



FICHE DE COURS GRATUITE

VBA EXCEL | FORMULAIRES

Les contrôles d'un formulaire

Un bon formulaire guide l'utilisateur : une liste quand les réponses sont connues, une case à cocher pour un oui ou non, des boutons d'option pour un choix unique. Voici chaque contrôle, à quoi il sert et comment le lire en VBA.

Version	V1.2
Dernière mise à jour	29/10/2026
Niveau	3 / 3, Avancé
Durée de lecture	9 min
Auteur	Clément Delobel



Version en ligne
mesetudesdecas.fr/vba/contrôles-userform.htm
1

01 Définition

EN UNE PHRASE

Un contrôle est un **élément interactif** placé sur un UserForm. Chaque type a ses propriétés, dont la plus utilisée est **Value** : ce que l'utilisateur a saisi ou choisi.

02 Quel contrôle pour quel besoin ?

Contrôle	Préfixe	Usage	Propriété clé
Label	lbl	afficher un intitulé ou un message	Caption
TextBox	txt	saisie libre : nom, montant, commentaire	Value
ComboBox	cbo	liste déroulante, un seul choix	Value, ListIndex
ListBox	lst	liste visible, un ou plusieurs choix	Value, Selected
CheckBox	chk	oui ou non	Value (True / False)
OptionButton	opt	un choix parmi plusieurs	Value (True / False)
Frame	fra	regrouper des contrôles	Caption
CommandButton	btn	lancer une action	Caption

03 Remplir une liste

TROIS FAÇONS DE REMPLIR UNE COMBOBOX

```
Private Sub UserForm_Initialize()  
    ' 1. Élément par élément  
    Me.cboStatut.AddItem "En cours"  
    Me.cboStatut.AddItem "Terminé"  
  
    ' 2. D'un coup, depuis une plage  
    Me.cboVille.List = Worksheets("Listes").Range("A2:A30").Value  
  
    ' 3. Liée à une plage (propriété RowSource)  
    Me.lstProduits.RowSource = "Listes!B2:B50"  
End Sub
```

💡 ASTUCE

Interdire la saisie libre

Par défaut, une ComboBox accepte aussi un texte tapé au clavier. Pour obliger à choisir dans la liste, réglez sa propriété **Style** sur `fmStyleDropDownList`.

04 Lire les choix

CHECKBOX, OPTIONBUTTON ET COMBOBOX

```
If Me.chkNewsletter.Value = True Then  
    Cells(ligne, 5).Value = "Oui"  
End If  
  
If Me.optParticulier.Value Then  
    Cells(ligne, 6).Value = "Particulier"  
ElseIf Me.optProfessionnel.Value Then  
    Cells(ligne, 6).Value = "Professionnel"  
End If  
  
If Me.cboVille.ListIndex = -1 Then  
    MsgBox "Choisissez une ville." ' -1 = rien de sélectionné  
End If
```

05 Une ListBox à sélection multiple

Réglez la propriété **MultiSelect** sur `fmMultiSelectMulti`, puis parcourez la liste avec une **boucle For Next** en testant `Selected`.

RÉCUPÉRER TOUS LES ÉLÉMENTS COCHÉS

```
Dim i As Long, choix As String
For i = 0 To Me.lstProduits.ListCount - 1
    If Me.lstProduits.Selected(i) Then
        choix = choix & Me.lstProduits.List(i) & ", "
    End If
Next i
```

La numérotation d'une liste commence à **0** : le premier élément est `List(0)`.

06 Pièges à éviter

⚠ PIÈGE

Des `OptionButton` qui ne s'excluent pas

Tous les boutons d'option d'un formulaire forment un seul groupe. Pour avoir deux questions séparées, placez chaque groupe dans un **Frame**, ou donnez-leur une propriété **GroupName** différente.

⚠ PIÈGE

`RowSource` sans nom de feuille

`RowSource = "A2:A30"` lit la feuille active du moment. Indiquez toujours la feuille : `"Listes!A2:A30"`.

⚠ PIÈGE

Value d'une `ListBox` multiple

En sélection multiple, `Value` ne renvoie rien d'utile : il faut parcourir les éléments et tester `Selected(i)`.