

VBA

COURS COMPLET

+PDF +QUIZ

Sommaire

2



Enregistrer le fichier

Enregistrer une macro

L'éditeur

Déclencher

Codes Simples

Commentaire

Écrire dans une cellule

Afficher un message

Lire une cellule

Fonctions de texte

Conditions

Message plusieurs boutons

Demander une information

Debugage

Messages d'erreur

Variables

Propriété

Boucles

Arrêts

Modifier ce qui s'affiche

Array

GOTO

Functions

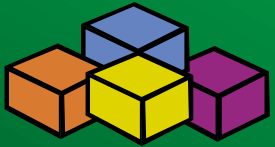
Système de fichiers

Protection

Mots clés

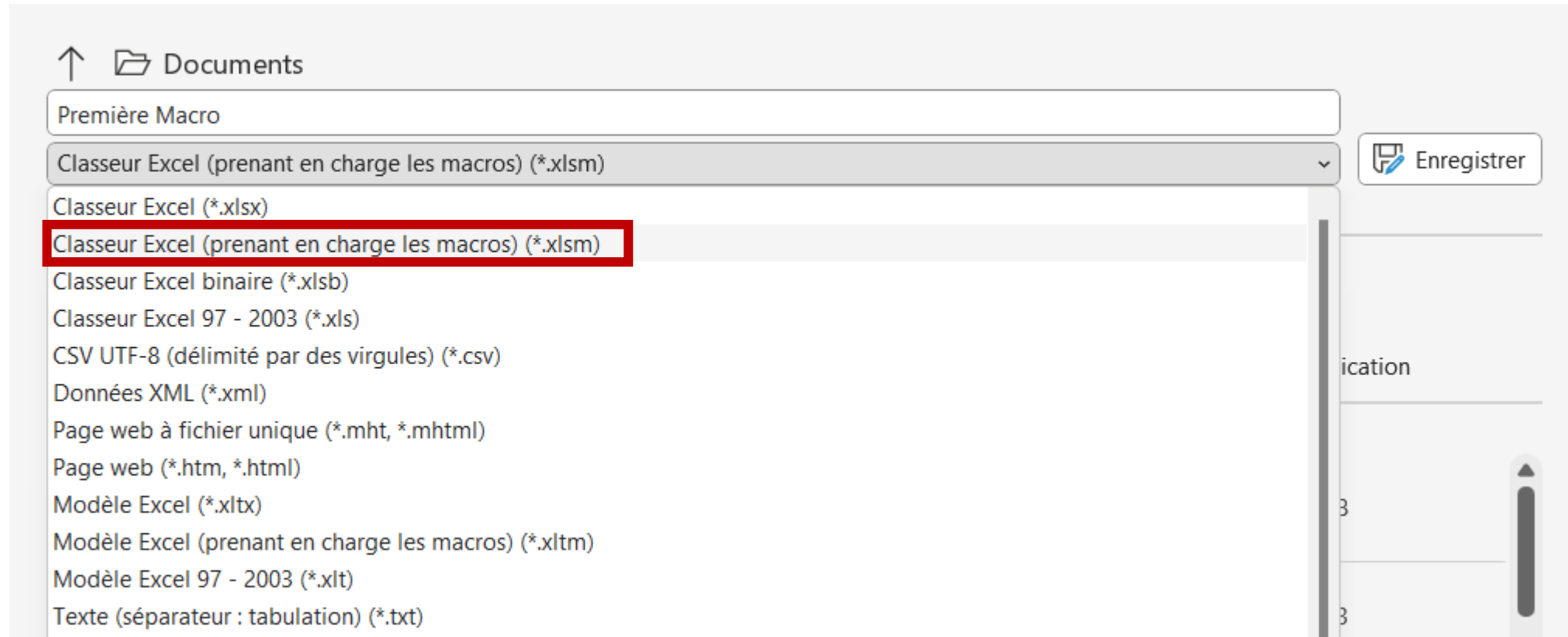
Formulaire

Classes et Structures
Programmation objet



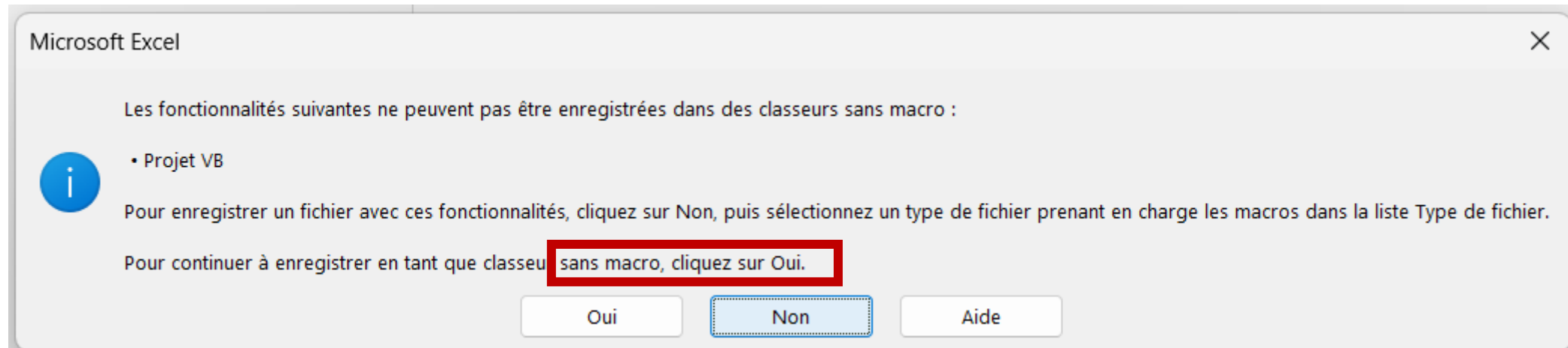
Enregistrer le fichier

VBA



Enregistrer SANS MACRO

5



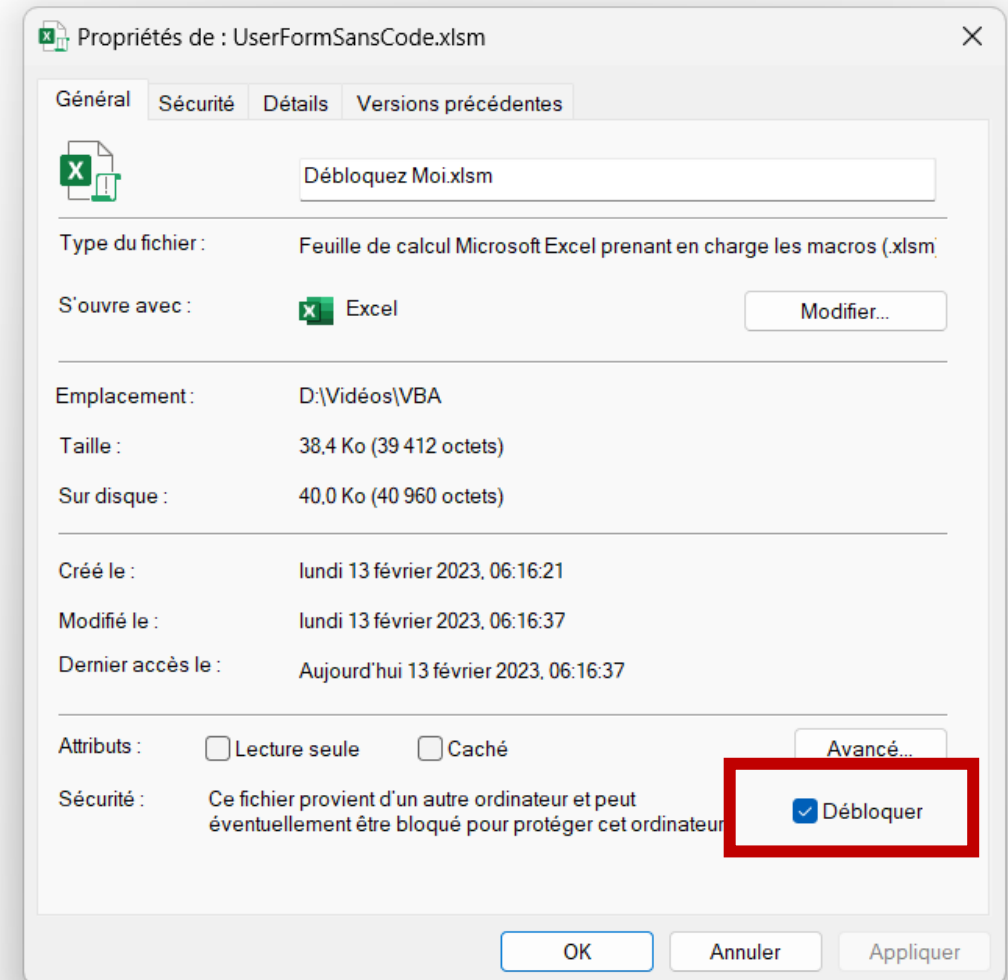
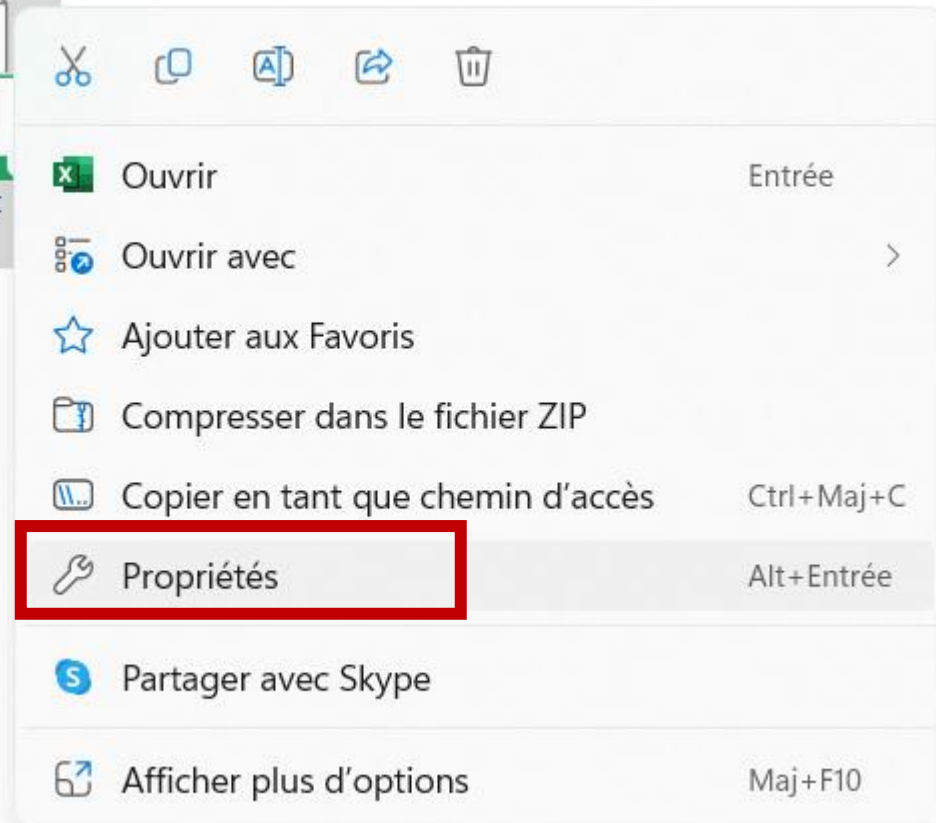
Débloquer un fichier téléchargé

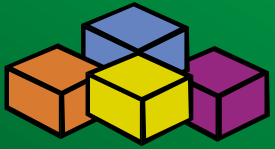
6



RISQUE DE SÉCURITÉ Microsoft a bloqué l'exécution des macros, car la source de ce fichier n'est pas approuvée.

En savoir plus



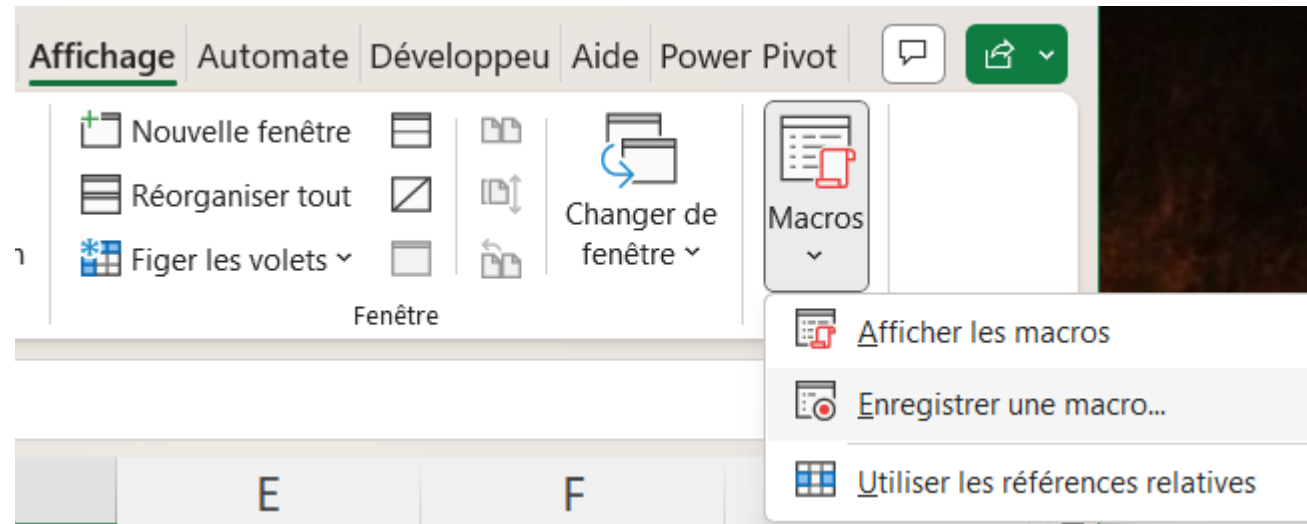


Enregistrer une macro

VBA

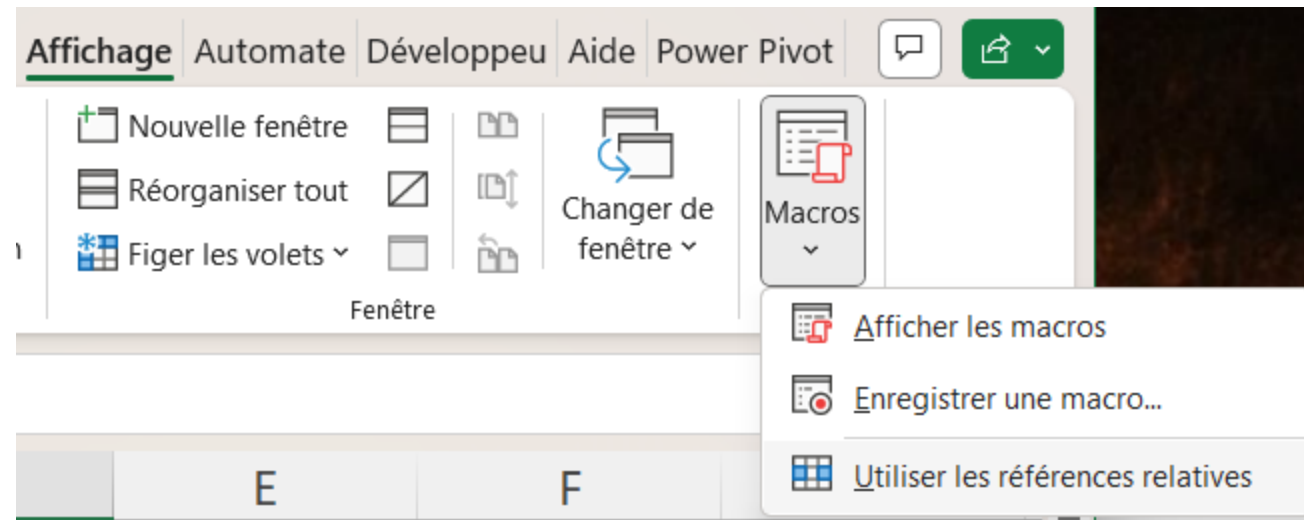
Onglet Affichage

8



Références relatives

9



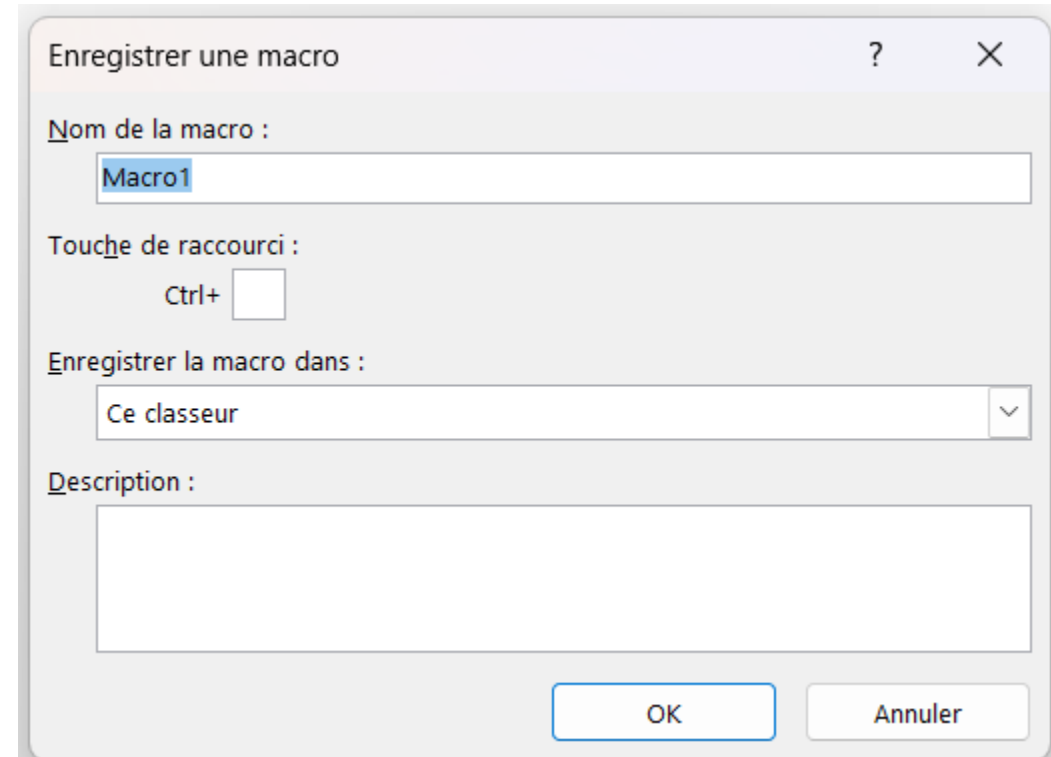
Nommer une macro

10

Pas d'espaces dans les noms

Bien choisir "Ce classeur"

Le raccourci est modifiable dans les options



The screenshot shows the 'Enregistrer une macro' (Record Macro) dialog box. It has a title bar with a question mark and a close button. The dialog contains the following fields and controls:

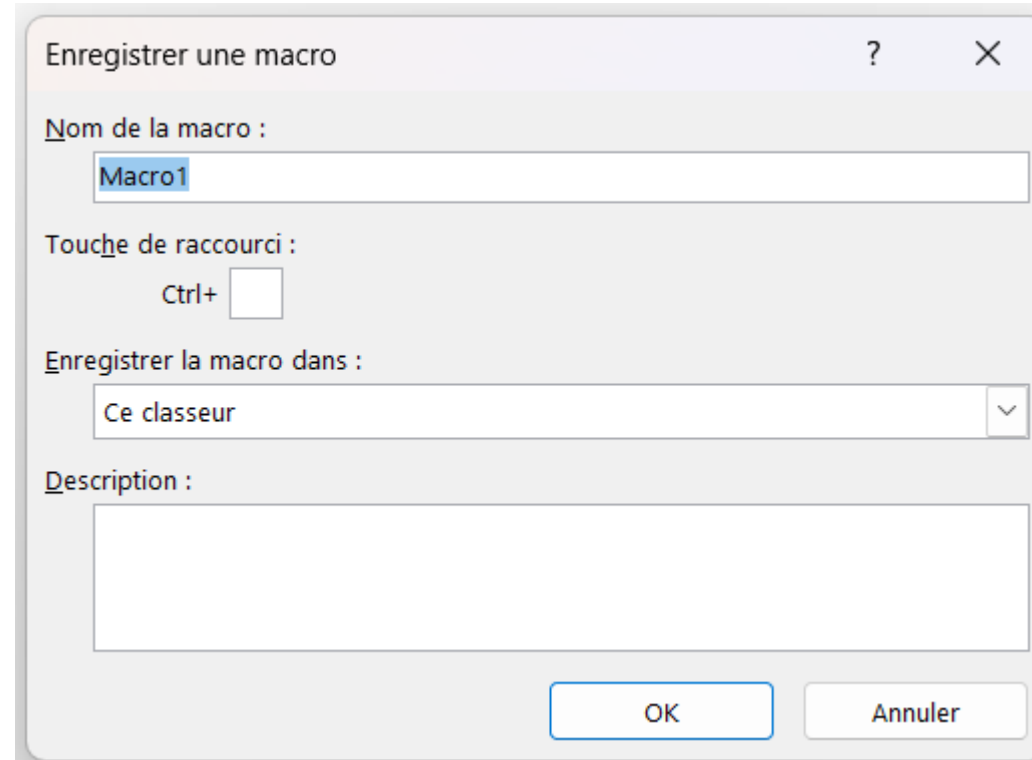
- Nom de la macro :** A text box containing 'Macro1'.
- Touche de raccourci :** A label followed by 'Ctrl+' and a small square button for selecting a key.
- Enregistrer la macro dans :** A dropdown menu with 'Ce classeur' selected.
- Description :** A large empty text area.
- Buttons:** 'OK' and 'Annuler' (Cancel) buttons at the bottom right.

Caractères de nommage

11

Sont interdits

- Espace
- Point .!?"(){}[]
- Arobase @&#\$
- Opérateur +-*/
- Autres...

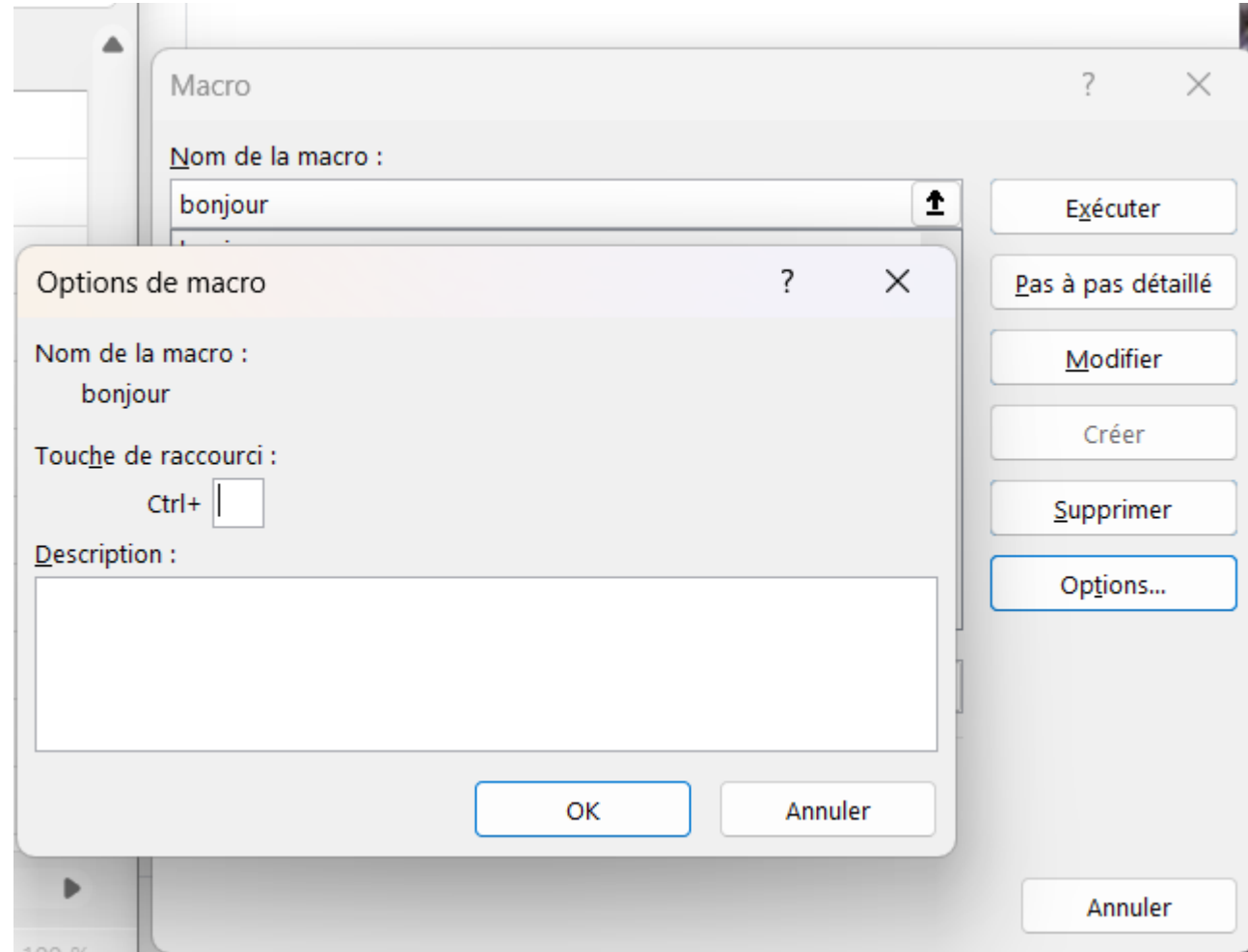


Sont autorisés

- Lettres minuscules
- Lettres majuscules
- Caractères accentués
- Souligné _
- Autres...

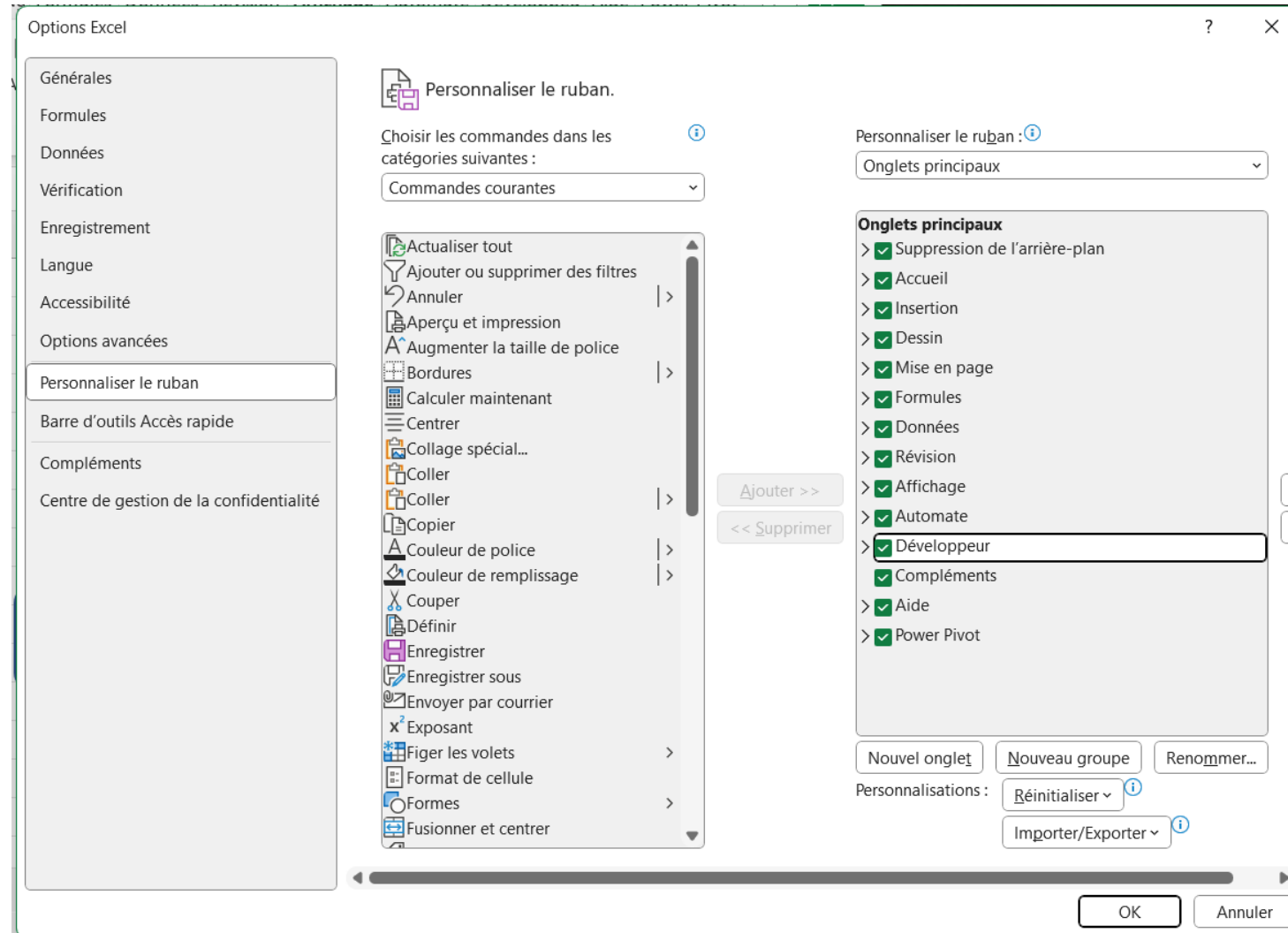
Raccourcis macros

12



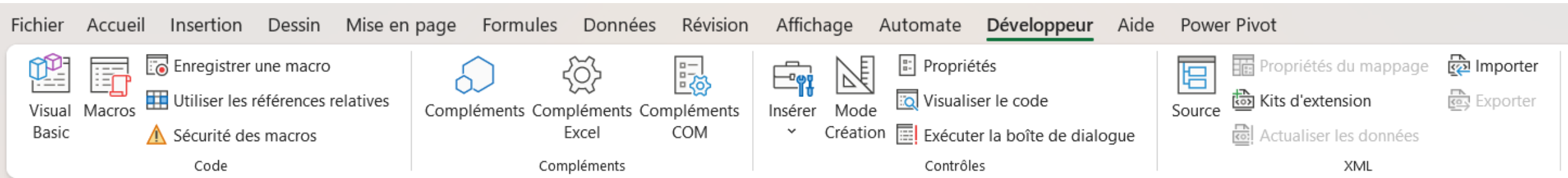
Onglet Développeur

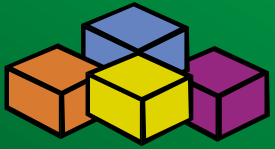
13



Onglet Développeur

14





L'éditeur

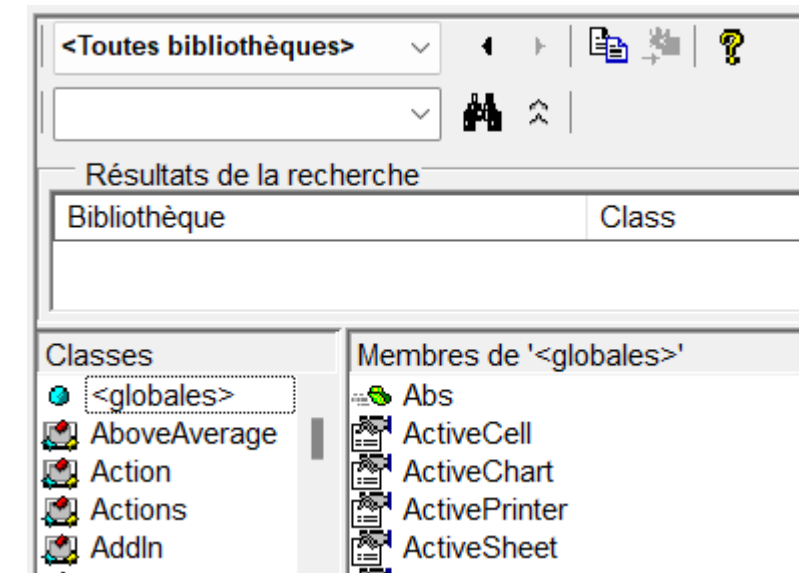
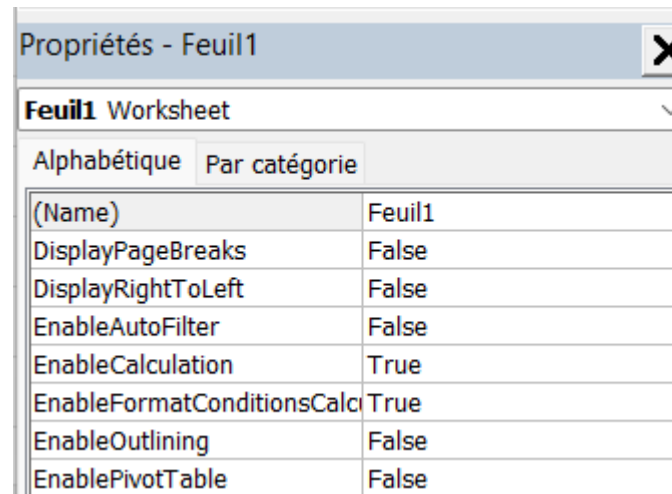
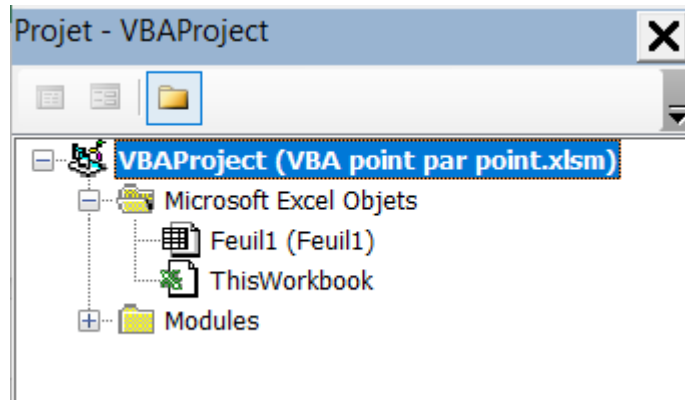
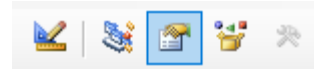
VBA

Alt + F11

(Alt+Fn+F11)

Fenêtres principales

17



<https://learn.microsoft.com/fr-fr/office/vba/language/reference/user-interface-help/icons-used-in-the-object-browser-and-code-windows>

Propriétés, méthodes, événements

18

<Toutes bibliothèques> listobjects

Résultats de la recherche

Bibliothèque	Class
Excel	ListObj
Excel	Worksh
Excel	Worksh
Excel	Worksh
Excel	Worksh
Excel	Worksh

Classes

- VbVarType
- VPageBreak
- VPageBreaks
- Walls
- Watch
- Watches
- WebOptions
- WebPageFont
- WebPageFonts
- Window
- Windows
- Workbook
- WorkbookConn
- WorkbookQuer
- Workbooks
- WorkflowTask
- WorkflowTasks
- WorkflowTempl
- Worksheet
- WorksheetData

Membres de 'Worksheet'

- HPageBreaks
- Hyperlinks
- Index
- LensGalleryRenderComplete
- ListObjects
- MailEnvelope
- Move
- Name**
- NamedSheetViews
- Names
- Next
- OLEObjects
- Outline
- PageSetup
- Parent
- Paste
- PasteSpecial
- PivotTableAfterValueChange
- PivotTableBeforeAllocateChanges
- PivotTableBeforeCommitChanges
- PivotTableBeforeDiscardChanges

Property **Name** As String
Membre de [Excel.Worksheet](#)

<Toutes bibliothèques> worksheet

Résultats de la recherche

Bibliothèque	Class
Excel	WorksheetDa
Excel	WorkbookCo
Excel	WorksheetFu
Excel	Application
Excel	Worksheets
Excel	Application

Classes

- Window
- Windows
- Workbook
- WorkbookConn
- WorkbookQuer
- Workbooks
- WorkflowTask
- WorkflowTasks
- WorkflowTempl
- Worksheet
- WorksheetData
- WorksheetFunc
- Worksheets**
- WorksheetView
- XIAboveBelow
- XIActionType
- XIAllocation
- XIAllocationMeth
- XIAllocationValu
- XIApplicationInte

Membres de 'Worksheets'

- Add
- Add2
- Application
- Copy
- Count
- Creator
- Delete**
- FillAcrossSheets
- HPageBreaks
- Item
- Move
- Parent
- PrintOut
- PrintPreview
- Select
- Visible
- VPageBreaks

Sub **Delete()**
Membre de [Excel.Worksheets](#)

<Toutes bibliothèques> workbooks

Résultats de la recherche

Bibliothèque	Class
Excel	Workboo
Excel	Applicati
Excel	Applicati

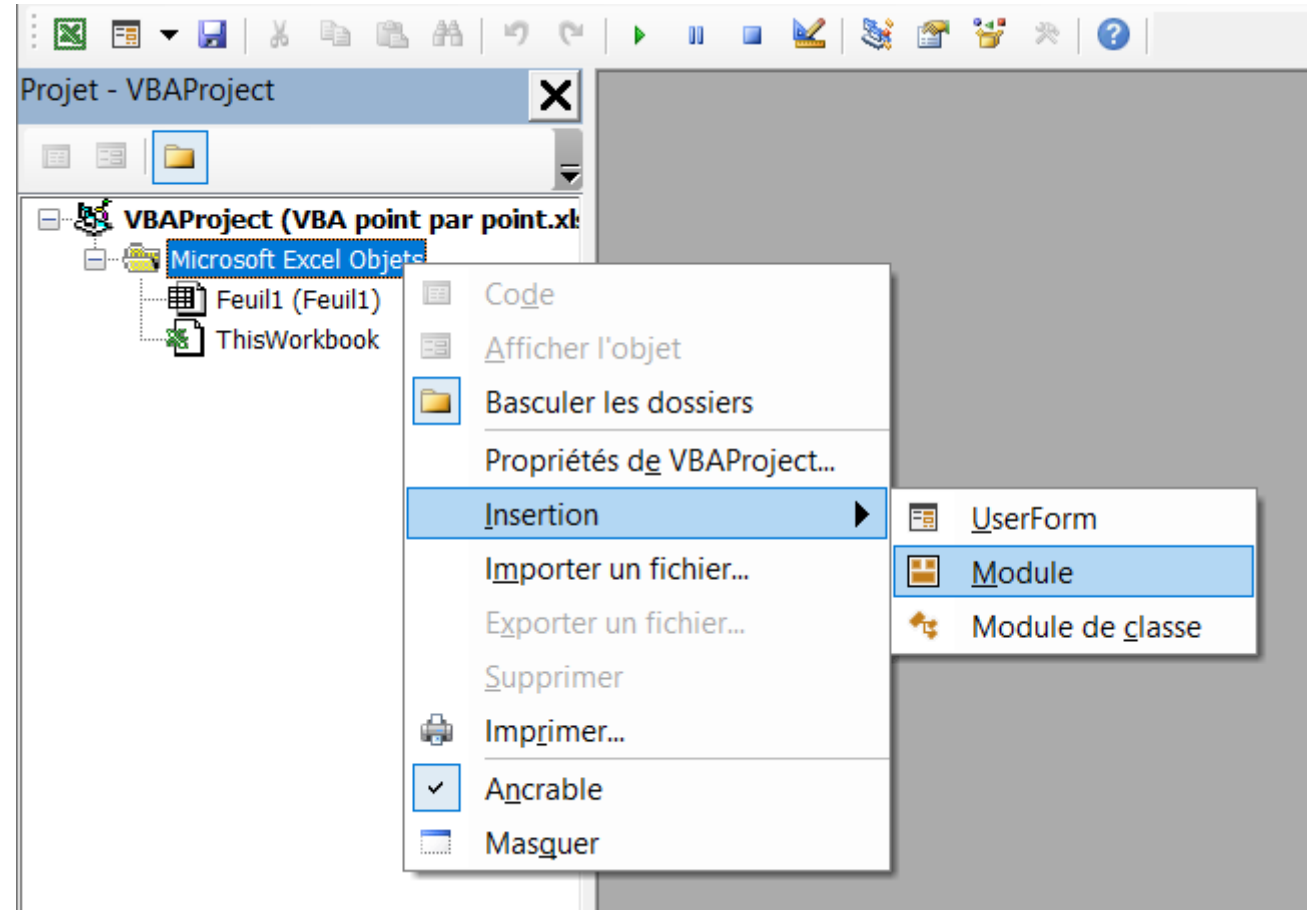
Classes

- Application**
- Areas
- Author
- AutoCorrect
- AutoFilter
- AutoRecover
- Axes
- Axis
- AxisTitle
- BackstageGrou
- Border
- Borders
- BulletFormat2
- CalculatedField
- CalculatedItems
- CalculatedMem
- CalculatedMem
- CalloutFormat
- CategoryCollect
- CellFormat
- CertificateDetail

Membres de 'Application'

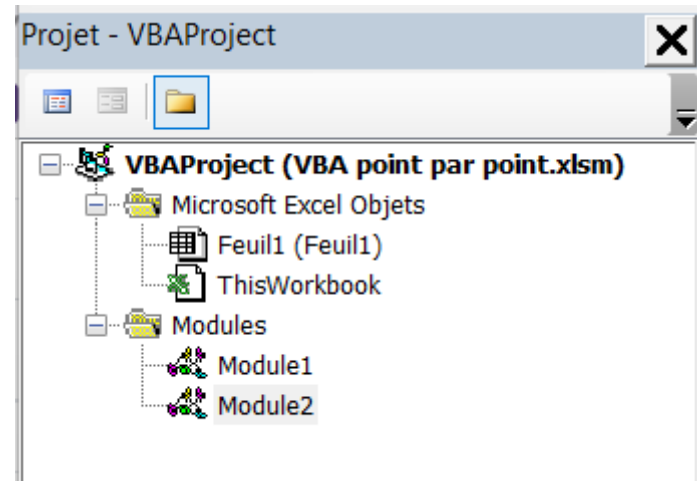
- Wait
- WarnOnFunctionNameConflict
- Watches
- Width
- WindowActivate**
- WindowDeactivate
- WindowResize
- Windows
- WindowsForPens
- WindowState
- WorkbookActivate
- WorkbookAddinInstall
- WorkbookAddinUninstall
- WorkbookAfterRemoteChange
- WorkbookAfterSave
- WorkbookAfterXmlExport
- WorkbookAfterXmlImport
- WorkbookBeforeClose
- WorkbookBeforePrint
- WorkbookBeforeRemoteChange
- WorkbookBeforeSave

Event **WindowActivate**(Wb As [Workbook](#), Wn As [Window](#))
Membre de [Excel.Application](#)



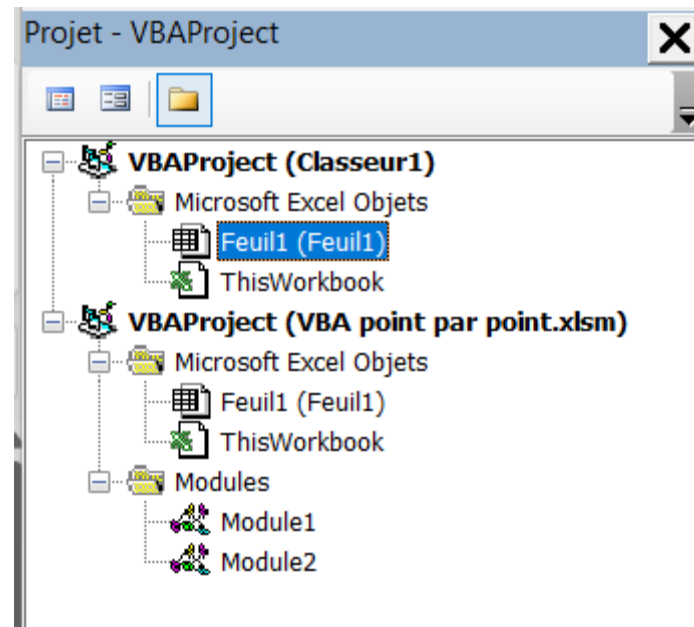
Un nouveau module à chaque ouverture

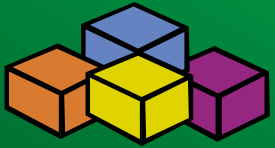
20



Tous les classeurs dans l'explorateur

21





Déclencher

VBA

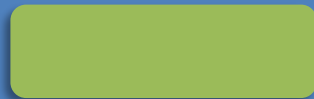
Trois façons de déclencher une macro

23

Avec Exécuter
dans le code



En créant un
bouton



A partir d'un
évènement

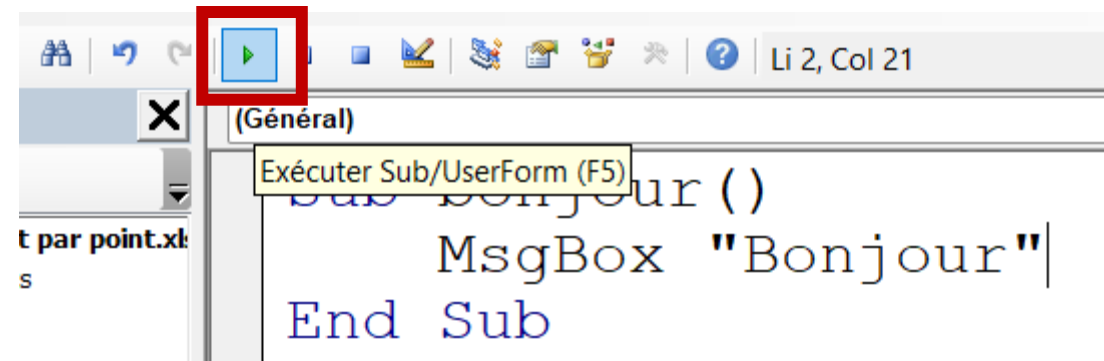


Avec un
formulaire



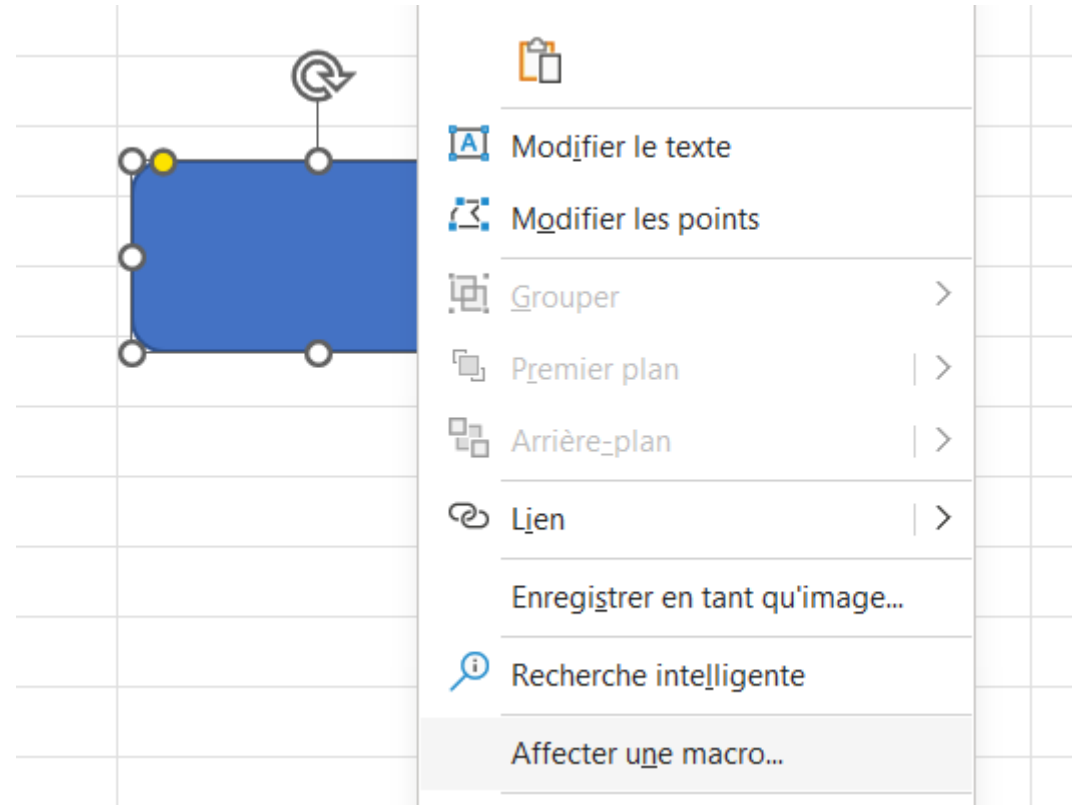
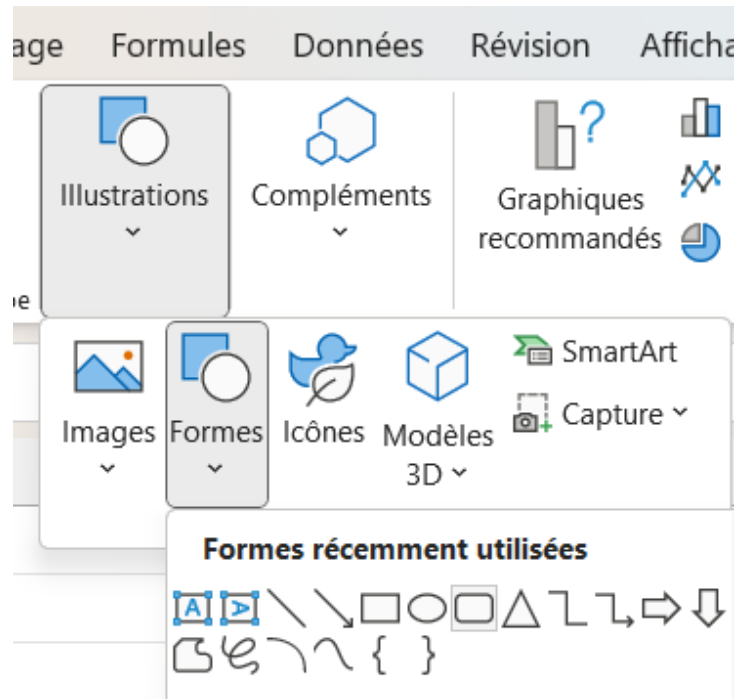
Exécuter dans le code

24



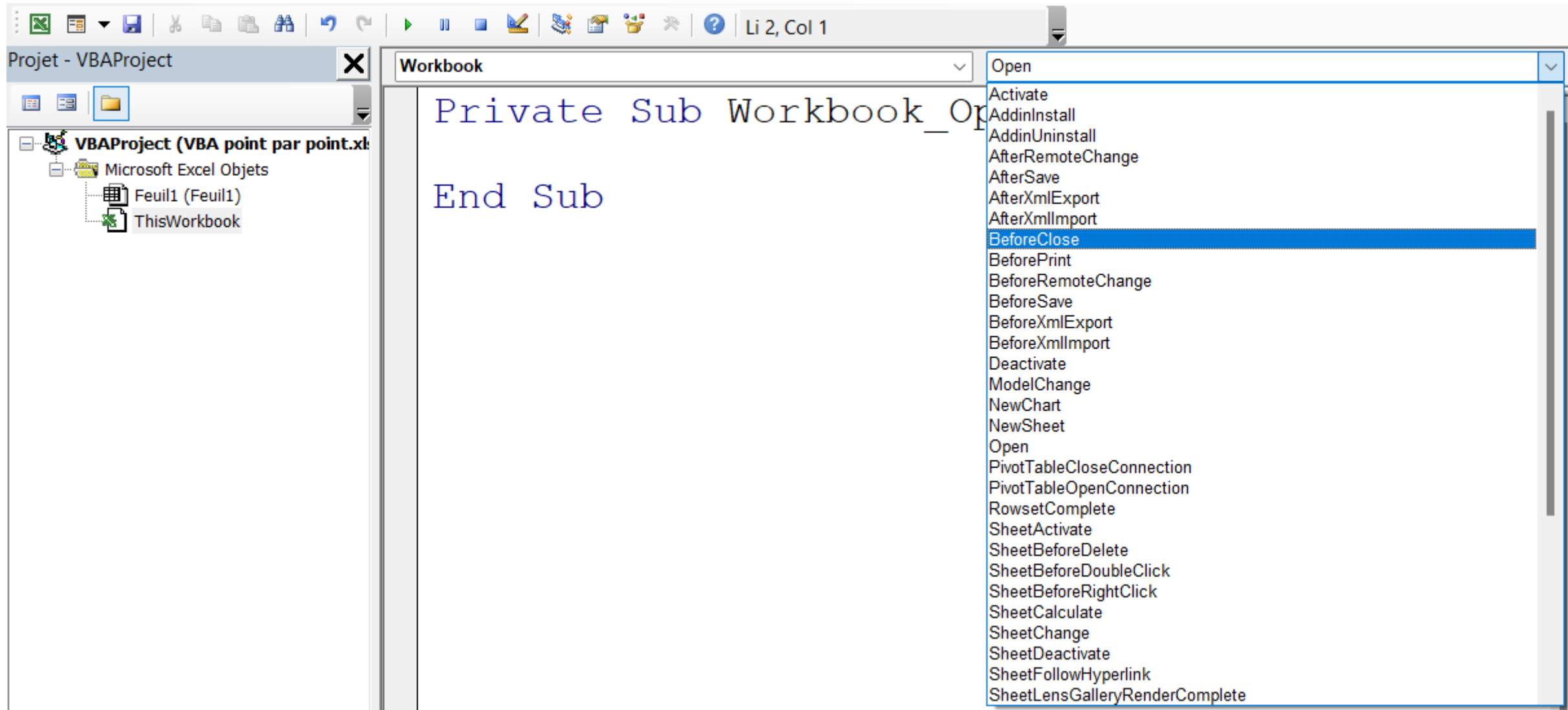
Créer un bouton et affecter la macro

25



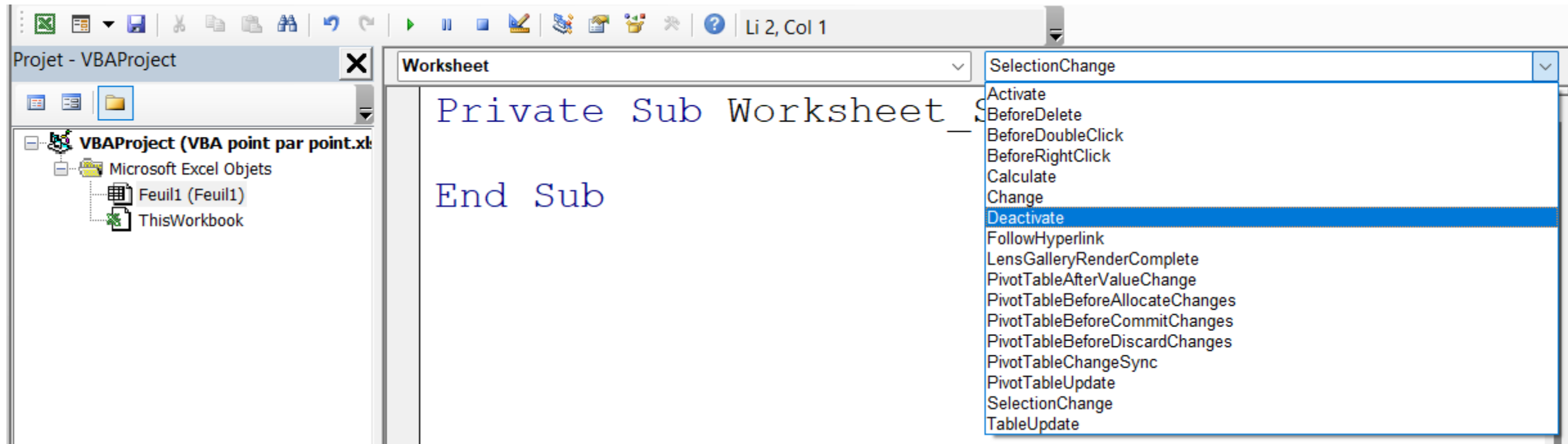
Évènement Classeur (PC)

26



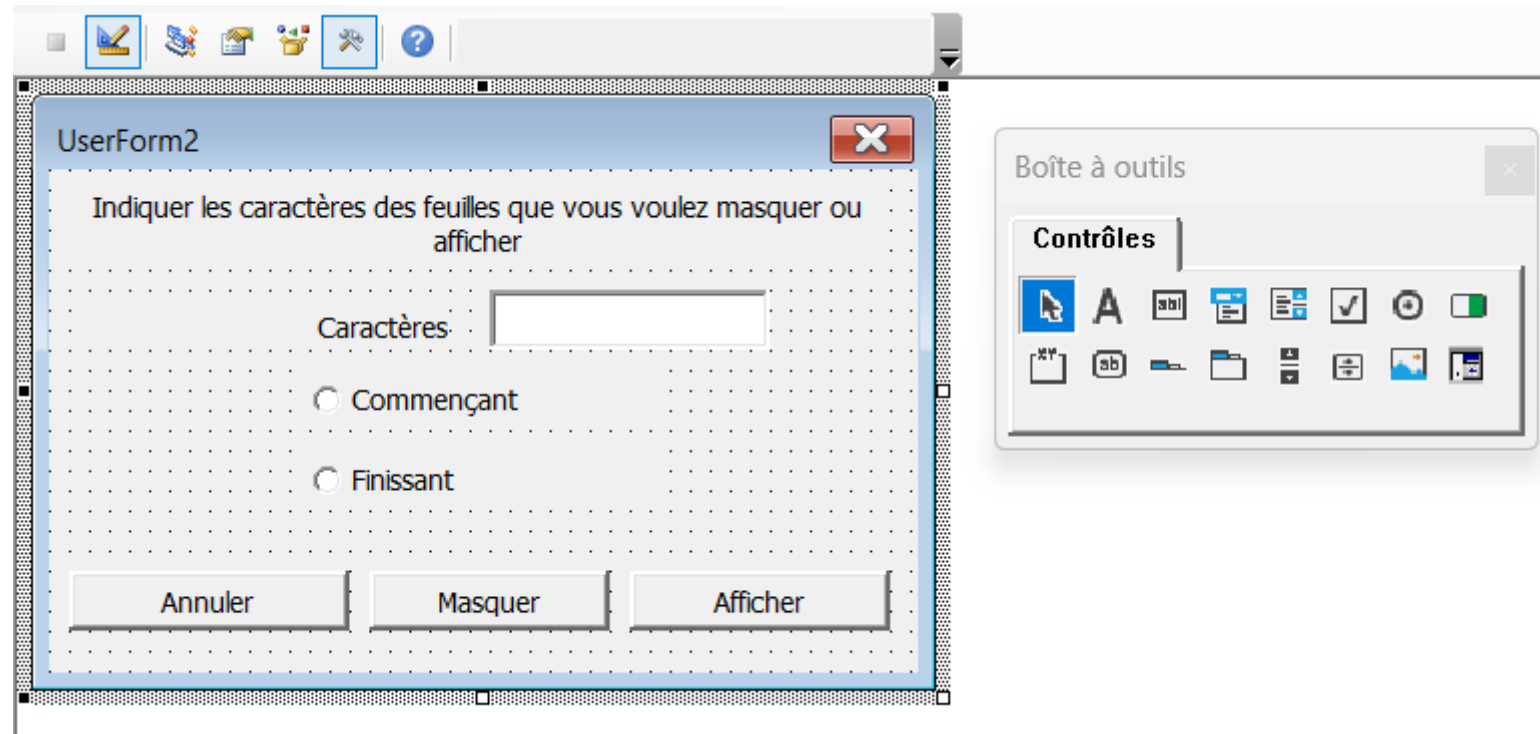
Évènement Feuille (PC)

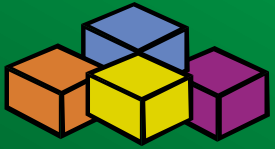
27



Formulaire et boîte à outil (PC)

28





Codes Simples

VBA

Clearcontents pour supprimer

30

```
Sub viderCellules()  
    Selection.ClearContents  
End Sub
```

	A	B	C	D
1				
2		Bonjour	11/02/2023	
3		500	12	hermione
4		Harry	Ron	36

	A	B	C	D
1				
2				
3				
4				

Ajuster la taille des colonnes

31

```
Sub testAutofit()
```

```
'ajuste la taille des colonnes de la sélection
```

```
Selection.EntireColumn.AutoFit
```

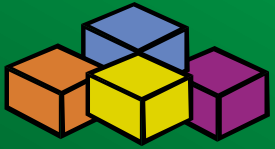
```
End Sub
```

	A	B	C	D
1		Feuil2	Feuil3	Feuil4
2				
3				
4				

	A	B	C	D
1		Feuil2	Feuil3	Feuil4
2				
3				
4				

```
Sub actualiserTout()  
' actualiserTout Macro  
    ActiveWorkbook.RefreshAll  
End Sub
```





Commentaire

VBA

Commentaire avec apostrophe

34

```
Sub viderCellules()
```

```
' Efface le contenu
```

```
    Selection.ClearContents
```

```
End Sub
```

```
Sub viderCellules()
```

```
    Selection.ClearContents ' Efface le contenu
```

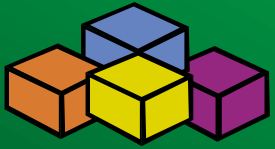
```
End Sub
```

```
Sub viderCellules()
```

```
    Rem Efface le contenu
```

```
        Selection.ClearContents
```

```
End Sub
```



Écrire dans une cellule

VBA

```
Sub écrire()  
    Range("A1") = "Bonjour"  
End Sub
```

```
Sub écrire()  
    Range("A1").value = "Bonjour"  
End Sub
```

	A	B
1	Bonjour	
2		
3		
4		

```
Sub lireCellule()  
    Cells(1, 2) = "Bonjour"  
End Sub
```

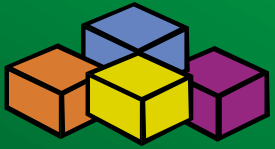
```
Sub lireCellule()  
    Cells(1, 2).value = "Bonjour"  
End Sub
```

	A	B
1		Bonjour
2		

```
Sub sélectionnerCellule()  
    Range("A1").Select  
End Sub
```

```
Sub sélectionnerCellule()  
    Cells(1,1).Select  
End Sub
```

```
Sub sélectionnerPlage()  
    Range("A1:A6"). Select  
End Sub
```



Afficher un message

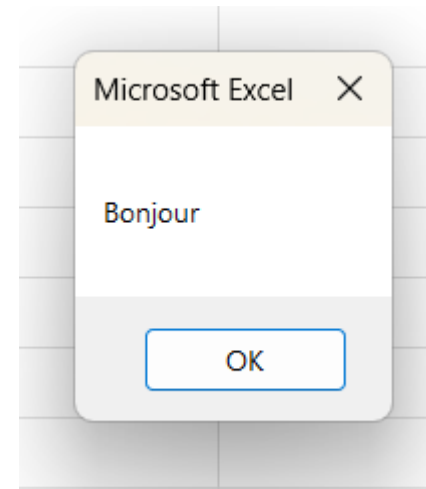
VBA

Lire une cellule: range value

41

```
Sub lireCellule()  
    MsgBox Range("A1").Value  
End Sub
```

	A	B
1	Bonjour	
2		
3		
4		
5		
6		



Concaténation avec esperluette &

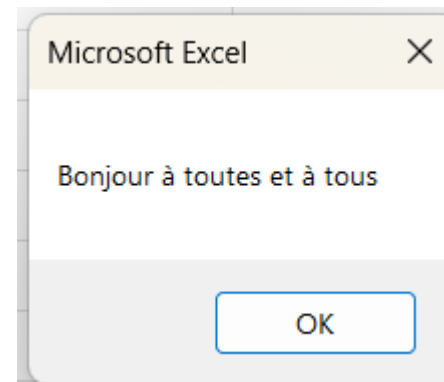
42

```
Sub lireCellule()
```

```
    MsgBox Range("A1").Value & " à toutes et à tous"
```

```
End Sub
```

	A	B
1	Bonjour	
2		
3		
4		
5		
6		



Retour à la ligne

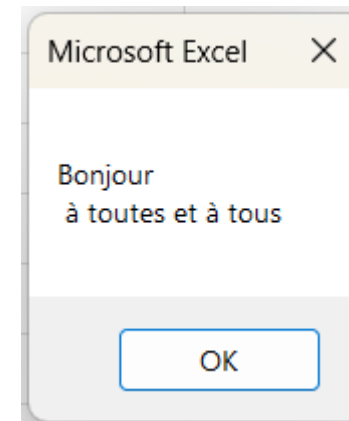
43

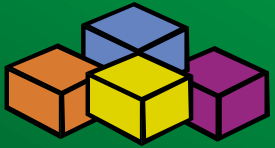
```
Sub lireCellule()
```

```
    MsgBox Range("A1").Value & vbCrLf & " à toutes et à tous"
```

```
End Sub
```

	A	B
1	Bonjour	
2		
3		
4		
5		
6		





Lire une cellule

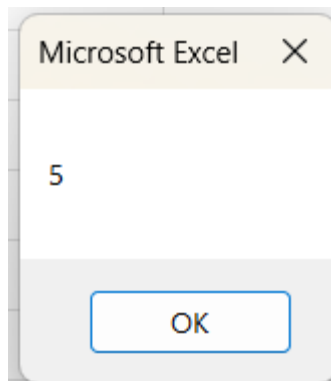
VBA

	A	B	C
1	2	3	=A1+B1
2			

```
Sub lire()
```

```
    MsgBox Range("C1")
```

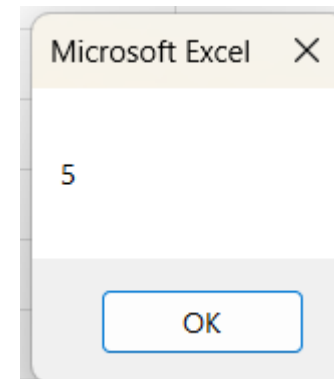
```
End Sub
```



```
Sub lire()
```

```
    MsgBox Range("C1").value
```

```
End Sub
```



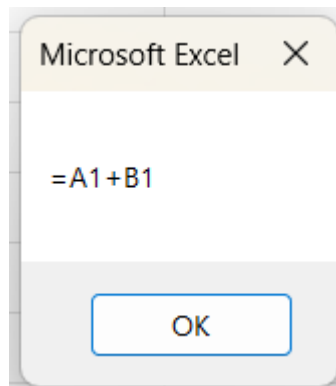
	A	B	C
1	2	3	=A1+B1
2			

Sub lire()

MsgBox Range("C1").Formula

'ou Formula2

End Sub

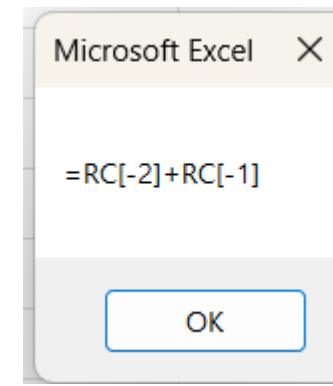


Sub lire()

MsgBox Range("C1").FormulaR1C1

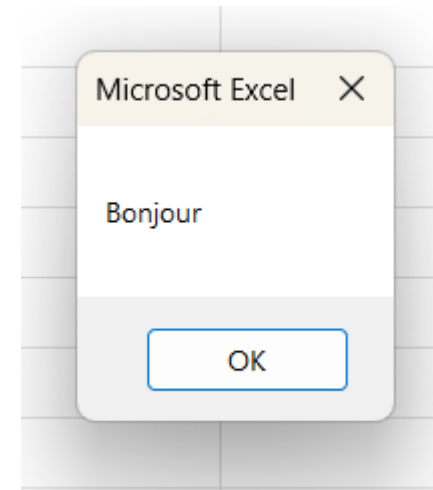
'ou Formula2R1C1

End Sub



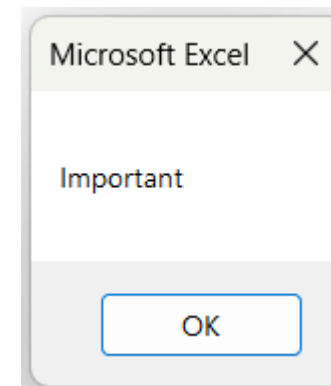
```
Sub lireCellule()  
    MsgBox Cells(1,2).Value  
End Sub
```

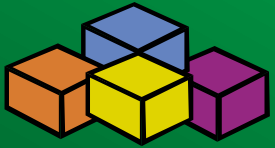
	A	B
1		Bonjour
2		



```
Sub lireCelluleDécalée()  
    MsgBox ActiveCell.Offset(0, 1).Value  
End Sub
```

	A	B
1	Mois	Commentaire
2	Janvier	
3	Février	Important
4		





Fonctions de texte

VBA

Fonctions de texte les plus fréquentes

50

' Gauche

Left(ActiveCell.Value,1)

' Droite

Right(ActiveCell.Value,1)

' Milieu (STXT)

Mid(ActiveCell.Value,2,3)

' Majuscule / Minuscule

UCase(ActiveCell.Value)

LCase(ActiveCell.Value)

' Enlever les espaces extérieures

TRIM(ActiveCell.Value)

' *versions simplifiées*

' Séparer (voir array)

Split(ActiveCell.Value,"_")

' Chercher en partant de la gauche

InStr(ActiveCell.Value, "p")

' Chercher en partant de la droite

InStrRev(ActiveCell.Value, "p")

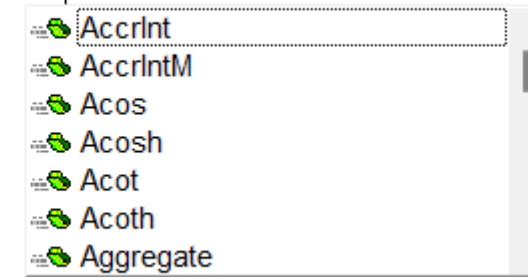
' Remplacer

Replace(ActiveCell.Value,"p","P")

Fonctions de l'application

52

`Application.WorksheetFunction.`



'Vérifier si la colonne A est vide

`MsgBox Application.WorksheetFunction.CountA(Range("A:A")) = 0`

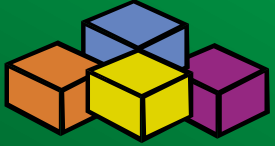
NB Count

NB.SI.ENS Countifs

NBVAL CountA

RechercheV Vlookups

NB.SI CountIF



Conditions

VBA

```
IF ... THEN
```

```
...
```

```
END IF
```

```
IF ... THEN
```

```
...
```

```
ELSE
```

```
...
```

```
END IF
```

```
IF ... THEN
```

```
...
```

```
ELSEIF ... THEN
```

```
...
```

```
ELSE
```

```
...
```

```
END IF
```

Condition avec Select Case

55

```
Select Range("A1").value
    Case "E"
        Range("B1") *0,1
    Case "A"
        Range("B1") *0,2
    Case Else
        Range("B1") *0,3
End Select
```

Condition avec Select Case et IS

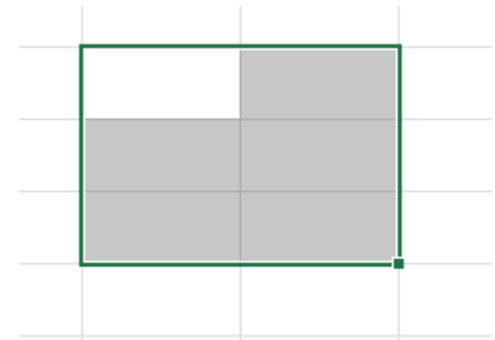
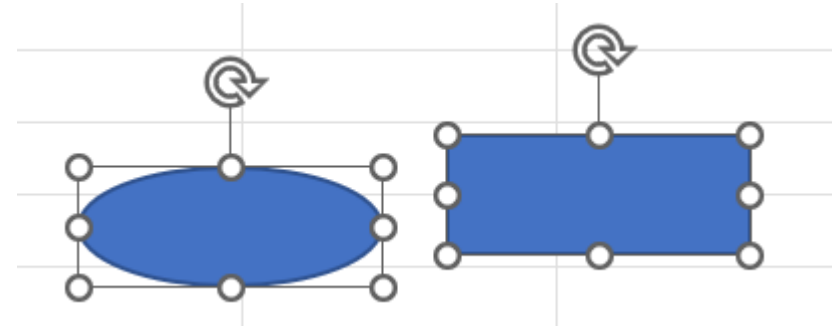
56

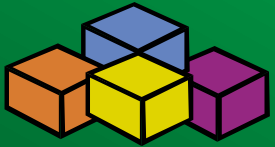
```
Select Range("C1").value
    Case Is < 500
        Range("B1") *0,1
    Case Is < 1000
        Range("B1") *0,2
    Case Else
        Range("B1") *0,3
End Select
```


Évaluer la sélection

57

```
Sub TestIfRange()  
  If TypeName(Selection) = "Range" Then  
    MsgBox "Une plage est sélectionnée"  
  Else  
    MsgBox "La selection contient: " & TypeName(Selection)  
  End If  
End Sub
```





Message plusieurs boutons







VBA

Arguments de MsgBox

59

```
msgbox("Tout supprimer",
```

`MsgBox(Prompt, [Buttons As VbMsgBoxStyle = vbOKOnly], [Title], [HelpFile], [Context]) As VbMsgBoxResult`

-  vbOKOnly
-  vbQuestion
-  vbRetryCancel
-  vbSystemModal
-  vbYesNo
-  vbYesNoCancel

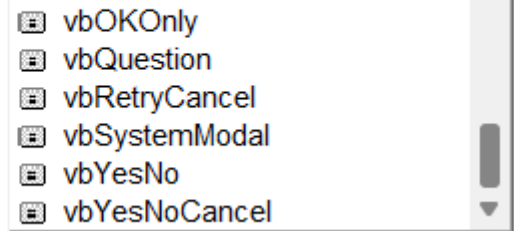
Arguments de MsgBox

```
Sub messageChoix()  
    Dim réponse  
    réponse = MsgBox("Tout supprimer?", vbOKCancel)  
    If réponse = vbCancel Then  
        MsgBox "rien n'a été supprimé"  
    Else  
        MsgBox "tout est perdu"  
    End If  
End Sub
```

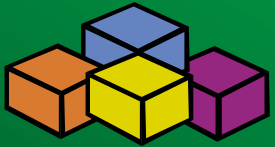
Arguments de MsgBox

61

```
msgbox("Tout supprimer",  
MsgBox(Prompt, [Buttons As VbMsgBoxStyle = vbOKOnly], [Title], [HelpFile], [Context]) As VbMsgBoxResult
```



réponse = MsgBox(Buttons:=vbOKCancel, prompt:="Tout supprimer?")



Demander une information

VBA

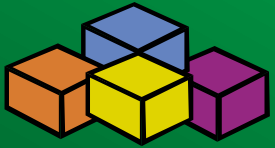
```
Sub demanderInfo()  
    Dim réponse  
    réponse = InputBox("Votre montant?")  
    If réponse < 5 Then  
        MsgBox réponse * 0.2  
    Else  
        MsgBox réponse * 0.5  
    End If  
End Sub
```

Application inputbox

64

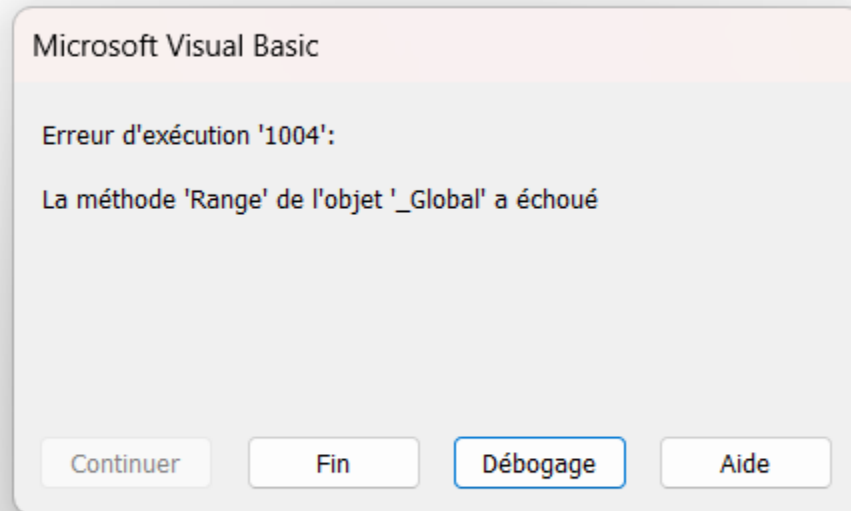
```
Sub demanderCellule()  
    Dim c  
    Set c = Application.InputBox("Choisir cellule", Type:=8)  
    c.Value = "bonjour"  
End Sub
```

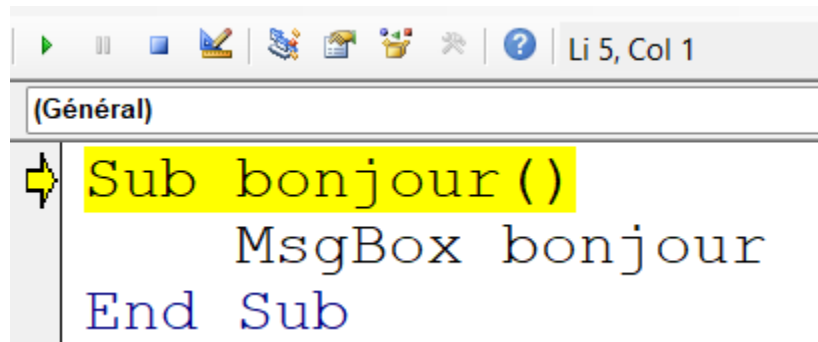
<https://learn.microsoft.com/fr-fr/office/vba/api/excel.application.inputbox>



Debogage

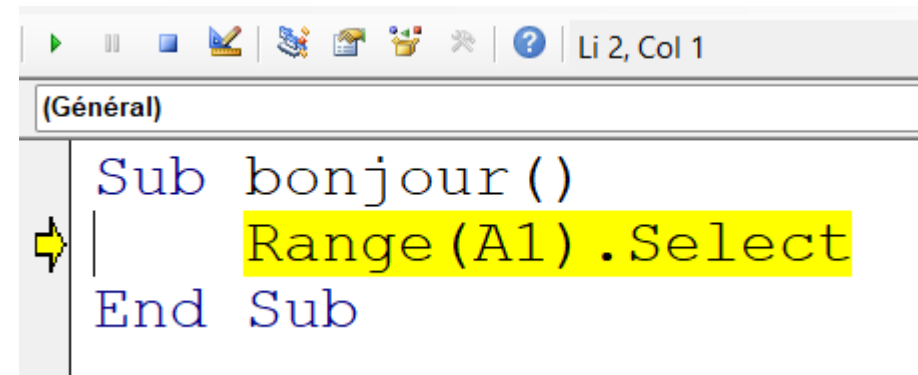
VBA





The screenshot shows a VBA editor window with a toolbar at the top containing icons for Run, Break, Insert, Add-Ins, View, Macros, and Help. The status bar indicates 'Li 5, Col 1'. The 'Général' tab is selected. The code is as follows:

```
Sub bonjour()  
    MsgBox bonjour  
End Sub
```

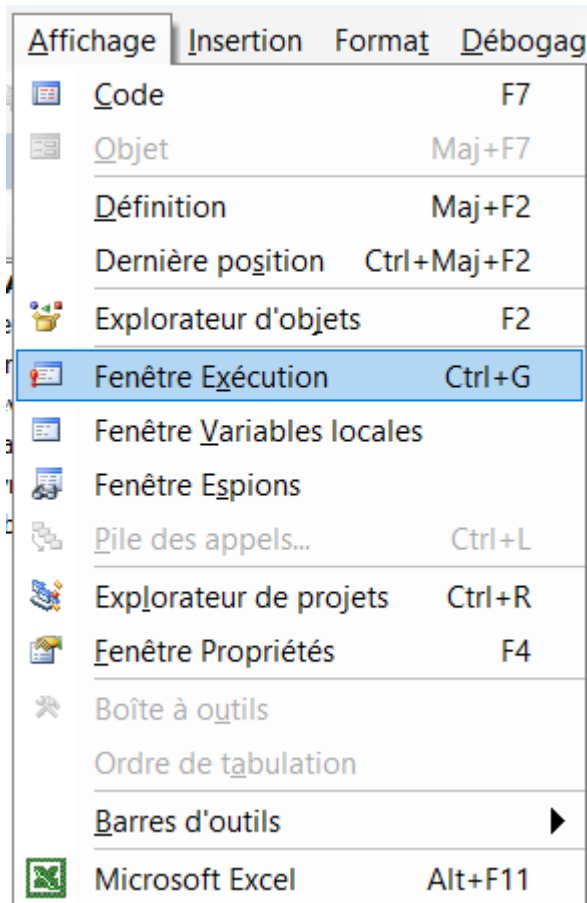


The screenshot shows a VBA editor window with a toolbar at the top containing icons for Run, Break, Insert, Add-Ins, View, Macros, and Help. The status bar indicates 'Li 2, Col 1'. The 'Général' tab is selected. The code is as follows:

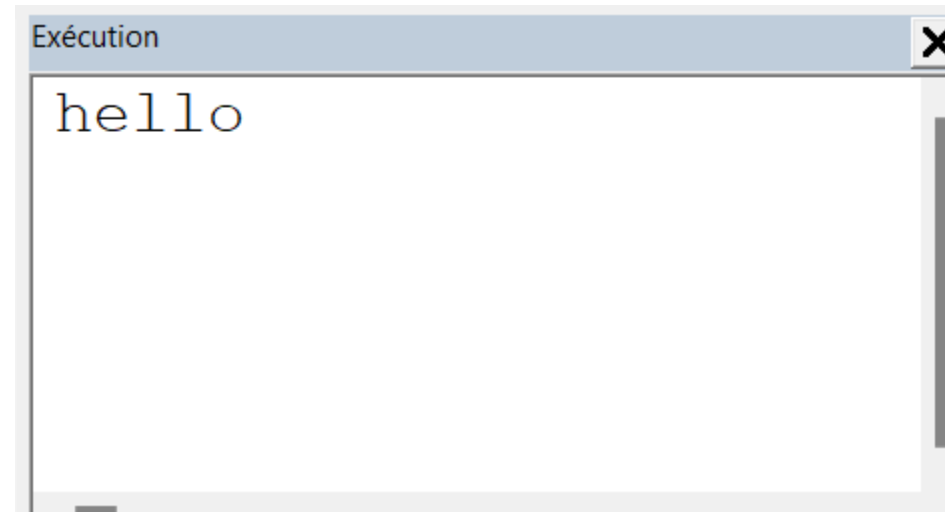
```
Sub bonjour()  
    Range(A1).Select  
End Sub
```

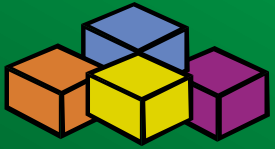
Debug.Print Fenêtre d'exécution

68



```
Sub InfoDébogage()  
    Debug.Print "hello"  
End Sub
```

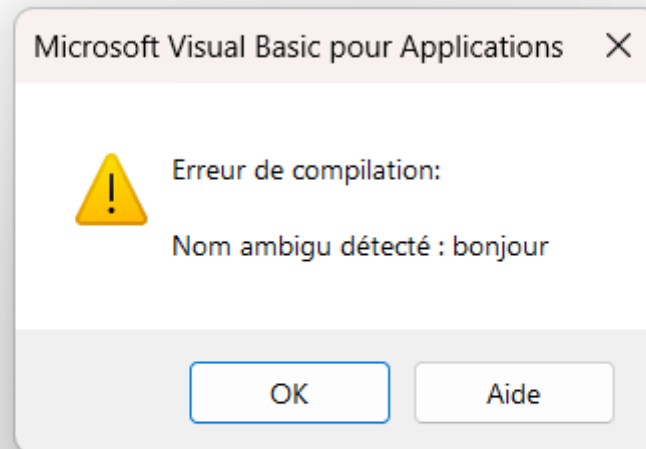




Messages d'erreur

VBA

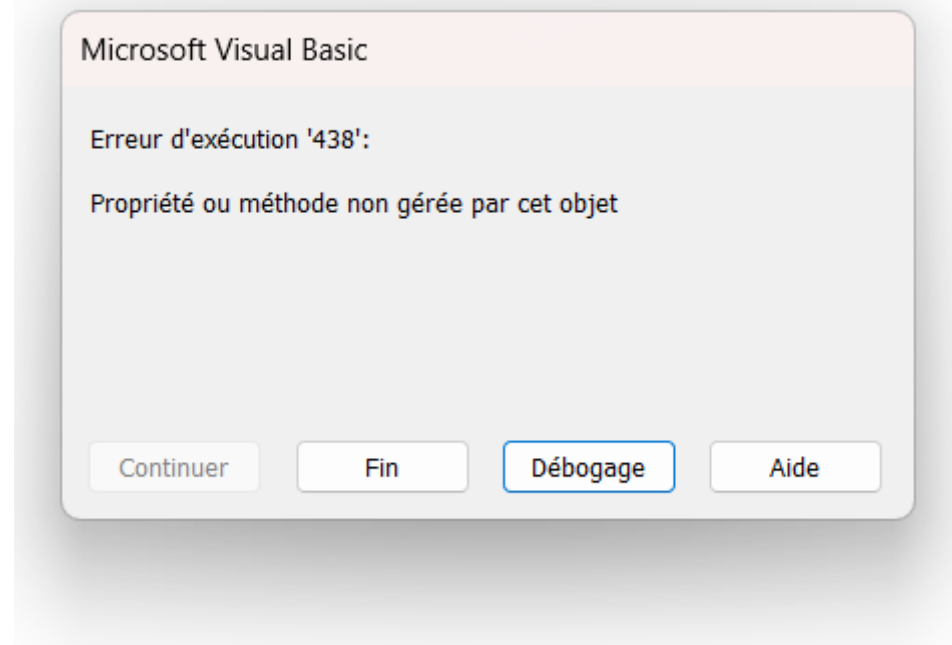
Il y a deux macros qui ont le même nom



Propriété ou méthode non gérée par cet objet 71

Exemple:

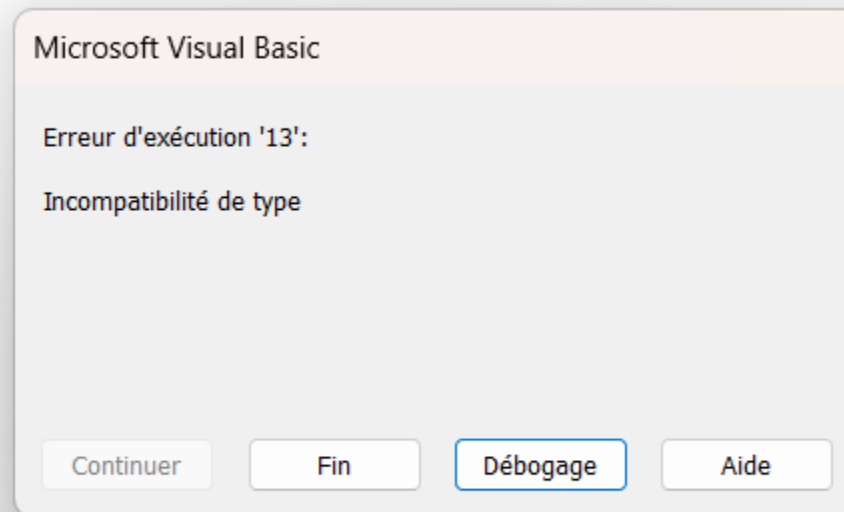
Selection.clearcontents alors que c'est une forme ou un graphique qui est sélectionné



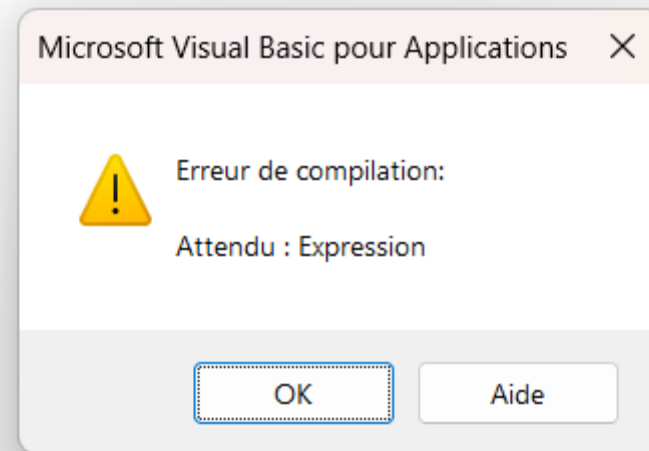
Incompatibilité de type

72

Vous essayez de faire une multiplication sur du texte



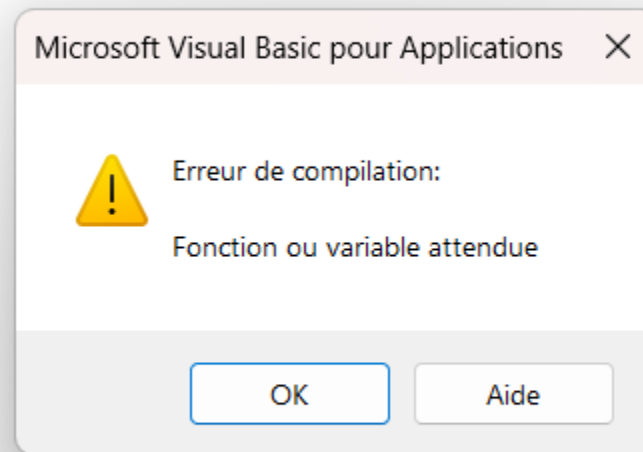
Une syntaxe n'est pas terminée (par exemple un IF)

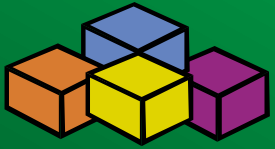


Attendu Expression

74

Vous n'avez pas peut-être pas mis de guillemet





Variables

VBA

Types les plus fréquents

76

' Pour le texte

Dim s as String

' Pour les entiers

' de -32 768 à 32 767

Dim i as Integer

' Pour gd entiers

Dim lo as Long

' Pour les décimaux

Dim de as Decimal

' Pour les dates

Dim d as Date

' Pour le Vrai / Faux

Dim b as Boolean

' Pour les objets

Dim ws as WorkSheet

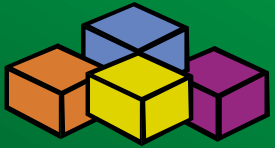
Dim wb as Workbook

' Pour le reste

Dim v as Variant

' Pour plus tard

Dim a as Array



Propriété

VBA

```
Sub mettreEnGras()
```

```
  ' met en gras la selection
```

```
    Selection.Font.Bold = True
```

```
End Sub
```

	A	B
1		
2		Bonjour
3		Au revoir
4		

Plusieurs propriétés avec WITH

79

```
Sub mettreEnJaune()
```

```
    'Met en jaune la cellule active
```

```
    With Selection.Interior
```

```
        .Pattern = xlSolid
```

```
        .PatternColorIndex = xlAutomatic
```

```
        .Color = 65535
```

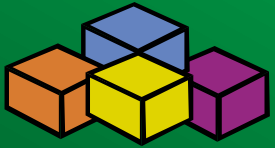
```
        .TintAndShade = 0
```

```
        .PatternTintAndShade = 0
```

```
    End With
```

```
End Sub
```

	A	B
1		
2		
3		
4		



Boucles

VBA

For Each

81

Sub répéter()

Dim c

For Each c In Selection

c.Value = c.Value + 1

Next c

End Sub

	A	B	C
1			
2		10	40
3		20	50
4		30	60

	A	B	C
1			
2		11	41
3		21	51
4		31	61

For Each

82

Sub répéter()

Dim i As Integer

For i = 1 To 5

Cells(1, i) = "Sem " & i

Next i

End Sub

	A	B	C	D	E
1	Sem 1	Sem 2	Sem 3	Sem 4	Sem 5

For Each

83

```
Sub répéter()
```

```
    Dim i As Integer
```

```
    For i = 20 To 10 Step -1
```

```
        Cells(1, i) = "Sem " & i
```

```
    Next i
```

```
End Sub
```

DO WHILE LOOP

84

Sub répéter()

Dim i As Integer

i = 2

Do While i < 5

Cells(i, 1) = Cells(i - 1, 1) + 2

i = i + 1

Loop

End Sub

	A
1	
2	2
3	4
4	6
5	
6	

CTRL+ PAUSE

Ctrl+Fn+PAUSE

Ctrl+Break

Ctrl+Fn+Break

Double boucle

86

```
Sub trierSelection()
```

```
'met les cellules sélectionnées par ordre alphabétique
```

```
Dim c, n As String
```

```
For i = 1 To Selection.Count
```

```
For j = i + 1 To Selection.Count
```

```
c = Selection(i).Value 'valeur de la cellule actuelle (current)
```

```
n = Selection(j).Value 'valeur de la cellule suivante (next)
```

```
If Selection(j).Value < Selection(i).Value Then
```

```
Selection(i).Value = n
```

```
Selection(j).Value = c
```

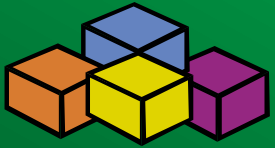
```
End If
```

```
Next j
```

```
Next i
```

```
End Sub
```

Guerlain	Dior	Azzaro	Chanel	Paco Rabanne	Armani	Cerutti
Armani	Azzaro	Cerruti	Chanel	Dior	Guerlain	Paco Rabanne

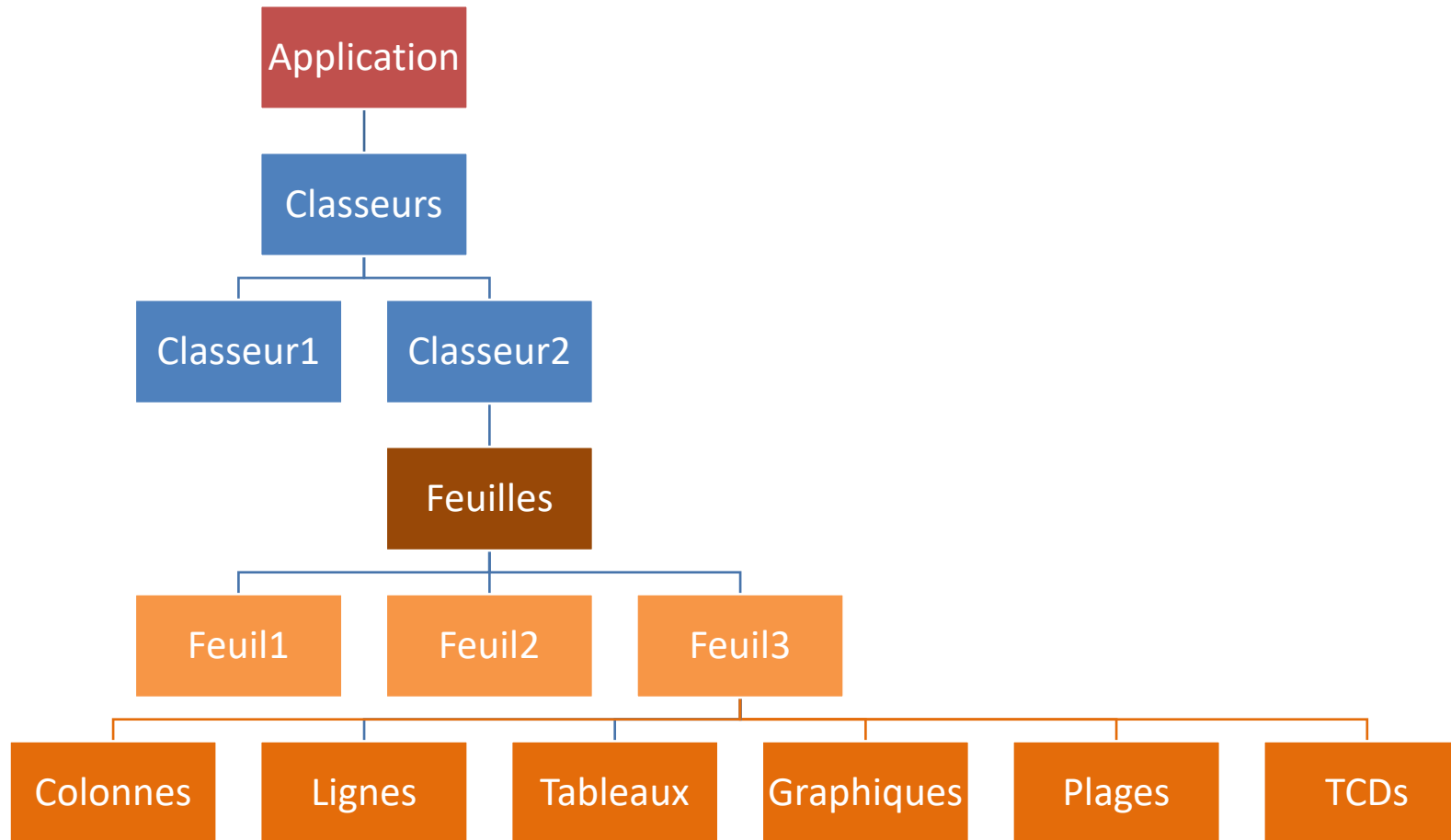


Collection

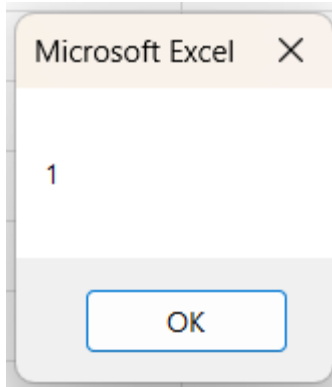
VBA

Principales collections dans Excel

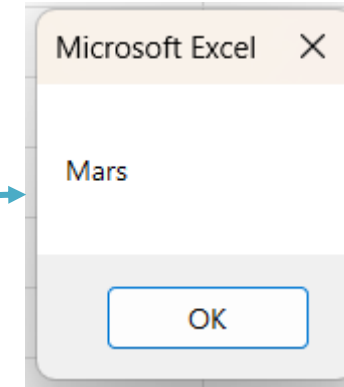
88



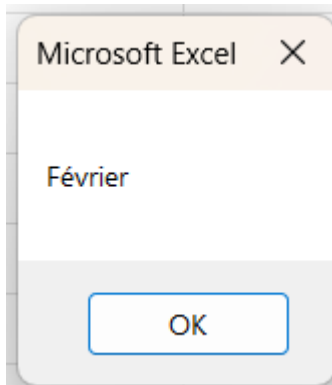
MsgBox Sheets("Janvier").Index



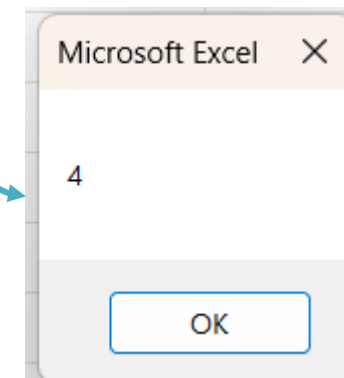
MsgBox ActiveSheet.Name



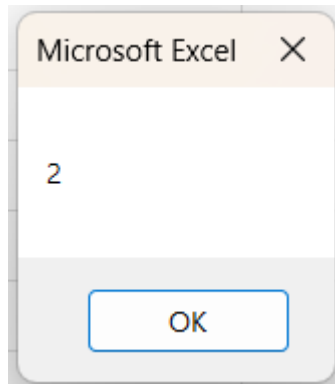
MsgBox Sheets(2).Name



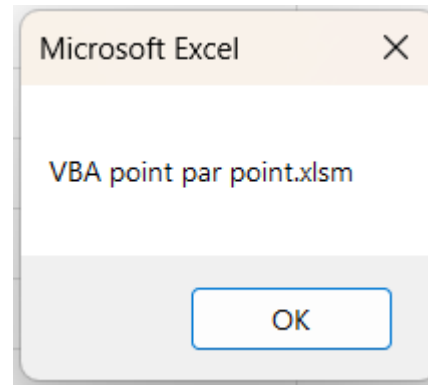
MsgBox Sheets.Count



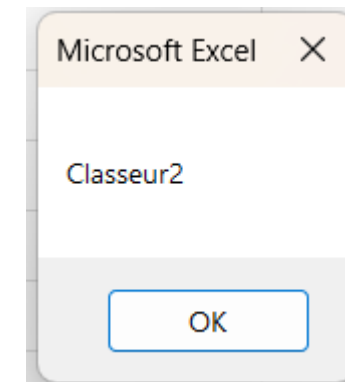
MsgBox Workbooks.Count



MsgBox ThisWorkbook.Name



MsgBox ActiveWorkbook.Name



Formes
Shapes



Graphiques
ChartObjects

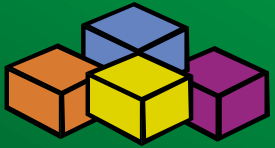


Tableaux
ListObjects



TCD
PivotTables





Méthodes

VBA

Ajouter Feuille

93

'Ajoute une feuille

Sheets.Add

Sheets.Add After:=Sheets("Input")

Sheets.Add.Name = "NewSheet"

Sheets.Add(After:=Sheets("Input")).Name = "NewSheet"

Sheets.Add After:=Sheets(Sheets.Count)

Insérer des lignes

94

Row insert

'insérer une ligne au-dessus de la ligne 4

`Rows(4).Insert`

`Range("b4").EntireRow.Insert`

	A
1	1
2	2
3	3
4	4
5	5

	A
1	1
2	2
3	3
4	
5	4
6	5

'insérer plusieurs lignes

`Rows("4:6").Insert`

	A
1	1
2	2
3	3
4	
5	
6	
7	4
8	5
9	6
10	7
11	8

En déduire comment on ajouterait une ou plusieurs colonnes...

Copy paste

```
Range("A1").Copy Range("B1")
```

```
Range("A1:A3").Copy Range("B1:B3")
```

'Couper Coller une ligne

```
Range("1:1").Cut Range("2:2")
```

'Collage spécial

```
Range("A1").Copy
```

```
Range("B1").PasteSpecial Paste:=xlPasteFormulas
```

'copier coller sur une autre feuille d'un autre classeur

```
Workbooks("book1.xlsm").Worksheets("sheet1").Range("A1").Copy _  
    Workbooks("book2.xlsm").Worksheets("sheet1").Range("B1")
```

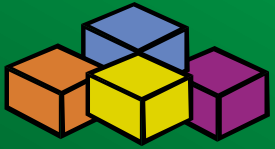
'simplifions cette longue commande avec Set

Dim sws, dws as worksheet 'feuilles source et destination

```
Set sws = Workbooks("book1.xlsx").Worksheets("sheet1")
```

```
Set dws = Workbooks("book2.xlsx").Worksheets("sheet1")
```

```
sws.range("A1").copy dws.range("B1")
```

Arrêts

VBA

Point d'arrêt (Pause)

98

```
Sub répéter()  
    Dim i As Integer  
    i = 2  
    Do While i < 10  
        Cells(i, 1) = "Sem " & i  
        i = i + 1  
    Loop  
End Sub
```

```
Sub répéter()  
    ' la boucle fait pause tous les multiples de 100  
    Dim i As Integer  
    i = 2  
    Do While i < 500  
        Cells(i, 1) = Cells(i - 1, 1) + 2  
        i = i + 1  
        ' mod est le modulo: le reste de la division  
        If i Mod 100 = 0 Then  
            Stop  
        End If  
    Loop  
End Sub
```

Exit loop

Exit Sub

'attend une certaine heure aujourd'hui

```
Application.Wait "18:23:00"
```

'indique si 10 secondes se sont écoulées

```
If Application.Wait(Now + TimeValue("0:00:10")) Then
```

```
    MsgBox "Time expired"
```

```
End If
```



Modifier ce qui s'affiche

'Arrête l'affichage

Application.ScreenUpdating = False

...

Application.ScreenUpdating = False

'Ne pas afficher les messages

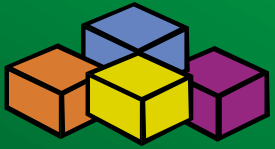
Application.DisplayAlerts = False

Sheets("Inutile").Delete

Sheets("Passé").Delete

Sheets("SansValeur").Delete

Application.DisplayAlerts = True



Array

VBA


```
Dim MaSem
```

```
MaSem = Array("Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun")
```

```
MsgBox MaSem(1) ' affiche "Tue" et non "Mon"!!!
```

Option Base 1

Dim MaSem

MaSem = Array("Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun")

MsgBox MaSem(1) ' affiche "Mon"

```
Sub agrandirArray()
```

```
    Dim A As Variant, B As Long, i As Long
```

```
    A = Array(10, 20, 30) ' A est une liste de 3 éléments indexés par défaut de 0 à 2
```

```
    B = A(2) ' B est maintenant égal à 30
```

```
    ReDim Preserve A(4) ' On étend la taille de A à 5 éléments
```

```
    A(4) = 40 ' On définit la valeur du cinquième élément
```

```
    For i = LBound(A) To UBound(A)
```

```
        Debug.Print "A(" & i & ") = " & A(i)
```

```
    Next i
```

```
End Sub
```

A(0) = 10

A(1) = 20

A(2) = 30

A(3) =

A(4) = 40

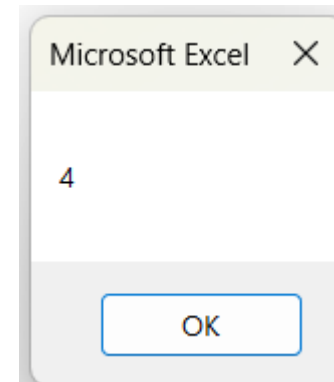
Sub découper()

Dim monArray() As String

monArray = Split("Les sanglot longs des violons", " ")

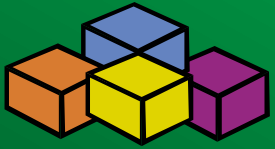
MsgBox UBound(monArray)

End Sub



```
Function trierMots(t)
    Dim myarray
    Dim i, j As Integer
    myarray = Split(t, " ")
    For i = LBound(myarray) To UBound(myarray)
        For j = i + 1 To UBound(myarray)
            If UCase(myarray(i)) > UCase(myarray(j)) Then
                Temp = myarray(j)
                myarray(j) = myarray(i)
                myarray(i) = Temp
            End If
        Next j
    Next i
    trierMots = Join(myarray, " ")
End Function
```

Les sanglots longs des violons de l'automne
de des l'automne Les longs sanglots violons



GOTO

VBA

```
Sub GotoDemo()  
    Dim Nombre, MonMessage  
    Nombre = 1  
    ' On oriente en fonction du nombre  
    If Nombre = 1 Then GoTo Ligne1 Else GoTo Ligne2  
  
    Ligne1:  
        MonMessage = "Nombre  gale 1"  
        GoTo Derni reLigne ' On va   Derni reLigne  
  
    Ligne2:  
        ' Cette partie n'est jamais ex cut e  
        MonMessage = "Nombre  gale 2"  
  
    Derni reLigne:  
        Debug.Print MonMessage '  crit "Nombre  gale 1" dans  
        ' la fen tre d'ex cution.  
  
End Sub
```

On Error Resume next

112

```
Sub Clear_All_Filters_Range()
```

```
'On utilise ShowAllData pour supprimer les filtres
```

```
'de la feuille. Mais une erreur apparaît s'ils sont déjà enlevés
```

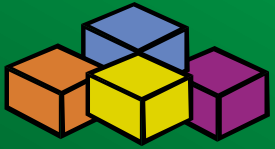
```
'(pour les tables utiliser: maTable.AutoFilter.ShowAllData)
```

```
On Error Resume Next
```

```
    ActiveSheet.ShowAllData
```

```
On Error GoTo 0
```

```
End Sub
```

Functions

VBA

UDF: Exemples de fonctions

114

```
Function tva(ht)
```

```
    tva = ht * 0.2
```

```
End Function
```

```
Function ttc(ht)
```

```
    ttc = ht + tva(ht)
```

```
End Function
```

```
Function age(ddn, Optional dateRef)
```

```
    If IsMissing(dateRef) Then
```

```
        age = (Date - ddn) / 365.25
```

```
    Else
```

```
        age = (dateRef - ddn) / 365.25
```

```
    End If
```

```
End Function
```



Systeme de fichiers

VBA

Choisir un dossier

116

```
Sub choisirDossier()
```

```
    'variable pour le chemin du dossier
```

```
    Dim cDossier As String
```

```
    'Ouvrir le sélecteur de fichier
```

```
    With Application.FileDialog(msoFileDialogFolderPicker)
```

```
        .Show
```

```
        cDossier = .SelectedItems(1)
```

```
    End With
```

```
    ' on écrit le chemin en A1
```

```
    Range("A1") = cDossier
```

```
End Sub
```

<https://learn.microsoft.com/fr-fr/office/vba/api/excel.application.filedialog>

Énumérer le contenu d'un dossier

117

```
Sub LoopThroughFiles()
```

```
    Dim oFSO As Object
```

```
    Dim oFolder As Object
```

```
    Dim oFile As Object
```

```
    Dim i As Integer
```

```
    Set oFSO = CreateObject("Scripting.FileSystemObject")
```

```
    Set oFolder = oFSO.GetFolder("C:\Documents\")
```

```
    For Each oFile In oFolder.Files
```

```
        Cells(i + 1, 1) = oFile.Name
```

```
        i = i + 1
```

```
    Next oFile
```

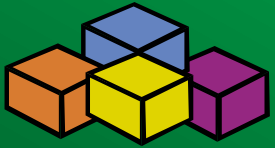
```
End Sub
```

Changer de dossier

118

ChDir

Permet de changer de dossier ("Directory")

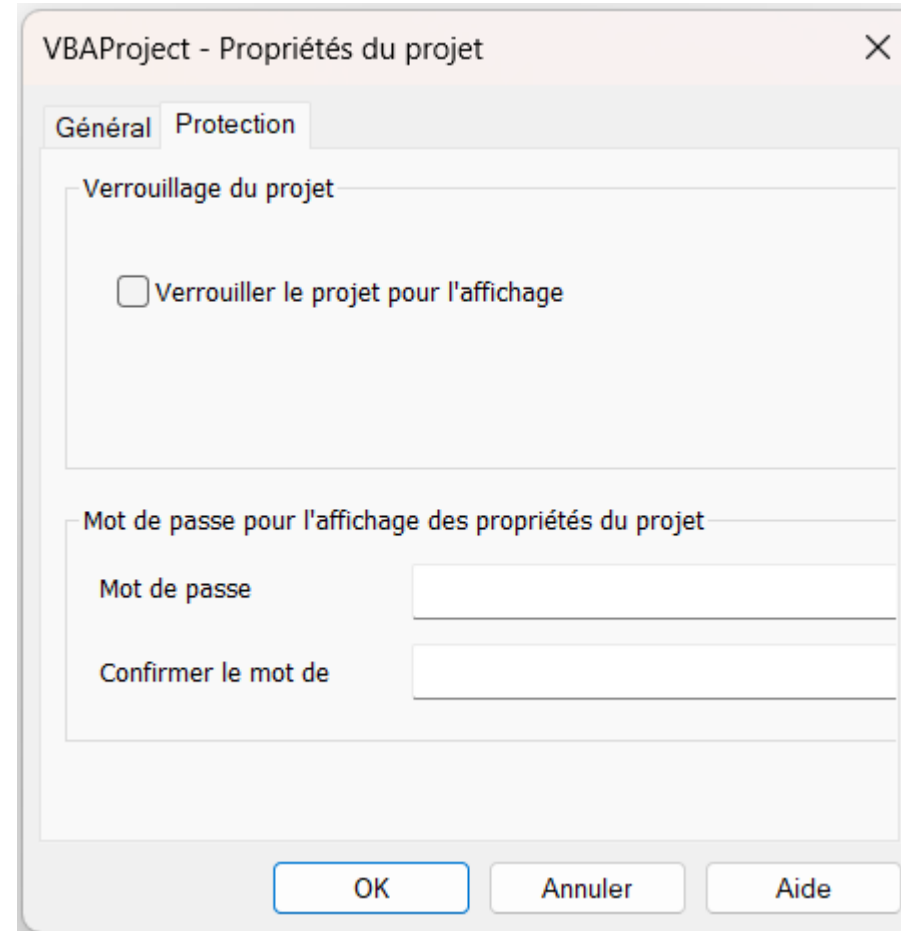
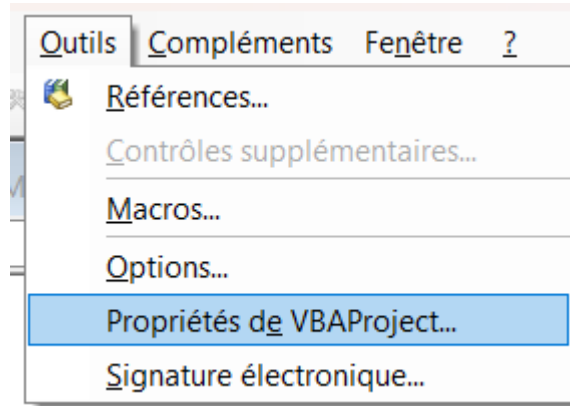


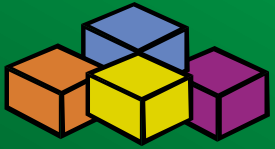
Protection

VBA

Mot de passe

120





Mots clés

STATIC

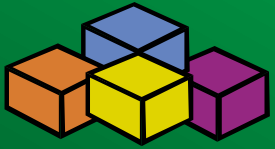
Tant que le classeur reste ouvert, la variable sera initialisée à sa dernière valeur à chaque fois que la macro sera exécutée

PUBLIC

La portée de la variable est globale à tous les modules standard du projet VBA

PRIVATE

La macro n'apparaît pas dans la liste des macros. Elle ne peut être lancée que par une autre macro.

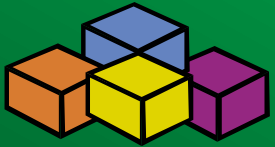


Formulaire

VBA



The image is a video thumbnail for an Excel VBA tutorial. At the top, there is a logo consisting of four 3D cubes in orange, blue, yellow, and purple. To the right of the logo, the text "EXCEL VBA" is written in green. Below this, a purple banner contains the text "MISSION USERFORM" in white. On the left side, a woman in a white shirt and black pants is giving a thumbs up. In the center, there is a screenshot of an Excel UserForm titled "UserForm2". The form contains the text "Indiquer les caractères des feuilles que vous voulez masquer", a text box labeled "Caractères", and two radio buttons labeled "Commencant" and "Finissant". At the bottom of the form are three buttons: "Annuler", "Masquer", and "Afficher". Overlaid on the right side of the form is a large yellow rounded rectangle with the text "Afficher" and "Masquer" in white. A large red play button icon is centered over the yellow rectangle.



Classes et Structures

Programmation objet

VBA

Dans le module de classe

```
Public name As String
```

```
Public Sub Hello()
```

```
    Debug.Print "Hello my name is " & name
```

```
End Sub
```

Dans un module standard

```
Sub Main()
```

```
    Dim class as New MyClass
```

```
    class.name = "John"
```

```
    class.Hello 'résultat: Hello my name is John
```

```
End Sub
```

<https://analystcave.com/vba-vba-class-tutorial/>

```
Private Structure employee
    Public givenName As String
    Public familyName As String
    Public phoneExtension As Long
    Private salary As Decimal
    Public Sub giveRaise(raise As Double)
        salary *= raise
    End Sub
    Public Event salaryReviewTime()
End Structure
```

<https://learn.microsoft.com/fr-fr/dotnet/visual-basic/programming-guide/language-features/data-types/how-to-declare-a-structure>

Contactez-moi sur

LinkedIn

François Parlant

Suivez-moi sur

YouTube

En parlant de bureautique