2007

Ce tutorial a pour but de vous expliquer l'utilisation des fonctions de domaine dans Access.

I - Introduction.....	3
I-A - Qu'est-ce-que c'est ?.....	3
I-B - Quand les utiliser ?.....	3
I-C - Objectif.....	3
II - Généralités.....	4
II-A - Liste des Fonctions.....	4
II-B - Particularités dans les Formulaire et les Etats.....	4
II-C - Syntaxe.....	5
II-D - Exemples.....	6
III - Les Fonctions.....	7
III-A - DMin() ou MinDom() et DMax ou MaxDom().....	7
III-A-1 - Définition.....	7
III-A-2 - Formulaire et Etats.....	8
III-A-3 - VBA.....	8
III-A-4 - SQL.....	8
III-B - DSum() ou SomDom().....	9
III-B-1 - Définition.....	9
III-B-2 - Formulaire et Etats.....	9
III-B-3 - VBA.....	10
III-B-4 - SQL.....	10
III-C - DCount() ou CpteDom().....	10
III-C-1 - Définition.....	10
III-C-2 - Formulaire et Etats.....	11
III-C-3 - VBA.....	11
III-C-4 - SQL.....	11
III-D - DAvg() ou MoyDom().....	12
III-D-1 - Définition.....	12
III-D-2 - Formulaire et Etats.....	12
III-D-3 - VBA.....	12
III-D-4 - SQL.....	12
III-E - DLookup() RechDom().....	13
III-E-1 - Définition.....	13
III-E-2 - Formulaire et Etats.....	13
III-E-3 - VBA.....	14
III-E-4 - SQL.....	14
III-F - 3.6. DFirst() ou PremDom() et DLast() ou DernDom().....	14
III-F-1 - Définition.....	14
III-G - DStDev() ou ÉcartTypeDom() et DstDevP() ou ÉcartTypePDom().....	16
III-G-1 - Définition.....	16
III-G-2 - Formulaire et Etats.....	16
III-G-3 - VBA.....	16
III-G-4 - SQL.....	16
III-H - DVar() ou VarDom() et DVarP() ou VarPDom().....	17
III-H-1 - Définition.....	17
III-H-2 - Formulaire et Etats.....	17
III-H-3 - VBA.....	17
III-H-4 - SQL.....	17
IV - REMARQUES.....	18
V - TRUCS ET ASTUCES.....	19
VI - CONCLUSION.....	20
VII - REMERCIEMENTS.....	20

## I - Introduction

### I-A - Qu'est-ce-que c'est ?

Les fonctions de domaine (ou d'agrégation) ont le même objectif que des requêtes de regroupement (ou d'agrégation), sauf que celles-ci permettent de résoudre de nombreux problèmes dans certaines situations

Elles permettent d'effectuer quelques calculs statistiques (somme, moyenne, minimum, maximum, ...).

### I-B - Quand les utiliser ?

Je vous ai dit qu'elles avaient le même effet que des requêtes de regroupement, alors pourquoi les utiliser ? En fait ces fonctions sont utiles dans les cas suivants :

- Vous voulez afficher des informations dans un formulaire ou un état, hors celles-ci viennent d'une autre table ne faisant pas partie de la source de ce formulaire ou de cet état.
- Vous voulez vérifier des informations particulières sur une requête ou une table (quantité, etc..) avant de lancer des calculs.

### I-C - Objectif

L'objectif de ce tutorial est de vous expliquer le fonctionnement de ces fonctions, quel que soit votre niveau sur Access, en effet nous allons aborder chaque fonction dans trois situations différentes.

- La saisie directe dans un champ indépendant d'un formulaire ou d'un état, ce qui vous permet d'utiliser ces fonctions, même si vous ne connaissez pas le VBA
- L'utilisation dans du code VBA.
- L'utilisation de l'équivalent de ces fonctions en SQL (la lecture de cet [Article](#) sur le SQL est recommandée si vous ne connaissez pas ce langage).

## II - Généralités

### II-A - Liste des Fonctions

Nom Anglais	Nom Français	Description
DMin()	MinDom()	Cette fonction va permettre de trouver la valeur minimale d'un jeu d'enregistrements
DMax()	MaxDom()	Cette fonction va permettre de trouver la valeur maximale d'un jeu d'enregistrements
DSum()	SomDom()	Cette fonction permet d'effectuer la somme de valeurs contenues dans des enregistrements
DCount()	CpteDom()	Cette fonction permet de compter le nombre d'enregistrements contenus dans une table ou une requête
DAvg()	MoyDom()	Cette fonction va permettre de calculer la moyenne d'un ensemble de valeurs dans un jeu d'enregistrements
DLookup()	RechDom()	Cette fonction va permettre de rechercher la première valeur d'un champ dans un jeu d'enregistrements
DFirst()	PremDom()	Cette fonction va permettre de trouver la première valeur d'un champ d'un jeu d'enregistrements correspondant à un critère
DLast()	DernDom()	Cette fonction va permettre de trouver la dernière valeur d'un champ d'un jeu d'enregistrements correspondant à un critère
DVar()	VarDom()	Cette fonction va permettre de calculer la variance d'un champ
DVarP()	VarPDom()	Cette fonction va permettre de calculer la variance d'un champ sur l'ensemble de la population
DStDev()	ÉcartTypeDom()	Cette fonction va permettre de calculer l'écart type d'un champ
DstDevP()	ÉcartTypePDom()	Cette fonction va permettre de calculer l'écart type sur l'ensemble de la population

Le nom Anglais de chaque fonction sera utilisé dans le code VBA, le nom Français dans les Formulaires et les Etats. Dans les exemples, vous trouverez dans le sous-chapitre SQL de chaque fonction comment celles-ci sont nommées.

### II-B - Particularités dans les Formulaires et les Etats

Dans les formulaires ou dans les états, les fonctions de domaine ne peuvent se trouver que dans des contrôles zone de texte indépendants (l'utilisation dans les listes est possible mais inutile, car les fonctions de domaine ne renvoient qu'une valeur). Cela est valable pour toutes les fonctions de domaine.

Vous pouvez utiliser ces fonctions pour remplir la propriété **Caption** d'un label (étiquette), mais vous ne pouvez le faire que par programmation.

Pour cela placez un contrôle zone de texte dans votre formulaire ou votre état; vous avez trois possibilités pour saisir la fonction de domaine (Si vous renseignez le nom en Anglais, normalement Access le convertit dans la langue de votre version d'Access).

**Vous pouvez saisir directement le code dans la zone de texte quand votre formulaire est en mode création.**

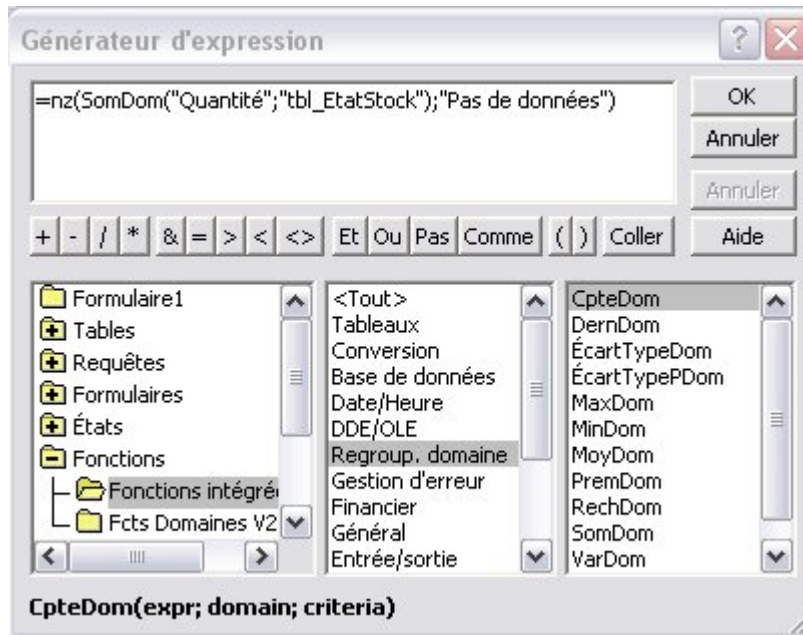
```
=nz(SomDom("Quantité";"tbl_EtatStock");"Pas de données")
```

Dans les propriétés de la zone de texte, onglet Données, dans Sources Contrôles saisir la fonction

```
Source contrôle ..... =nz(SomDom("Quantité";"tbl_EtatStock");"f
```

**Vous avez également la possibilité d'utiliser le générateur d'expression**

Pour cela cliquez sur les trois points qui sont au bout de la ligne Source Contrôle pour ouvrir le générateur d'expression.



Dans la première fenêtre sélectionnez : Fonctions Intégrées  
 Dans la deuxième fenêtre sélectionnez : Regroup. Domaine  
 Dans la troisième fenêtre vous trouverez les fonctions, cliquez sur une des fonctions pour la mettre dans la fenêtre du haut et remplissez les paramètres.

## II-C - Syntaxe

Les fonctions de domaine ont toutes la même syntaxe.

### **LaFonctionDeDomaine(expression ;Domaine ; critère)**

#### 1 - Expression :

Il s'agit de la valeur à traiter, en général cette valeur correspond au nom de la table ou de la requête où se trouvent les données. Cette expression étant de type texte, elle est à mettre entre guillemets et il faut rajouter des crochets au cas où celle-ci comporte des espaces ou des caractères spéciaux.

"Le champ" : ce n'est pas bon

"[Le champ]": c'est correct

Le fait de respecter les conventions de nommage vous évitera bien des soucis , je vous incite donc à lire cet article d'Argyronet :

## Descriptif des conventions typographiques du code Visual Basic

### 2 - Domaine :

Il s'agit du nom de la table où de la requête ou se trouvent les données à traiter, comme pour le paramètre Expression, il faut entourer le Domaine de guillemets et mettre les crochets si nécessaires.

### 3 - Critère :

Le critère va permettre de déterminer sur quels champs (expression) de la table (ou requête) s'applique la fonction. Il s'agit tout simplement de la clause Where d'une instruction SQL sans le mot Where. Ce critère n'est pas obligatoire, en effet s'il est absent, la fonction traitera tous les enregistrements.

Je vous conseille la lecture de ces deux tutoriaux de Caféine :

Pour le débogage


<http://cafeine.developpez.com/access/tutoriel/debugprint/>

Pour un exemple d'utilisation des fonctions de domaine au niveau du chapitre VI où vous verrez l'utilisation de trois fonctions :

<http://cafeine.developpez.com/access/tutoriel/recherchemulti/>

Vous trouverez un autre exemple dans ce tuto de Loufab sur la recherche multicritères avec la fonction DMax au chapitre VI-B.

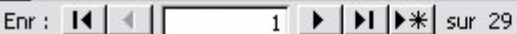
<http://loufab.developpez.com/recursivite3/>

 *Le séparateur d'arguments est le point virgule dans la syntaxe française (Formulaires et Etats), et la virgule dans la syntaxe Anglaise (VBA).*

## II-D - Exemples

Nous allons pour les exemples prendre les données d'une table qui contient les notes des élèves d'une classe. Ne prenez pas compte de la structure de la table, c'est uniquement pour avoir des données.

tbl_Classe : Table						
	Id	strNom	strPrenom	strMatiere	bytNote	dteNote
▶	1	Dupont	Martin	Français	15	15/09/2007
	2	Dupont	Martine	Mathématiques	12	15/09/2007
	3	Durant	Alexandre	Anglais	12	16/09/2007
	4	Durant	Alexandra	Français	15	17/09/2007
	5	Durant	Alexandra	Mathématiques	12	18/09/2007
	6	Durant	Alexandra	Français	10	19/09/2007
	7	Durant	Alexandra	Français	20	20/09/2007
	8	Dupond	Marc	Histoire	18	22/09/2007
	9	Dupond	Marc	Histoire	5	21/09/2007
	10	Lefevre	Henri	Mathématiques	15	21/09/2007
	13	Lefevre	Henri	Français	2	03/09/2007
	14	Lefevre	Henri	Français	12	04/09/2007
	15	Lefevre	Henri	Mathématiques	5	04/09/2007
	16	Lefevre	Henri	Mathématiques	2	05/09/2007
	17	Lefevre	Henri	Histoire	6	05/09/2007
	18	Lefevre	Henri	Histoire	12	06/09/2007
	19	Dupond	Marc	mathématiques	18	06/09/2007
	20	Dupond	Marc	mathématiques	18	06/09/2007
	21	Dupond	Marc	Français	15	01/09/2007
	22	Dupond	Marc	Français	16	02/09/2007
	23	Dupond	Marc	Histoire	15	23/07/2007
	24	Dupont	Martin	Français	0	14/09/2007
	26	Dupond	Marc	Histoire	20	30/09/2007
	27	Dupond	Marc	Français	10	03/09/2007
	28	Dupond	Marc	Français	5	04/09/2007

Enr :  1 sur 29

### III - Les Fonctions

#### III-A - DMin() ou MinDom() et DMax ou MaxDom()

Nous allons regrouper dans ce chapitre ces deux fonctions, en effet leur principe est le même et elles vont de paires, car l'une est l'opposé de l'autre.

##### III-A-1 - Définition

- La fonction **DMin** permet de déterminer les valeurs minimales d'un jeu d'enregistrements (définition de l'aide Access).
- La fonction **DMax** permet de déterminer les valeurs maximales d'un jeu d'enregistrements (définition de l'aide Access).

Ces deux fonctions renvoient une donnée de type Variant, en effet en cas de non correspondance (ex : le champ ne fait pas partie du domaine), la fonction renvoie une valeur Null.


### III-A-2 - Formulaires et Etats

Comme exemple, nous voulons avoir la note minimum et la note maximum de l'élève Dupond Marc en Français.

```
=NZ(MinDom("bytNote";"tbl_Classe";"strNom='Dupond' and strPrenom='Marc' and strMatiere='Français');"Pas de Notes")
```

```
=NZ(MaxDom("bytNote";"tbl_Classe";"strNom='Dupond' and strPrenom='Marc' and strMatiere='Français');"Pas de Notes")
```

Ce qui nous donne : 0 et 16.

 *J'ai utilisé la fonction **NZ**, car en cas d'erreur, la fonction renvoie une valeur nulle, cela permet d'afficher dans la zone de texte une autre information au lieu d'avoir : #Erreur.*

Notez que nous passons du texte dans le critère, il faut donc utiliser les simples quotes ('). A la place des simples quotes vous pouvez utiliser deux fois des doubles quotes, ainsi

'Dupond' devient ""Dupond"", mais personnellement je préfère la simple quote, cela rend le code plus clair.

Comme vous le voyez, la syntaxe est identique à celle que l'on applique dans la clause Where d'une requête SQL.

### III-A-3 - VBA

Nous allons prendre le même exemple pour réaliser ces fonctions en VBA, cependant nous allons un peu modifier les choses, en passant des paramètres à une fonction.

Celle-ci n'est pas faite pour être opérationnelle, mais pour montrer que l'on peut utiliser des variables (cela peut aussi être les contrôles d'un Formulaire) comme paramètres de cette fonction.

```
Public Function CalculMin(strExpression As String, strDomaine As String, Optional strCritere As String) As Variant
    CalculMin = DMin(strExpression, strDomaine, strCritere)
End Function
```

Il suffit de faire appel à cette fonction de la manière suivante :

```
CalculMin("bytNote", "tbl_Classe", "strNom='Dupond' and strPrenom='Marc' and strMatiere='Français'")
```

Vous pouvez faire la même en remplaçant DMin par DMax.

Je n'ai pas utilisé la fonction **NZ**, car comme la fonction est déclarée comme variant, elle renvoie tout simplement Null sans générer d'erreur, c'est au reste de l'application de gérer ce null.

### III-A-4 - SQL

Voici une fonction qui permet de calculer la note minimum. Pour que celle-ci fonctionne n'oubliez pas d'utiliser la référence Microsoft DAO 3.6 Object Library.

```
Public Function CalculMinSQL(strNomEleve As String, strPrenomEleve As String, strMatiereEleve As String) As Variant
```

```

' ===== déclaration des variables =====
Dim strSql As String
Dim rst As DAO.Recordset

' ===== affectation =====
strSql = "SELECT Min(bytNote) AS MinDeNote " & _
        "FROM tbl_Classe " & _
        "GROUP BY strNom, strPrenom, strMatiere " & _
        "HAVING strNom='" & strNomEleve & "' AND strPrenom='" & strPrenomEleve & "' " & _
        " AND strMatiere='" & strMatiereEleve & "';"
Set rst = CurrentDb.OpenRecordset(strSql)
If rst.RecordCount = 0 Then
    CalculMinSQL = "Pas de Note"
Else
    CalculMinSQL = rst("MinDeNote")
End If

' ===== libération =====
rst.Close
Set rst = Nothing

End Function
    
```

Pour avoir la note maximum, vous devez remplacer tous les Min par des Max.

Comme vous pouvez le voir la clause Having (qui correspond au Where pour les requêtes de regroupement) a la même syntaxe que l'expression en Français MinDom.

Contrairement aux fonctions DMin et DMax, on teste le nombre d'enregistrements récupérés avec If rst.RecordCount = 0, et suivant la réponse on agit en conséquence.

### III-B - DSum() ou SomDom()

Nous n'allons pas prendre notre classe d'élèves pour cet exemple, mais nous avons une table de stockage nommée tbl\_EtatStock, et un champ Qte pour les quantités.

#### III-B-1 - Définition

La fonction **DSum** calcule la somme d'un ensemble de valeurs relatives à un jeu d'enregistrements (Définition de l'aide Access).

Cette fonction renvoie une donnée de type Variant, en effet en cas de non correspondance (ex : le champ ne fait pas partie du domaine), la fonction renvoie une valeur Null.

#### III-B-2 - Formulaires et Etats

```
=NZ(SomDom("Qte";"tbl_EtatStock");"Pas de données")
```

J'ai utilisé la fonction **NZ**, car en cas d'erreur, la fonction renvoie une valeur nulle, cela permet d'afficher dans la zone de texte une autre information au lieu d'avoir : #Erreur.

Comme vous pouvez le voir, il n'y a pas de critères, cela montre bien que celui-ci est facultatif, donc en cas d'absence de critère le traitement se fait sur tous les enregistrements.

### III-B-3 - VBA

```
DSum("Qte", "tbl_EtatStock")
```

Il n'y a pas grand-chose à dire, à part penser à faire un contrôle de nullité.

### III-B-4 - SQL

```
Public Function CalculSumSQL(strChamp As String, strTable As String) As Variant

    On Error GoTo GestError
    ' ===== déclaration des variables =====
    Dim strSql As String
    Dim rst As DAO.Recordset

    ' ===== affectation =====
    strSql = "SELECT Sum(" & strChamp & ") AS Somme " & _
            "FROM " & strTable & ";"
    Set rst = CurrentDb.OpenRecordset(strSql)
    If rst.RecordCount = 0 Then
        CalculSumSQL = "Pas de Données"
    Else
        CalculSumSQL = rst("Somme")
    End If

    ' ===== libération =====
    rst.Close
    Set rst = Nothing
    Exit Function

GestError:

    If Err.Number = 3061 Then
        MsgBox "Une des donnée entrée en paramètre est erronée", vbOKOnly + vbExclamation
    End If

End Function
```

Cette fonction accepte en paramètre le nom du champ, et celui de la table.

J'ai rajouté une gestion d'erreur car en cas de mauvaise saisie des données en entrée la chaîne SQL ne pourra être interprétée.

### III-C - DCount() ou CpteDom()

#### III-C-1 - Définition

La fonction **DCount** détermine le nombre d'enregistrements contenus dans un jeu d'enregistrements (Définition de l'aide Access).

Cette fonction renvoie une donnée de type Variant, en effet en cas de non correspondance (ex : le champ ne fait pas partie du domaine), la fonction renvoie une valeur Null.

Lorsque vous utilisez cette fonction, assurez vous que le champ ne contient pas de valeurs nulles, car celles-ci seront ignorées.

### III-C-2 - Formulaires et Etats

Nous allons compter le nombre de notes inférieures à 10 de l'élève Lefevre Henri par rapport à toutes ces notes.

```
=CpteDom("bytNote";"tbl_Classe";"strNom='Lefevre' and strPrenom='Henri' and bytNote<10") & "/" &
CpteDom("bytNote";"tbl_Classe";"strNom='Lefevre' and strPrenom='Henri'")
```

Ici on a utilisé deux fois la fonction DCount, d'abord pour compter le nombre de notes inférieures à 10, ensuite pour calculer le nombre total de notes, ce qui donne à l'affichage 4/7.

Vous pouvez remarquer que le chiffre n'est pas entouré de quotes ("), cela est réservé au texte.

### III-C-3 - VBA

```
DCount("bytNote", "tbl_Classe", "strNom='Lefevre' and strPrenom='Henri' and bytNote<10") & _
"/" & DCount("bytNote", "tbl_Classe", "strNom='Lefevre' and strPrenom='Henri'")
```

### III-C-4 - SQL

Voici la même fonction en SQL

```
Public Function NoteInfMoyenne(strNomEleve As String, strPrenomEleve As String) As Variant

    On Error GoTo GestError
    ' ===== déclaration des variables =====
    Dim strSql As String
    Dim rst As DAO.Recordset

    ' ===== affectation =====
    strSql = "SELECT Count(bytNote) AS InferMoy " & _
            "FROM tbl_Classe " & _
            "WHERE bytNote < 10 " & _
            "GROUP BY strNom, strPrenom " & _
            "HAVING strNom='" & strNomEleve & "' And strPrenom='" & strPrenomEleve & "' " & _
            "UNION SELECT Count(bytNote) AS NbrNote " & _
            "FROM tbl_Classe " & _
            "GROUP BY strNom, strPrenom " & _
            "HAVING strNom='" & strNomEleve & "' AND strPrenom='" & strPrenomEleve & "';"
    Set rst = CurrentDb.OpenRecordset(strSql)
    If rst.RecordCount = 0 Then
        NoteInfMoyenne = "Pas de Données"
    Else
        NoteInfMoyenne = rst("InferMoy")
        rst.MoveNext
        NoteInfMoyenne = NoteInfMoyenne & "/" & rst("NbrNote")
    End If

    ' ===== libération =====
    rst.Close
    Set rst = Nothing
    Exit Function

GestError:

    If Err.Number = 3061 Then
        MsgBox "Une des donnée entrée en paramètre est erronée", vbOKOnly + vbExclamation
    Else
        MsgBox Err.Number & " - " & Err.Description
    End If
```

## End Function

Il suffit de passer en paramètres le nom et le prénom de l'élève.

## III-D - DAvg() ou MoyDom()

### III-D-1 - Définition

La fonction **DAvg** calcule la moyenne d'un ensemble de valeurs dans un jeu d'enregistrements (définition de l'aide Access).

Cette fonction renvoie une donnée de type Variant, en effet en cas de non correspondance (ex : le champ ne fait pas partie du domaine), la fonction renvoie une valeur Null.

Lorsque vous utilisez cette fonction, assurez vous que le champ ne contient pas de valeurs nulles, car celles-ci seront ignorées.

### III-D-2 - Formulaires et Etats

Nous allons établir la moyenne de l'élève Dupond Marc du 01 au 15 septembre 2007 inclus.

```
=NZ(MoyDom("bytNote";"tbl_Classe";"strNom='Dupond' and strPrenom='Marc' and dteNote Between #09/01/2007# and # 09/15/2007#");"Pas de Notes")
```

Nous avons incorporé une nouvelle notion dans nos critères, c'est la date. Vous noterez :

- La date doit être entourée de #
- La date est au format US (mm/dd/yyyy), ce qui est un piège, car lorsque vous utilisez le QBE pour vos requêtes vous devez la saisir en Français.

### III-D-3 - VBA

```
DAvg("bytNote", "tbl_Classe", "strNom='Dupond' and strPrenom='Marc' " & _
    "and dteNote Between #09/01/2007# and # 09/15/2007#"), "Pas de Notes")
```

### III-D-4 - SQL

Voici la même fonction en SQL

```
Public Function MoyenneQuinzaine(strNomEleve As String, strPrenomEleve As String) As Variant

    On Error GoTo GestError
    ' ===== déclaration des variables =====
    Dim strSql As String
    Dim rst As DAO.Recordset

    ' ===== affectation =====
    strSql = "SELECT strNom, strPrenom, Avg(bytNote) AS MoyenneDeNote " & _
        "FROM tbl_Classe " & _
        "WHERE dteNote Between #9/1/2007# And #9/15/2007# " & _
        "GROUP BY strNom, strPrenom " & _
        "HAVING strNom='" & strNomEleve & "' AND strPrenom='" & strPrenomEleve & "';"
```

```

Set rst = CurrentDb.OpenRecordset(strSql)
If rst.RecordCount = 0 Then
    MoyenneQuinzaine = "Pas de Données"
Else
    MoyenneQuinzaine = rst("MoyenneDeNote")
End If

' ===== libération =====
rst.Close
Set rst = Nothing
Exit Function

GestError:

If Err.Number = 3061 Then
    MsgBox "Une des donnée entrée en paramètre est erronée", vbOKOnly + vbExclamation
Else
    MsgBox Err.Number & " - " & Err.Description
End If

End Function

```

### III-E - DLookup() RechDom()

Cette fonction est la seule qui ne fait pas de calculs, il s'agit d'une fonction de recherche.

#### III-E-1 - Définition

La fonction **DLookup** recherche la valeur d'un champ particulier qui fait partie d'un jeu d'enregistrements (Définition de l'aide Access).

Cette fonction renvoie une donnée de type Variant, en effet en cas de non correspondance (ex : le champ ne fait pas partie du domaine), la fonction renvoie une valeur Null.

Attention, la fonction DLookup va rechercher le premier enregistrement correspondant aux critères, hors si vous avez plusieurs résultats possible vous n'en verrez qu'un, le premier trouvé.



*Si vous voulez recevoir plusieurs données en retour, passez par la méthode DAO et les recordset, pour cela je vous conseille la lecture de cet [article](#) de Christophe Warin sur la manipulation des données avec DAO.*

#### III-E-2 - Formulaires et Etats

Nous allons rechercher la date de la première note inférieure à la moyenne de l'élève Dupond Marc et différente de 0. Hors comme nous avons jusqu'à maintenant une table comme domaine, nous allons nous appuyer sur une requête comme domaine, ce qui nous permettra d'avoir les données triées par date.

Cette requête se nommera qry\_TriDate, dont voici la conversion SQL.

```

SELECT strNom, strPrenom, bytNote, dteNote
FROM tbl_Classe
ORDER BY dteNote;

```

Ce qui nous donne :

```
=RechDom("dteNote";"qry_TriDate";"strNom='Dupond' and strPrenom='Marc' and bytNote between 1 and 9")
```

Comme vous le voyez, le fonctionnement avec une requête est similaire.

Nous avons réutilisé l'instruction `between` (entre) qui a été utilisée sur les dates, nous voyons que les critères sont entre 1 et 9, car les bornes sont incluses, en mettant 0 à 10, cela ne correspondait pas à la demande.

### III-E-3 - VBA

```
DLookup("dteNote", "qry_TriDate", "strNom='Dupond' and strPrenom='Marc' and bytNote between 1 and 9")
```

### III-E-4 - SQL

Il n'y a pas d'équivalent pour cette fonction, car en fait il s'agit d'une fonction de recherche avec une instruction `Select` et les critères adéquats dans la clause `Where`.

### III-F - 3.6. DFirst() ou PremDom() et DLast() ou DernDom()

Nous allons regrouper dans ce chapitre ces deux fonctions, en effet leur principe est le même et elles vont de paires, car l'une est l'opposée de l'autre.

#### III-F-1 - Définition

La fonction **DFirst** renvoie un enregistrement aléatoire d'un champ particulier dans une table ou requête quand vous avez besoin d'une valeur quelconque de ce champ (Définition de l'aide Access).

La fonction **DLast** renvoie un enregistrement aléatoire d'un champ particulier dans une table ou requête quand vous avez besoin d'une valeur quelconque de ce champ (Définition de l'aide Access).

Comme vous le voyez ces définitions sont assez ambiguës. Il est conseillé d'utiliser ces fonctions à partir d'une requête qui donne déjà un ordre aux données, plutôt qu'une recherche sur une table.

Pour vous montrer l'ambiguïté de ces fonctions, on va prendre une requête dont voici le code SQL :

```
SELECT strNom, strPrenom, bytNote, dteNote
FROM tbl_Classe
WHERE strNom="Dupond" AND strPrenom="Marc"
ORDER BY dteNote;
```

Ce qui nous donne :

	strNom	strPrenom	bytNote	dteNote
▶	Dupond	Marc	15	23/07/2007
	Dupond	Marc	0	01/09/2007
	Dupond	Marc	15	01/09/2007
	Dupond	Marc	10	02/09/2007
	Dupond	Marc	16	02/09/2007
	Dupond	Marc	15	03/09/2007
	Dupond	Marc	10	03/09/2007
	Dupond	Marc	5	04/09/2007
	Dupond	Marc	0	05/09/2007
	Dupond	Marc	18	06/09/2007
	Dupond	Marc	18	06/09/2007
	Dupond	Marc	5	21/09/2007
	Dupond	Marc	18	22/09/2007
	Dupond	Marc	20	30/09/2007

Donc en théorie si nous recherchons la première note, nous devrions avoir 15 comme résultat, hors le code suivant nous donne 18.

```
=PremDom("bytNote";"qry_TriDate";"strNom='Dupond' and strPrenom='Marc'")
```

Et de même nous devrions théoriquement avoir 20 pour ce code hors il nous donne 15


```
=DernDom("bytNote";"qry_TriDate";"strNom='Dupond' and strPrenom='Marc'")
```

C'est pour cela que je vous conseille d'utiliser ce type de requête SQL si vous voulez vraiment avoir la première valeur de cette requête :

```
SELECT TOP 1 strNom, strPrenom, First(tbytNote) AS PremierDebytNote, dteNote
FROM tbl_Classe
GROUP BY strNom, strPrenom, dteNote
HAVING strNom="Dupond" AND strPrenom="Marc"
ORDER BY dteNote;
```

Et le code suivant pour le dernier en changeant l'OrderBy de sens :

```
SELECT TOP 1 strNom, strPrenom, First(tbytNote) AS PremierDebytNote, dteNote
FROM tbl_Classe
GROUP BY strNom, strPrenom, dteNote
HAVING strNom="Dupond" AND strPrenom="Marc"
ORDER BY dteNote DESC;
```

 *L'utilisation de ces deux fonctions est donc à éviter, voici un article du support Microsoft que m'a fourni Arkham46 sur la création de deux fonctions DStart et DEnd qui peuvent remplacer celles-ci.*

## Creating Functions Similar to DFirst() and DLast()

## III-G - DStDev() ou ÉcartTypeDom() et DstDevP() ou ÉcartTypePDom()

### III-G-1 - Définition

La fonction **DStDev** évalue l'écart type d'un ensemble de valeurs relatives à un jeu d'enregistrements spécifié (Définition de l'aide Access).

La fonction **DStDevP** évalue l'écart type d'un ensemble de valeurs relatives à un jeu d'enregistrements sur une population spécifié (Définition de l'aide Access)

Ces fonctions renvoient une donnée de type Variant, en effet en cas de non correspondance (ex : le champ ne fait pas partie du domaine), la fonction renvoie une valeur Null.


Lorsque vous utilisez ces fonctions, assurez-vous que le champ ne contient pas de valeurs nulles, car celles-ci seront ignorées.

Je vous renvoie vers vos cours de Statistiques pour la signification de ces fonctions, ce qui n'est pas l'objet de ce tutorial.

### III-G-2 - Formulaires et Etats

Nous allons calculer l'EcartType des notes de la classe.

```
=ÉcartTypeDom("bytNote";"tbl_Classe")
```

 Vous constatez que la première lettre est un é en majuscule, mais rassurez vous, si vous tapez l'instruction en minuscule avec le é, Access vous le convertira en É.

### III-G-3 - VBA

```
= DStDev("bytNote", "tbl_Classe")
```

### III-G-4 - SQL

```
Public Function EcartTypeClasse() As Variant

    On Error GoTo GestError
    ' ===== déclaration des variables =====
    Dim strSql As String
    Dim rst As DAO.Recordset

    ' ===== affectation =====
    strSql = "SELECT StDev(bytNote) AS EcartTypeDebytNote " & _
            "FROM tbl_Classe;"

    Set rst = CurrentDb.OpenRecordset(strSql)
    If rst.RecordCount = 0 Then
        EcartTypeClasse = "Pas de Données"
    Else
        EcartTypeClasse = rst("EcartTypeDebytNote")
    End If

    ' ===== libération =====
    rst.Close
    Set rst = Nothing
End Function
```

```
Exit Function

GestError:

If Err.Number = 3061 Then
    MsgBox "Une des donnée entrée en paramètre est erronée", vbOKOnly + vbExclamation
Else
    MsgBox Err.Number & " - " & Err.Description
End If

End Function
```

Remarque : Il n'y a pas d'équivalence de DstDevP() en SQL.

## III-H - DVar() ou VarDom() et DVarP() ou VarPDom()

### III-H-1 - Définition

La fonction **DVar** évalue la variance d'un ensemble de valeurs relatives à un jeu d'enregistrements spécifié (Définition de l'aide Access).

La fonction **DVarP** évalue la variance d'un ensemble de valeurs relatives à un jeu d'enregistrements sur une population spécifié (Définition de l'aide Access).

Ces fonctions renvoient une donnée de type Variant, en effet en cas de non correspondance (ex : le champ ne fait pas partie du domaine), la fonction renvoie une valeur Null.

Lorsque vous utilisez ces fonctions, assurez-vous que le champ ne contient pas de valeurs nulles, car celles-ci seront ignorées.

Je vous renvoie vers vos cours de Statistiques pour la signification de ces fonctions, ce qui n'est pas l'objet de ce tutorial.

### III-H-2 - Formulaires et Etats

Nous allons calculer la variance des notes de la classe.

```
=VarDom("bytNote";"tbl_Classe")
```

### III-H-3 - VBA

```
=DVar("bytNote", "tbl_Classe")
```

### III-H-4 - SQL

```
Public Function VarianceClasse() As Variant

    On Error GoTo GestError
    ' ===== déclaration des variables =====
    Dim strSql As String
    Dim rst As DAO.Recordset

    ' ===== affectation =====
    strSql = "SELECT Var(bytNote) AS VarDebytNote " & _
```

```

"FROM tbl_Classe;"

Set rst = CurrentDb.OpenRecordset(strSql)
If rst.RecordCount = 0 Then
    VarianceClasse = "Pas de Données"
Else
    VarianceClasse = rst("VarDebytNote")
End If

' ===== libération =====
rst.Close
Set rst = Nothing
Exit Function

GestError:

If Err.Number = 3061 Then
    MsgBox "Une des donnée entrée en paramètre est erronée", vbOKOnly + vbExclamation
Else
    MsgBox Err.Number & " - " & Err.Description
End If

End Function

```

Remarque : Il n'y a pas d'équivalence de DVarP en SQL.

## IV - REMARQUES

Voici quelques remarques à la suite de ce tutorial.

### **Fonctions de domaine ou SQL.**

Vous avez vu que l'écriture d'une fonction de domaine dans un formulaire est beaucoup plus simple que la fonction équivalente en SQL, alors pourquoi s'embêter avec le SQL ?

Tout simplement parce que le SQL est beaucoup plus rapide en traitement, malgré leur simplicité, les fonctions de domaine sont lentes. Quand vos tables sont volumineuses, passez par le SQL.

De plus il serait tentant de dire, puisque le SQL est plus rapide, je mets l'instruction directement dans la zone de texte de la manière suivante

```
= SELECT * FROM #.
```

Hors Access ne saura interpréter cette instruction, et vous affichera #Erreur dans la zone de texte.

### **Passage de variables**

Dans tous les exemples sur les formulaires et les Etats, j'ai écrit en dur le nom et le prénom de l'élève, mais dans la réalité il en va tout autrement, en effet il faut passer ces données par des variables ou utiliser le nom des zones de texte.

Par exemple dans un formulaire j'ai deux zones de texte (txtNom et txtPrenom) et une liste déroulante (cboMatière) qui me donne la table des matières.

L'instruction suivante :

```
=NZ(MinDom("bytNote";"tbl_Classe";"strNom='Dupond' and strPrenom='Marc' and strMatiere='Français');"Pas de Notes"
```

Devient :

```
=NZ(MinDom("[bytNote]";"[tbl_Classe]";"[strNom]='" & [txtNom] & "' and [txtPrenom]='" & [txtPrenom] & "' _ and [strMatiere]='" & [cboMatiere] & "'");"Pas de Notes")
```

## V - TRUCS ET ASTUCES

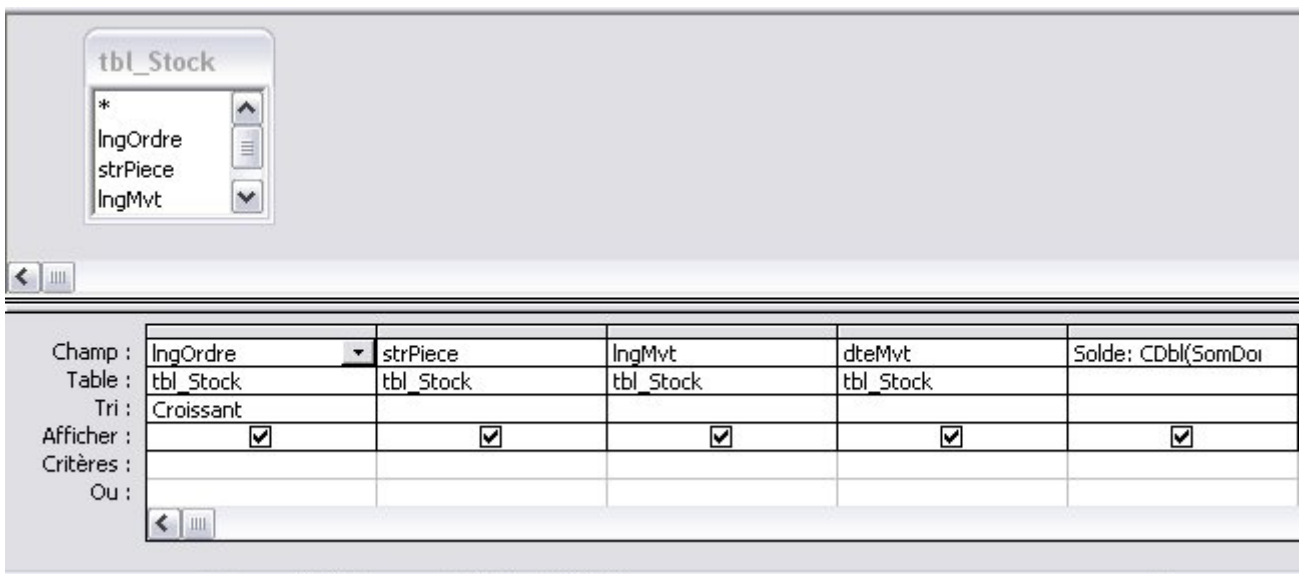
### Visualiser le cumul dans une requête

Certaines fonctions de domaine permettent d'autres utilisations que celles évoquées dans ce tutorial. Une utilisation de la fonction DSum va permettre d'afficher un cumul dans une requête, comme vous pouvez visualiser un solde dans vos comptes bancaires.

Nous allons utiliser une table qui reprend les mouvements de stock d'une pièce.

	IngOrdre	strPiece	IngMvt	dteMvt
	1	La Pièce	50	01/09/2007
	2	La Pièce	25	01/09/2007
	3	La Pièce	10	02/09/2007
	4	La Pièce	-25	03/09/2007
	5	La Pièce	-10	04/09/2007
	6	La Pièce	-55	05/09/2007
	7	La Pièce	25	06/09/2007
	8	La Pièce	125	07/09/2007
	9	La Pièce	-102	08/09/2007

Voici comment construire la requête avec le QBE.



Champ :	IngOrdre	strPiece	IngMvt	dteMvt	Solde: CDBl(SomDoi
Table :	tbl_Stock	tbl_Stock	tbl_Stock	tbl_Stock	
Tri :	Croissant				
Afficher :	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Critères :					
Ou :					

Et pour Solde nous écrivons

```
Solde: CDb1(SomDom("lngMvt";"tbl_Stock";"[lngOrdre]<=" & [lngOrdre]))
```

Le principe est d'afficher dans ce champ la somme des mouvements en prenant une référence (l'ordre), de manière à mettre dans le critère la somme des mouvements

Le résultat visuel est parlant

	lngOrdre	strPiece	lngMvt	dteMvt	Solde
▶	1	La Pièce	50	01/09/2007	50
	2	La Pièce	25	01/09/2007	75
	3	La Pièce	10	02/09/2007	85
	4	La Pièce	-25	03/09/2007	60
	5	La Pièce	-10	04/09/2007	50
	6	La Pièce	-55	05/09/2007	-5
	7	La Pièce	25	06/09/2007	20
	8	La Pièce	125	07/09/2007	145
	9	La Pièce	-102	08/09/2007	43

Vous pouvez extrapoler ce principe avec d'autres fonctions de domaine :

- Avec la fonction DCount vous pouvez faire un compteur.
- Avec la fonction DMoy vous pouvez voir une moyenne évoluer.

## VI - CONCLUSION

Les conclusions sur ce tutorial sont les suivantes, en posant le pour et le contre.

**Pour :**

- Syntaxe très simple.
- Permet de mettre des informations d'autres sources (tables ou requêtes) dans un formulaire ou un état ayant une autre source.
- L'avantage est de pouvoir faire de petits calculs statistiques sans entrer dans le VBA, c'est l'idéal pour les débutants.

**Contre :**

- Temps de traitement plus long que le passage avec DAO et SQL, cela est significatif sur de grosses bases.

## VII - REMERCIEMENTS

Je remercie toute l'équipe Office de Developpez.com pour leur relecture et leur soutien.

