

A. Objectif du chapitre

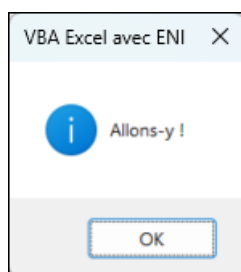
Afin de mieux vous équiper pour communiquer avec les utilisateurs, et ce sans avoir besoin d'écrire des centaines de lignes de code, vous allez découvrir dans ce chapitre les fonctions **MsgBox** et **InputBox**. L'affichage de message aux utilisateurs peut être à plusieurs vocations, comme lui souhaiter la bienvenue, l'informer sur le processus de votre programme mais également pour le prévenir en cas d'étapes critiques.

Ce chapitre passera ainsi en revue la fonction **MsgBox**, avec les boutons et les icônes que vous pouvez ajouter à vos boîtes de dialogue, et la fonction **InputBox**, avec les différentes options qui s'offrent à vous lorsque vous souhaitez interagir plus spécifiquement avec vos utilisateurs.

En fin de chapitre, vous trouverez une série d'exercices pour vous entraîner à adapter au mieux votre communication au travers des outils VBA.

B. Afficher un message à l'utilisateur : MsgBox

Quel que soit le langage informatique, il est toujours important d'afficher des informations à l'utilisateur, et d'obtenir de sa part une confirmation ou une affirmation. En VBA, cette fonction d'affichage de message se nomme **MsgBox**, elle se prononce « Message Box ».



La fonction **MsgBox** est probablement la fonction qui est la plus utilisée dans les fonctions VBA disponibles. Il est important d'en connaître l'écriture, mais également les options qui s'offrent à vous lorsque vous l'utilisez.

1. Affichage élémentaire avec MsgBox

- Après avoir ajouté un module dans lequel vous allez pouvoir écrire votre code, commencez par taper ces quelques lignes :

```
Sub Salut()  
'Afficher Salut à l'utilisateur  
End Sub
```

Exemple 1 : écriture d'une première macro

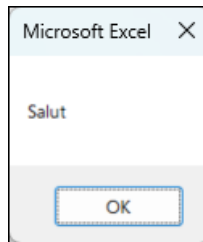
Les mots-clés **Sub** et **End Sub** seront abordés dans le chapitre Procédures, fonctions et macros. Pour l'instant votre code ne contient rien d'autre qu'un commentaire. Si vous exécutez la macro, il ne se passera rien.

Pour afficher **Salut** à l'utilisateur, il suffit d'ajouter l'instruction suivante pour obtenir le code complet suivant :

```
Sub Salut()  
'Afficher Salut à l'utilisateur  
MsgBox "Salut"  
End Sub
```

Exemple 2 : appel de la fonction MsgBox

- Exécutez la macro, vous devriez avoir sur votre écran le message suivant qui s'affiche :



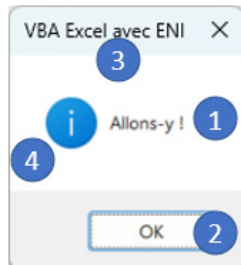
Félicitations, vous avez écrit et exécuté votre première macro ! Continuez dès à présent pour découvrir comment faire mieux.

2. Propriétés et syntaxe générale de la boîte de dialogue MsgBox

Lorsque vous souhaitez afficher un message, il vous faut donc écrire la fonction **MsgBox**, suivie du texte que vous souhaitez afficher. Le texte est compris entre guillemets.

Par défaut, la fenêtre affiche un seul bouton OK et a pour titre Microsoft Excel.

Les quatre parties qui vont vous intéresser dans les prochains paragraphes sont représentées dans l'image suivante.



Le texte est l'information principale qui est affichée dans la boîte de dialogue (1). Vous disposez ensuite d'un ou plusieurs boutons dans la partie inférieure de la boîte (2). En haut de la boîte se trouve le titre (3). Enfin, vous pouvez voir apparaître à la gauche du texte ce qu'on appelle une icône (4).

Chacune de ces propriétés sera manipulable par votre instruction, au travers de l'appel à la fonction MsgBox et des informations que vous fournirez à cette fonction.

a. Syntaxe générale de la fonction MsgBox

La syntaxe générale de la fonction MsgBox est la suivante :

```
MsgBox (MessageAffiche, [ boutons, ] [ titre, ]  
[ fichieraide, contexte ])
```

Exemple 3 : syntaxe générale de la fonction MsgBox

La première information qui est fournie à la fonction, que l'on appelle un **paramètre** ou un **argument**, est le texte qui sera affiché dans la boîte de dialogue.

Les autres informations, dont les noms sont par convention dans cet ouvrage entre crochets, sont facultativement fournies lors de l'appel.

Les informations lors de l'appel de la fonction MsgBox sont séparées par des virgules (.). Le caractère virgule est celui qui sépare les informations en VBA.

Vous allez voir dans les sections suivantes les valeurs qui peuvent être fournies pour chacune des informations.

3. Texte et affichage sur plusieurs lignes

L'information principale de la boîte de dialogue est le texte, il est obligatoirement fourni lors de l'appel de la fonction MsgBox.

```
MsgBox "Salut" 'Affiche Salut
```

Exemple 4 : affichage d'un texte codé en dur

Vous pouvez également utiliser le contenu d'une cellule :

```
MsgBox Range("B5").Value 'Affiche la valeur contenue dans la cellule B5
```

Exemple 5 : affichage du contenu d'une cellule

Ou encore, utiliser le contenu d'une variable (cette notion sera abordée plus en détail dans le chapitre Les variables, les constantes):

```
Dim MonTexte As String
MonTexte = "Texte personnalisé"
MsgBox MonTexte
```

Exemple 6 : affichage de la valeur d'une variable

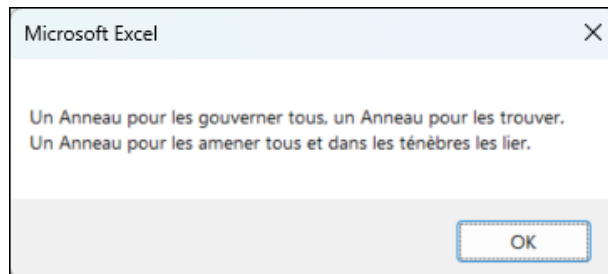
a. Affichage sur plusieurs lignes, constante vbCrLf

Il est possible d'afficher du texte sur plusieurs lignes. Pour passer à la ligne, il vous faut utiliser le mot-clé **vbLf**, qui est une constante VB, entouré du symbole & (qui se nomme esperluette), qui sert à la concaténation. Les constantes seront abordées dans la section Les variables, les constantes et les chaînes de caractères dans le chapitre Manipuler les chaînes de caractères.

```
MsgBox "Un Anneau pour les gouverner tous, un Anneau pour les trouver."
& vbCrLf & _
      "Un Anneau pour les amener tous et dans les ténèbres les lier."
```

Exemple 7 : afficher du texte sur plusieurs lignes

En exécutant le code de l'exemple 7, la fenêtre de message suivante s'affichera.



4. Les boutons

Le second élément qui vous intéresse lorsque vous affichez une boîte de dialogue est l'ensemble de boutons que l'utilisateur verra à l'écran. Qu'il y ait un, deux ou trois, voire quatre boutons, ce sont autant de possibilités qui s'offrent à l'utilisateur.

En partant d'une syntaxe similaire à l'ensemble des boîtes qui s'afficheront, votre code sera le suivant :

```
MsgBox "Message affiché", CombinaisonDeBoutons, "Titre de la boîte"
```

Exemple 8 : ensemble commun aux affichages de boutons

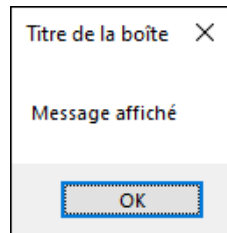
a. Affichage d'un seul bouton

Lorsque votre boîte de dialogue ne comporte qu'un bouton, celui-ci affichera le texte OK. C'est le comportement par défaut. Vous disposez de deux syntaxes possibles pour ce cas de figure :

```
MsgBox "Message affiché", , "Titre de la boîte" 'aucune information  
fournie, VBA appliquera la valeur par défaut
```

```
MsgBox "Message affiché", vbOKOnly, "Titre de la boîte" 'le bouton OK  
seul est représenté par la valeur constante vbOkOnly
```

Exemple 9 : affichage d'une boîte de dialogue avec un seul bouton



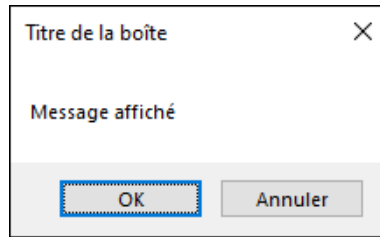
b. Affichage de deux boutons

Il existe trois ensembles offrant deux boutons à l'utilisateur.

Le premier ensemble comporte les boutons OK et **Annuler**. Vous pouvez afficher cet ensemble de boutons avec la valeur **vbOKCancel**.

```
MsgBox "Message affiché", vbOKCancel, "Titre de la boîte" 'OK + Annuler
```

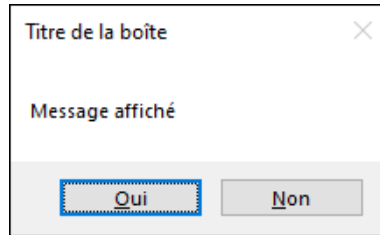
Exemple 10 : affichage des boutons OK et Annuler



Le deuxième ensemble comporte les boutons **Oui** et **Non**. Vous pouvez afficher cet ensemble de boutons avec la valeur **vbYesNo**.

```
MsgBox "Message affiché", vbYesNo, "Titre de la boîte" 'Oui + Non
```

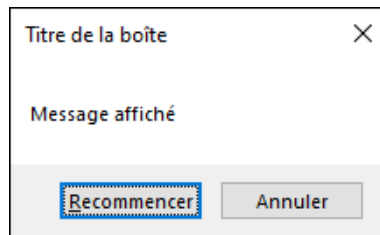
Exemple 11 : affichage des boutons Oui et Non



Le troisième ensemble comporte les boutons **Recommencer** et **Annuler**. Vous pouvez afficher cet ensemble de boutons avec la valeur **vbRetryCancel**.

```
MsgBox "Message affiché", vbRetryCancel, "Titre de la boîte"  
'Recommencer + Annuler
```

Exemple 12 : affichage des boutons Recommencer et Annuler



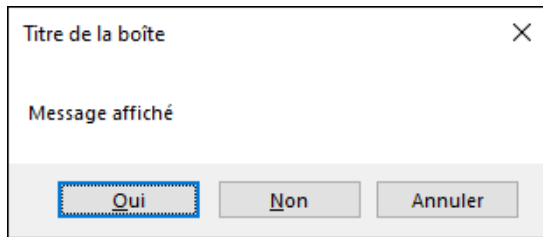
c. Affichage de trois boutons

Il existe également deux ensembles présentant des boîtes de dialogue avec trois boutons.

Le premier ensemble affiche les boutons **Oui**, **Non** et **Annuler**. Vous pouvez afficher cet ensemble de boutons avec la valeur **vbYesNoCancel**.

```
MsgBox "Message affiché", vbYesNoCancel, "Titre de la boîte" 'Oui + Non
+ Annuler
```

Exemple 13 : affichage des boutons *Oui, Non et Annuler*

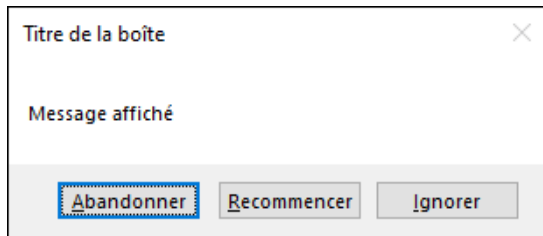


Enfin, le dernier ensemble de cette liste affiche quant à lui les boutons **Abandonner**, **Recommencer** et **Ignorer**.

Cet ensemble est représenté par la valeur **vbAbortRetryIgnore**.

```
MsgBox "Message affiché", vbAbortRetryIgnore, "Titre de la boîte"
'Abandonner + Recommencer + Ignorer
```

Exemple 14 : affichage des boutons *Abandonner, Recommencer et Ignorer*



d. Tableau récapitulatif des boutons possibles

Pour vous y retrouver, voici les constantes à utiliser ainsi que les valeurs numériques associées à chacune de ces constantes.

Constante	Boutons affichés	Valeur numérique
vbOkOnly	OK seulement	0
vbOkCancel	OK et Annuler	1
vbAbortRetryIgnore	Abandonner, Recommencer et Ignorer	2
vbYesNoCancel	Oui, Non et Annuler	3
vbYesNo	Oui et Non	4
vbRetryCancel	Recommencer et Annuler	5