

Document n°1 : le langage VBA

1. Variables et types, affectation, entrée et sortie d'information

Variables : une variable est définie par son identificateur (son nom) et son type (sa catégorie). Une variable ainsi définie désigne alors une case-mémoire. La valeur qui est mise dans cet emplacement est la valeur de la variable ; elle doit respecter les contraintes imposées par le type.

Types des variables : exemples :

Dim x As Integer x est de type Integer (nbs entiers de -32768 à 32767)
Dim y As Double y est de type Double (nbs décimaux)
Dim z As String z est de type String (chaîne de caractères)
Dim t As Boolean t est de type booléen et prend la valeur True ou False
Dim p As Object p est de type Object (p désigne un objet)
Dim u As Variant u peut prendre une valeur de n'importe quel type

Instruction affectation : elle est définie avec = Exemples :

x=2 x prend la valeur 2

z=x+5.3 z prend la valeur de 2+5.3 et z vaut 7.3

z=z+1 z qui a pour valeur 7.3 est augmenté de 1 et z vaut alors 8.3

Instruction entrée d'information : x=InputBox("...")

InputBox affiche une fenêtre (boîte de dialogue) et permet à l'utilisateur d'y mettre une valeur à partir du clavier. La valeur tapée est transmise par InputBox au sein de la case-mémoire désignée par x. Il s'agit d'une entrée d'information. "... " est une chaîne de caractères qui est affichée au sein de la boîte de dialogue. Exemple : x=InputBox("donne la valeur de x").

Instruction sortie d'information : MsgBox(y)

Elle assure la recopie de la valeur de la variable y de la mémoire centrale à l'écran. Il s'agit d'une sortie d'information. L'information s'affiche alors au sein d'une fenêtre (boîte de message). Cas particulier : MsgBox("Bonjour") affiche la chaîne de caractères "Bonjour".

Concaténation de chaînes de caractères : &

Si t1, t2 et t3 sont de type String et si t1="Bonjour. ", t2="Comment vas-tu?", on peut écrire :

t3=t1 & t2. L'opérateur & réalise la concaténation des deux chaînes de caractères.

t3 vaut alors : "Bonjour. Comment vas-tu?"

Opérateurs :

calcul numérique : +, -, *, /

calcul arithmétique : \ donne le résultat de la division entière : ex : 17 \ 3 donne 5

Mod donne le reste de la division entière : ex : 17 Mod 3 donne 2

Calculs :

res = Int(res*100)/100 => la valeur de res est tronquée avec 2 décimales

res = Int(res*100+0.5)/100 => la valeur de res est arrondie avec 2 décimales

Tirage au hasard : Rnd()

La fonction Rnd() donne un résultat numérique aléatoire compris entre 0 et 1, bornes exclues.

x = Rnd()*100 => x prend une valeur aléatoire décimale telle que 0<x<100

x = Int(Rnd()*100) => x prend une valeur aléatoire entière telle que 0<=x<100

x = Int(1+Rnd()*6) => x prend comme valeur : 1 ou 2 ou 3 ou 4 ou 5 ou 6 (lancer d'un dé)

Exemple 11 : deux variables x et y

```
Option Explicit
Sub exemple()
Dim x As Integer, y As Integer
x = InputBox("donne x : ")
y = x + 5
MsgBox(y)
MsgBox ("x = " & x & " y=" & y)      'affichage sur une seule ligne
MsgBox ("x = " & x & vbCr & "y=" & y)  'affichage sur deux lignes avec vbCr
End Sub
```

Exemple 12 : calcul de la moyenne de 3 nombres

```
Option Explicit
Sub moyenne()
Dim x As Integer, y As Integer, z As Integer
Dim m As Double
x = InputBox("donne x : ")
y = InputBox("donne y : ")
z = InputBox("donne z : ")
m=(x+y+z)/3
MsgBox ("la moyenne vaut " & m)
End Sub
```

Exemple 13 : tirage au hasard de 3 nombres

```
Option Explicit
Sub nbs()
Dim x As Integer, y As Integer, z As Integer
Randomize
x = Int(Rnd()*10)
y = Int(Rnd()*10)
z = Int(Rnd()*10)
MsgBox("Résultats:"& x & " " & y & " " & z)
End Sub
```

2. Instruction conditionnelle

```
Syntaxe :      If C Then
                  I1
                  I2
                  ... (autant d'instructions que nécessaire)
Else
                  J1
                  J2
                  ... (autant d'instructions que nécessaire)
End If
```

avec : C est une condition logique, vraie ou fausse
 I1, I2,..., J1, J2,... : des instructions
 une disposition des termes selon la présentation ci-dessus

Exécution : Si la condition C est vraie, alors exécution des instructions I1, I2,...
 Si la condition C est fausse, alors exécution des instructions J1, J2,...

Cas particulier : la partie "Else J1 J2 ..." peut être omise si elle n'est pas nécessaire.

Exemple 21 : un nombre tiré au hasard compris entre 0 et 100 est-il ou non supérieur à 50 ?

```
Option Explicit
Sub nbhasard()
Dim x As Integer
Randomize      'rend le tirage aléatoire
x = Int(Rnd() * 100)  'donne un nb : 0<=x<100
If x > 50 Then
  MsgBox (x & " > 50 : vous avez gagné!!")
Else
  MsgBox (x & " <= 50 : vous avez perdu!!")
End If
End Sub
```

Exemple 22 : le montant m d'un billet peut être réduit selon un taux de réduction t qui dépend de nb , nombre des enfants :

$nb=0 \Rightarrow t=0$
 $nb=1 \Rightarrow t=0.1$
 $nb=2 \Rightarrow t=0.2$
 $nb=3 \text{ ou plus } \Rightarrow t=0.3$

Calculer le montant à payer selon le nombre des enfants.

Option Explicit

Sub billet()

Dim nb As Integer

Dim m As Double, t As Double, res As Double

m = InputBox("donne le montant initial")

nb = InputBox("donne le nombre d'enfants")

If nb=0 Then

 t=0

End If

If nb=1 Then

 t=0.1

End If

If nb=2 Then

 t=0.2

End If

If nb>=3 Then

 t=0.3

End If

res = m*(1-t)

MsgBox("le montant à payer vaut " & res)

End Sub

Autre version :

les instructions "If C Then II End If"

peuvent être simplifiée car elles

ne comportent qu'une seule instruction II

(...)

If nb=0 Then t=0

If nb=1 Then t=0.1

If nb=2 Then t=0.2

If nb>=3 Then t=0.3

res = m*(1-t)

MsgBox("le montant à payer vaut " & res)

End Sub

Opérateurs : les conditions logiques sont structurées à l'aide d'opérateurs :

Opérateurs de **comparaison** : =, <>, <, <=, >, >=

Opérateurs logiques **And** et **Or** : les expressions logiques **A And B** et **A Or B** sont vraies ou fausses selon les cas suivants :

A	B	A And B	A Or B
Vraie	Vraie	Vraie	Vraie
Vraie	Fausse	Fausse	Vraie
Fausse	Vraie	Fausse	Vraie
Fausse	Fausse	Fausse	Fausse

Exemple 23 : Soit trois nombres tirés au hasard. Est-ce que deux nombres au moins sont égaux ?

Option Explicit

Sub nbsegalite()

Dim x As Integer, y As Integer, z As Integer

Randomize

x = Int(Rnd()*10) 'x est tel que : 0<=x<10

y = Int(Rnd()*10) 'y est tel que : 0<=y<10

z = Int(Rnd()*10) 'z est tel que : 0<=z<10

If x=y Or y=z Or z=x Then

 MsgBox("OK : " & x & " " & y & " " & z)

Else

 MsgBox("NON : " & x & " " & y & " " & z)

End If

End Sub

3. Instruction de répétition (ou boucle)

Syntaxe n°1 : **For** i = n1 **To** n2

```
    I1
    I2
    ... (autant d'instructions à répéter que nécessaire)
Next
```

avec : i est une variable de type Integer ; n1 et n2 : deux valeurs entières
I1, I2,... : des instructions

Exécution : Pour chaque valeur de i allant de n1 à n2 (bornes comprises), For...Next exécute les instructions I1, I2,...

Exemple : "For i=1 to 20 I1 I2 Next" répète 20 fois l'exécution de I1 et I2.

Exemple 31 : *Faire la somme de 20 nombres donnés par l'utilisateur*

```
Option Explicit
Sub som20nbs()
Dim x As Integer, s As Integer, i As Integer
s = 0
For i = 1 To 20
    x = InputBox("donne le nombre n°" & i)    'x prend une nouvelle valeur
    s = s + x                                'x est ajouté à s
Next
MsgBox ("la somme vaut " & s)
End Sub
```

Exemple 32 : *Calculer et afficher les 10 premiers termes d'une progression arithmétique, de premier terme 1 et de raison r donnée par l'utilisateur.*

```
Option Explicit
Sub arith()
Dim r As Integer, x As Integer, i As Integer
Dim res As String
r = InputBox("valeur de la raison : ")
x = 1                                'x vaut 1, valeur du premier terme
res = "1"
For i = 1 To 10
    x = x + r                        'r est ajoutée à x et x prend une nouvelle valeur
    res = res & " " & x              'la valeur de x est concaténée à res
Next
MsgBox (res)
End Sub
```

Autres syntaxes :

n°2 : **While** C
 I1
 I2
 (...)
Wend

n°3 : **Do**
 I1
 I2
 (...)
Loop Until C

Exécution :

Tant que la condition logique C reste vraie, alors exécution de I1, I2, ...
Dès que C devient fausse, fin de l'exécution de While...Wend, puis exécution de l'instruction suivante du programme.

Exécution :

Exécution de I1, I2, ... jusqu'à ce que la condition logique C soit vraie.
Dès que C est vraie, fin de l'exécution de Do...Loop Until, puis exécution de l'instruction suivante du programme.

Exemple 33 : *Faire la somme d'un nombre quelconque de nbs et arrêter cette somme en tapant -1.*

Option Explicit	Option Explicit
Sub somnbs()	Sub somnbs2()
Dim s As Integer, x As Integer	Dim s As Integer, x As Integer
s = 0	s = 0
x = InputBox("donne un nombre ")	x = 0
While x <> -1	Do
s = s + x	s = s + x
x = InputBox("donne un nombre ")	x = InputBox("donne un nombre ")
Wend	Loop Until x = -1
MsgBox ("La somme vaut " & s)	MsgBox ("La somme vaut " & s)
End Sub	End Sub

Double boucle :

Exemple 34 : *Afficher un tableau de 4 lignes et 5 colonnes où chaque élément est une étoile.*

```
Option Explicit
Sub tableau()
Dim i As Integer, j As Integer
Dim res As String
res = ""
For i = 1 To 4
    For j = 1 To 5
        res = res & "*"
    Next j
    res = res & vbCr 'ajout d'un retour à la ligne pour chaque valeur de i
Next i
MsgBox (res)
End Sub
```

4. Exemples de programmes

Exemple 41 : *20 nbs dont les valeurs sont comprises entre 0 et 99 sont tirés au hasard. Calculer la somme de ces 20 nbs. Si la somme est supérieure à 1100, c'est gagné !*

```
Option Explicit
Sub somsup()
Dim x As Integer, s As Integer
Dim i As Integer
Randomize
s = 0
For i = 1 To 20
    x = Int(Rnd() * 100) 'x est tel que : 0<=x<100
    s = s + x
Next
If s > 1100 Then
    MsgBox ("BRAVO, c'est gagné car s=" & s)
Else
    MsgBox ("ECHEC, car s=" & s)
End If
End Sub
```

Exemple 42 : *20 nbs dont les valeurs sont comprises entre 0 et 9 sont tirés au hasard. S'il y a plus de 7 nbs dont la valeur dépasse 6, c'est gagné !*

```
Option Explicit
Sub nbssup6()
Dim x As Integer, nb As Integer
Dim i As Integer
Dim res As String
```

'suite ->

```

Randomize
nb = 0
res = ""
For i = 1 To 20
    x = Int(Rnd() * 10)
    res = res & " " & x
    If x > 6 Then nb = nb + 1
Next
MsgBox (res)
If nb > 7 Then
    MsgBox ("BRAVO, c'est gagné car nb=" & nb)
Else
    MsgBox ("ECHEC, car nb=" & nb)
End If
End Sub

```

Exemple 43 : 5 nbs compris entre 0 et 9 sont tirés au hasard. Sont-ils tous supérieurs à 2 ?

```

Option Explicit
Sub nbssup()
Dim x As Integer, val As Integer
Dim i As Integer
Dim res As String
Randomize
val = 0 'exprime que tous les nbs sont a priori >2
res = ""
For i = 1 To 5
    x = Int(Rnd() * 10)
    res = res & " " & x
    If x <= 2 Then val = 1 'dès qu'une valeur de x est <=2, val prend la valeur 1
Next
If val = 0 Then
    MsgBox ("Tous les nbs sont >2 car : " & res)
Else
    MsgBox ("Tous les nbs ne sont pas >2 car : " & res)
End If
End Sub

```

5. Fonctions et procédures

Exemple 51 : Calculer $3!+5!+8!$ à l'aide d'une fonction *facto(x)*

Le code est constitué de la procédure calcul() et de la fonction facto(x). Cette fonction est définie avec une variable x de type Integer ; la valeur calculée par la fonction est de type Double.

Sub calcul()

```

Dim res As Double
res = facto(3) + facto(5) + facto(8) 'permet de calculer 3!+5!+8!
MsgBox ("le résultat vaut " & res)

```

End Sub

'variable de définition 'type de la valeur calculée par facto(x)

Function facto(x As Integer) As Double

```

Dim f As Double
Dim i As Integer 'f et i sont les variables locales à la fonction
f = 1
For i = 1 To x
    f = f * i
Next
facto = f 'facto "récupère" la valeur calculée f (écriture indispensable)

```

End Function

EXERCICES

Affectation, entrée et sortie d'information

1. Rédiger un programme qui lors de son exécution permet à l'utilisateur de donner son année de naissance, puis qui après avoir calculé l'âge de l'utilisateur affiche par exemple :
"Bravo! Vous avez 20 ans" si l'utilisateur est né en 1986.
2. Rédiger un programme qui lors de son exécution permet à l'utilisateur de donner son prénom et son année de naissance, puis qui après avoir calculé l'âge de l'utilisateur affiche par exemple :
"cher Joseph, vous avez 20 ans. Bonne chance!" (sur une seule ligne, puis sur 2 lignes)
3. Rédiger un programme qui demande à l'utilisateur de donner le prix unitaire d'un produit (type Double), le nombre de produits correspondants (type Integer) et le taux de TVA à appliquer sur le montant (type Double), puis qui calcule le montant final.
4. Soit x une valeur de type Double donnée par l'utilisateur. Calculer $f(x)=(1+x+x^2)/(2+x)$. Afficher le résultat.
5. Soit une durée en secondes donnée par l'utilisateur. Calculer puis afficher la décomposition en heures, minutes et secondes (utiliser les opérateurs \backslash et Mod).
6. Soit les coordonnées des points A et B. Calculer la longueur du segment AB. ($\text{Sqr}(x)$ calcule \sqrt{x})

Instructions conditionnelles

- 7- Soit les valeurs x et y , résultats du lancer de 2 dés. Tester si x et y sont égaux.
- 8- Soit une note donnée par l'utilisateur. Examiner 4 cas :
1^{er} cas : la note est inférieure ou égale à 7, elle reste inchangée ;
2^{ème} cas : la note est comprise entre 8 et 12, elle est augmentée de 2 points ;
3^{ème} cas : la note est comprise entre 13 et 17, elle est augmentée de 1 point ;
4^{ème} cas : la note est supérieure ou égale à 18, elle reste inchangée.
- 9- Soit m la moyenne des notes d'un étudiant donnée par l'utilisateur. Déterminer la décision du jury (ajourné, admis passable, admis AB, admis B, admis TB).
- 10- Un nombre entier compris entre 0 et 100 est tiré au hasard. Est-il compris entre deux bornes a et b données par l'utilisateur ?
- 11- Soit 3 nombres entiers tirés au hasard. Sont-ils tous des nombres pairs ?
- 12- Soit 3 nombres positifs x , y et z donnés par l'utilisateur. Déterminer si l'un des nombres est égal à la somme des deux autres (une seule réponse affichée « oui » ou « non »).
- 13- Soit deux nombres entiers donnés par l'utilisateur. Tester si l'un est le multiple de l'autre.
- 14- Soit 2 nombres x et y donnés par l'utilisateur, puis une lettre donnée par l'utilisateur. Si la lettre tapée est « p », le résultat calculé par le programme vaut le produit $x*y$; si la lettre tapée est « s », le résultat calculé vaut la somme $x+y$.
- 15- Deux nombres x et y compris entre 0 et 500 sont tirés au hasard. Sont-ils proches l'un de l'autre ? (Deux nombres sont proches si leur écart est inférieur à un seuil donné).

Répétitions

16- Calculer la somme des entiers de 1 à n (ne pas utiliser la formule $n*(n+1)/2$).

17- Soit un entier positif de type Integer donné par l'utilisateur. Calculer sa factorielle. Que se passe-t-il lorsque l'entier est « grand » ? Comment remédier au problème posé ?

18- Soit un entier positif n. Afficher sur n lignes successives les valeurs de k et de k ! pour $k=1, \dots, n$

19- Soit une progression arithmétique : son premier terme a, sa raison r et son nombre de termes n sont donnés par l'utilisateur. Calculer la somme des termes de la progression arithmétique.

20- Faire la somme de 10 nombres aléatoires : chaque nombre est le résultat d'un tirage au hasard, et la somme des nombres est calculée au fur et à mesure. Calculer ensuite la moyenne de ces nombres.

21- Faire la somme de plusieurs nombres positifs en arrêtant ce calcul lors de la saisie de -1. Calculer ensuite la moyenne de ces nombres.

22- On effectue des lancers successifs d'un dé (entier aléatoire compris entre 1 et 6) jusqu'à obtenir 6. Combien de lancers ont-ils été nécessaires ?

23- Résoudre $U_n = a*U_{n-1} + b*U_{n-2}$ pour a, b, U_0 , U_1 et n donnés par l'utilisateur.

Répétitions + instructions conditionnelles

24- Donner successivement 10 nombres. Quelle est la valeur du nombre le plus grand ?

25- 20 nombres entiers compris entre 1 et 100 sont tirés au hasard. Combien y a-t-il de nombres pairs ?

26- Donner successivement 10 caractères (type String). Quel est le nombre de voyelles ?

27- On effectue 10 lancers successifs d'un dé. Si le nombre de 6 obtenu à la suite du tirage est supérieur ou égal à 2, afficher "c'est gagné", sinon afficher "c'est perdu".

28- On effectue 10 lancers successifs de 2 dés : x est le résultat du premier dé, y celui du deuxième dé. Si on obtient au moins une fois $x=y$, afficher "c'est gagné", sinon afficher "c'est perdu".

29- Donner deux entiers n et p. Calculer la somme des nombres pairs compris dans l'intervalle $[n,p]$.

30- Donner un nombre entier positif. Est-il parfait ? (un nombre est parfait s'il est égal à la somme de ses diviseurs, que ses diviseurs soient premiers ou non, en incluant 1 comme diviseur, mais pas le nombre étudié. Exemples : 6 et 28 sont des nombres parfaits).

31- Quels sont les nombres parfaits parmi les 500 premiers nombres entiers positifs ? parmi les 2000, parmi les 10000 premiers nombres entiers positifs ?

32- Donner un nombre entier positif. Est-il premier ?

Fonctions

33- Calculer $n!$ pour n et p donnés par l'utilisateur à l'aide d'une fonction facto(x) qu'il convient de définir.

34- Calculer $x^4+x^7+x^{12}$ à l'aide d'une fonction puissance(x,n) qu'il convient de définir.

35- Calculer e^x pour x et n donnés en utilisant la formule