

CRÉATION DE BOÎTES DE DIALOGUE

Un formulaire est une boîte de dialogue permettant de saisir des données qui seront stockées dans des variables ou des cellules d'une feuille de calcul. Le formulaire est donc très pratique pour communiquer des informations à un programme en cours d'exécution. De plus, celui-ci peut être personnalisé avec des zones de saisies, des cases à cocher, des listes déroulantes, des boutons de validation, etc.

LES BOÎTES DE DIALOGUE

AFFICHAGE D'UN MESSAGE

La commande `MsgBox` est un formulaire qui affiche une instruction, voire de stocker une réponse dans une variable :

```
' Boîte de dialogue avec juste un bouton OK
MsgBox "La mise à jour des données est terminée"
' Boîte de dialogue qui affiche le contenu des cellules A2 et A3 et un bouton OK
MsgBox Range("A2") & Chr(10) & Range("A3")
```

✓ L'instruction `Chr(10)` permet d'aller à la ligne

BOÎTE DE MESSAGE AVEC DES BOUTONS

Il est possible de définir deux boutons afin de poser une question à l'utilisateur.

Structure

`MsgBox(Texte à afficher, Type de bouton choisi, Titre de la boîte de message)`

Si l'on ne souhaite avoir qu'un bouton **OK**, il suffit de ne rien mettre comme type de bouton

Exemple

Pose une question en proposant de répondre en cliquant sur le bouton **OK** ou **Annuler**

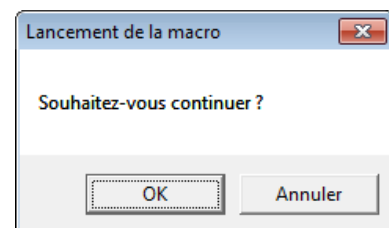
```
Choix = MsgBox("Souhaitez-vous continuer ?", vbOKCancel, "Lancement de la macro")
```

✓ Si le bouton **OK** est cliqué, la variable `Choix` sera égale à `vbOK`, si c'est le bouton **Annuler**, `Choix` sera égale à `vbCancel`.

Quelques exemples de boutons :

Le type de bouton peut être sous forme de texte ou sous forme de numéro :

- **vbOKCancel** ou **1** : boutons OK et Annuler renvoie `vbOK` (ou 1) ou `vbCancel` (ou 2)
- **vbYesNo** ou **4** : boutons Oui et Non renvoie `vbYes` (ou 6) ou `vbNo` (ou 7)
- **vbRetryCancel** ou **5** : boutons Réessayer et Annuler renvoie `vbRetry` (ou 4) ou `vbCancel` (ou 2)
- **vbAbortRetryIgnore** ou **2** : boutons Abandonner, Recommencer et Ignorer renvoie `vbAbort` (ou 3), `vbRetry` (ou 4) ou `vbIgnore` (ou 5)



Exemple de boîte de dialogue suivi de la confirmation du bouton choisi

Avec le nom des boutons :

```
Choix = MsgBox("Voulez-vous confirmer votre choix ?", vbOKCancel, "Lancement ")  
If Choix = vbOK Then MsgBox "Choix validé"  
If Choix = vbCancel Then MsgBox "Choix annulé"
```

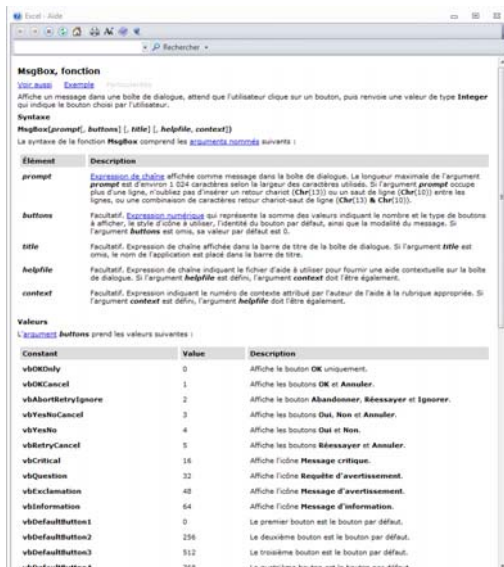
Avec le numéro des boutons :

```
Choix = MsgBox("Voulez-vous confirmer votre choix ?", 1, "Lancement ")  
If Choix = 1 Then MsgBox "Choix validé"  
If Choix = 2 Then MsgBox "Choix annulé"
```

POUR AVOIR LA LISTE DES CODES DES BOUTONS DANS L'AIDE D'EXCEL VBA

1. Dans son code VBA, cliquer sur *MsgBox* dans une ligne de son code
2. Appuyer sur la touche du clavier F1

✓ Le code des touches apparaît dans la fenêtre d'aide :



BOÎTE DE DIALOGUE DEMANDANT DE SAISIR UNE RÉPONSE

La structure est très proche de la commande MsgBox mais cette fois-ci, c'est la réponse saisie par l'utilisateur qui sera stocké dans une variable :

STRUCTURE

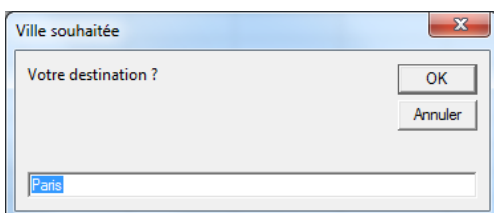
InputBox(Texte de la boîte de dialogue, Titre de la boîte de dialogue, Réponse par défaut affichée dans la zone de texte)

✓ Le titre et la réponse par défaut sont facultatifs.

EXEMPLE

```
Choix = InputBox("Votre destination ?", "Ville souhaitée", "Paris")
```

Stocke dans la variable Choix la réponse saisie dans la zone de texte



CRÉATION DE FORMULAIRES PERSONNALISÉS

Excel VBA permet également de créer ses propres formulaires en personnalisant les différents contrôles — zones de texte, listes déroulantes, boutons, etc. — et en leur affectant différentes procédures.

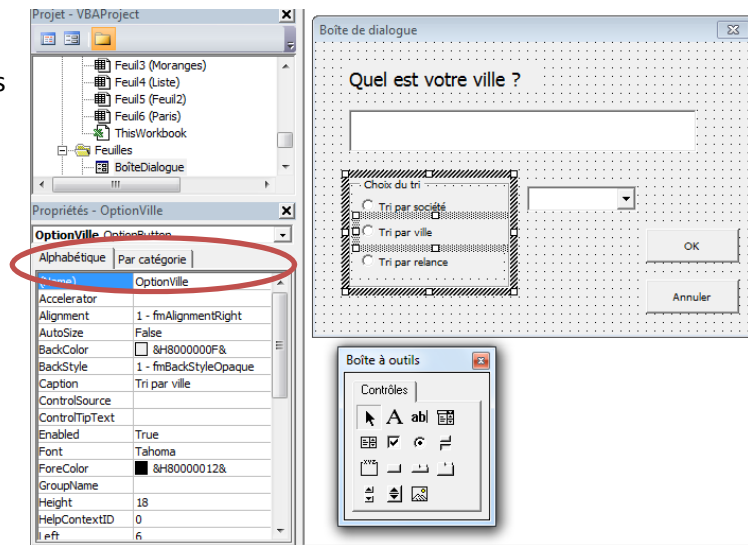
3. Dans l'éditeur de VBA : Menu Insertion
UserForm

4. Modifier le nom du formulaire dans le volet des propriétés, à gauche :

- (Name) : donner un nom à son formulaire (sans espace)
Exemple : Formulaire01
- Caption : Titre de l'en-tête du formulaire

5. Dessiner les boutons voulus :

- Boutons de commandes pour le bouton OK, Annuler, etc.



LES TYPES DE CONTRÔLES

1. Dans la boîte à outil, cliquer sur le contrôle voulu
2. Dans son formulaire, tracer un cadre qui contiendra le contrôle choisi



Fonctions des différents contrôles

	Sélectionner les objets	Permet de sélectionner les contrôles déjà créés dans le formulaire
	Intitulé	Étiquette de texte libre
	Zone de texte	Saisie d'un texte libre par l'utilisateur
	Zone de liste modifiable	Fait apparaître une liste déroulante pour choisir une option
	Zone de liste	Cadre contenant une liste déroulante
	Case à cocher	Permet de cocher une ou plusieurs options ou aucune (mettre les différentes cases dans un cadre)
	Bouton d'option (ou bouton radio)	Permet de ne choisir qu'une option et une seule (mettre les différents boutons dans un cadre)
	Bouton Bascule	Ne contient que deux possibilités (activé ou désactivé)
	Cadre	Permet d'accueillir les cases à cocher ou les boutons d'option
	Bouton de commande	Pour créer des boutons OK, Annuler, etc.
	Contrôle onglet	Choisi une valeur en fonction de l'onglet cliqué (faire un clic droit sur l'un des onglets pour créer plusieurs onglets)
	Multipage	Permet de créer des onglets de page qui vont contenir des contrôles (en faisant un clic droit sur l'un des onglets)
	Barre de défilement	Pour faire évoluer une valeur entre deux extrêmes



Toupe

Propriétés Max et Min pour saisir les valeurs maximales et minimales possibles et SmallChange pour l'incrément (valeur de l'évolution)



Image

Insert une image dans le formulaire
(pratique pour y placer un logo)

- ✓ Les options des listes déroulantes, des valeurs par défaut, etc. sont paramétrées par des codes VBA placés dans la procédure `_Initialize` du formulaire (cf. ci-dessous)

POUR FAIRE RÉAPPARAÎTRE LA BOÎTE À OUTIL SI ELLE DISPARAÎT

1. Cliquer une fois sur le formulaire
2. Menu Affichage
Boîte à outils

CHANGER LE NOM D'UN CONTRÔLE

1. Cliquer sur le contrôle voulu (bouton, liste déroulante, zone de texte, etc.)
 2. Dans le volet des propriétés, à gauche, dans la case (Name), saisir le nom voulu (exemple : BoutonOK, etc.) sans espace.
- ✓ La propriété `TabIndex` permet de modifier l'ordre d'accès à un contrôle de formulaire en appuyant sur la touche Tabulation (Mettre 0 pour le premier contrôle voulu puis 1 pour le suivant et ainsi de suite).

UTILISATION DES ÉVÉNEMENTS DES CONTRÔLES DU FORMULAIRE

OUVERTURE DU FORMULAIRE

Créer la macro suivante :

```
Sub Ouverture()  
    Formulaire01.Show  
End Sub
```

(ou `Formulaire01.Show False` pour pouvoir continuer à accéder à la feuille de calcul)

DÉFINITIONS DES VALEURS DES CONTRÔLES À L'OUVERTURE DU FORMULAIRE

1. Dans l'éditeur des codes VBA, cliquer deux fois sur le formulaire

La procédure suivante apparaît :

```
Private Sub UserForm_Click()  
  
End Sub
```

2. Remplacer `Click` par `Initialize`

```
Private Sub UserForm_Initialize()  
  
End Sub
```

3. Saisir, dans cette procédure, les valeurs que prendront des différents contrôles à l'ouverture du formulaire, exemple :

```
Private Sub UserForm_Initialize()  
    ZoneVille = "Bordeaux"  
End Sub
```

- ✓ Le contrôle `ZoneVille` proposera la valeur `Bordeaux` à l'ouverture du formulaire.

QUELQUES EXEMPLES DE VALEURS POUR DES BOUTONS :

Pour définir les valeurs des boutons, celles-ci doivent être saisies dans la procédure `Private Sub UserForm_Initialize()`

Liste déroulante

```
' Liste déroulante en fonction des cellules d'une feuille
For i = 2 To 100
    If Cells(i, 1) = "" Then Exit For
    Liste_Sociétés.AddItem Sheets(1).Cells(i, 1)
Next i
```

Boutons radio

☞ Avant : tracer un cadre qui va contenir les boutons radios (afin qu'un seul soit actif)

```
OptionSociété.Value = True
```

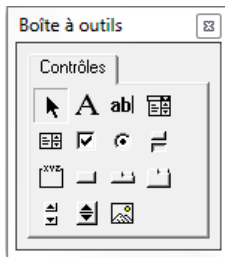
✓ Ainsi, seul le bouton appelé `OptionSociété` sera coché à l'ouverture du formulaire

ÉVÈNEMENTS LORS DE L'ACTIVATION D'UN CONTRÔLE DU FORMULAIRE

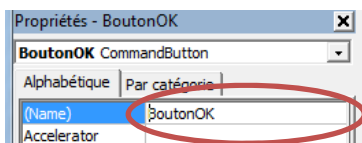
CRÉATION D'UN BOUTON OK

En général, c'est en cliquant sur le bouton **OK** que vont être déclenchés les événements — modification du contenu de cellules, lancement de macros, etc. — liés aux options choisies dans les contrôles du formulaire ainsi que la fermeture du formulaire

1. Cliquer sur le bouton Commande



2. Dans son formulaire, dessiner un cadre qui va contenir le bouton OK
3. Cliquer deux fois sur son texte pour saisir « OK »
4. Renommer ce bouton en `BoutonOK` dans le volet des propriétés



5. Cliquer deux fois sur ce bouton pour saisir les commandes de codes à effectuer, exemple :

```
Private Sub BoutonOK_Click()
    Range("A2").Value = ZoneVille ' La cellule A2 reçoit le contenu de ZoneVille
    ' Ces trois lignes déclenchent une macro suivant le bouton d'option choisi
    If OptionSociété Then Tri_Société_Ville
    If OptionVille Then Tri_Ville_Société
    If OptionRelance Then Tri_Relance
    ChoixTri.Hide ' Masque le formulaire de saisie (mais conserve les valeurs)
End Sub
```

✓ Ces actions se déclencheront à la fermeture du formulaire, en cliquant sur le bouton **OK**

POUR DÉCLENCHER UNE ACTION DÈS QU'UN CONTRÔLE EST MODIFIÉ (SANS FERMER LE FORMULAIRE) :

1. Cliquer deux fois sur le contrôle voulu
2. Saisir le code des commandes à réaliser, exemple :

```
Private Sub Liste_Sociétés_Change()  
    Range("A3").Value = Liste_Sociétés  
End Sub
```

- ✓ Dès qu'un élément de Liste_Sociétés est choisi, la cellule A3 reçoit la valeur choisie dans Liste_Sociétés alors que le formulaire est toujours à l'écran

CRÉATION D'UN BOUTON ANNULER

Un bouton peut fermer la boîte de dialogue et vider les variables mémoires (Unload), afin que les valeurs saisies dans les contrôles du formulaire ne réapparaissent pas à sa prochaine ouverture.

- ✓ La création du bouton Annuler se fait comme pour le bouton OK, à partir du bouton Commande.

```
Private Sub BoutonAnnuler_Click()  
    Unload ChoixTri  
End Sub
```

- ✓ Ou écrire *ChoixTri.Hide* pour que les valeurs choisies restent lors de la prochaine réouverture du formulaire.

LIENS SUR LES FORMULAIRES VBA

Site en anglais expliquant les paramètres des contrôles des formulaires :

fring.developpez.com/vba/excel/msgbox : sur les boîtes de message

www.excel-pratique.com/fr/vba/userform.php : pour débiter

www.excel-easy.com/vba/userform.html : plus complet mais en anglais

<http://jml85.pagesperso-orange.fr/Pages/Excel/Cours/UserForm/Cours16.html> : explication sur les différentes options des contrôles des formulaires ■