

## Découvrir le VBA Excel (article invité)

*Suite aux questions que me posent certains abonnés sur le VBA, j'ai proposé à [Thierry Courtot](#) du blog [Excel-plus !](#) d'écrire un article invité à ce sujet 😊.*

*Contrôleur de gestion, passionné par Excel qui est mon outil de travail principal, je souhaite partager mes connaissances et en apprendre encore sur ce logiciel ultra diffusé et devenu une référence dans l'entreprise et dans le quotidien de tous. Également curieux de beaucoup de choses, je vis mes passions ou lubies du moment et, dans la mesure du possible, les partage avec vous. Blogueur à temps partiel depuis 4 ans, j'aime diffuser ce que j'ai appris.*

### Qu'est-ce que le code VBA ?

Très souvent, quand on commence à parler de VBA (ou Visual Basic for Applications), beaucoup de gens se sentent rebutés et estiment, à tort, que ce n'est pas dans leurs cordes. Bien sûr, le langage de programmation qu'est le VBA est un véritable code normé et qui ne s'invente pas. Les experts en VBA ont réellement des compétences certaines et peuvent ainsi créer, plus que des macros, de véritables applications avec Excel, jusqu'à pouvoir en modifier l'apparence pour l'utilisateur.

Mais même si vous ne souhaitez pas créer d'application à vendre sur le marché des outils informatiques, vous pouvez découvrir et utiliser ce langage pour vos propres fichiers.

Pour vous le démontrer, nous allons créer ensemble une macro, la modifier dans Visual Basic et, peut-être, vous donner l'envie d'en découvrir plus.

Je vous propose donc de créer une macro qui vous demandera de supprimer les données d'une plage de cellules en vous demandant de confirmer par *OUI* ou *NON*. Dans cet exemple, nous utiliserons des fonctions qu'il vous faudra connaître pour utiliser le VBA, et d'autres astuces pour éviter d'avoir à en apprendre trop.

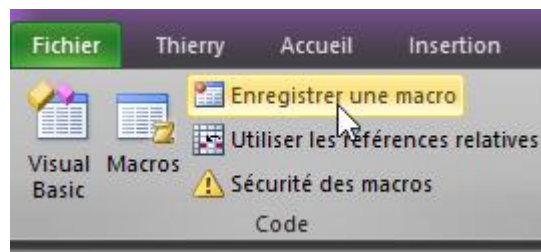
Vous retrouverez le fichier utilisé, avec la macro que nous allons créer, au bas de l'[article](#).

## Création d'une macro via l'enregistreur de macro d'Excel

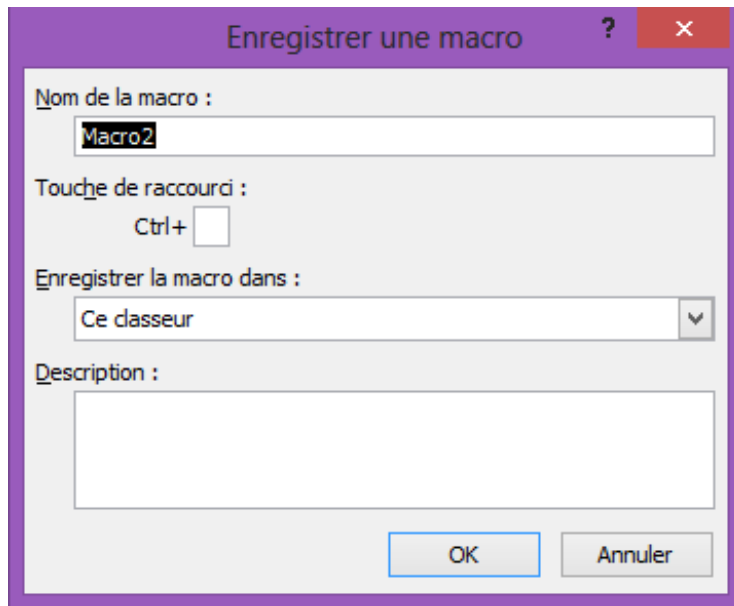
Prenons donc un tableau simple, une liste de noms avec des codes affectés. Les noms devant changer régulièrement, nous voulons pouvoir les supprimer sur demande en cliquant sur un simple bouton.

|    | A    | B          |  |
|----|------|------------|--|
| 1  | Code | Nom        |  |
| 2  | 1    | Jean-Luc   |  |
| 3  | 2    | Clémentine |  |
| 4  | 3    | Emeline    |  |
| 5  | 4    | Fabien     |  |
| 6  | 5    | Serge      |  |
| 7  | 6    | René       |  |
| 8  | 7    | Jeanine    |  |
| 9  | 8    | Françoise  |  |
| 10 | 9    | Lucien     |  |
| 11 | 10   | Fabrice    |  |

Nous allons commencer par enregistrer l'action de la suppression. En effet, plutôt que de taper le code de cette action, il est plus rapide de l'exécuter en utilisant l'**Enregistreur de macros** que vous trouvez dans l'onglet **Développeur** du ruban (à rajouter via les options si vous ne le voyez pas, car masqué par défaut).

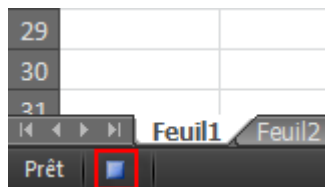


Une fois cliqué sur ce bouton, Excel vous ouvre une fenêtre pour lancer l'enregistrement.



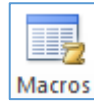
Vous pouvez donner un nom à votre macro ou le laisser en automatique, l'important étant de la retrouver plus tard. Cliquez sur **Ok** et l'enregistrement de vos faits et gestes commence.

Vous allez donc sélectionner les cellules de *B2* à *B11* puis taper sur la touche *Suppr* de votre clavier. Dans le coin gauche, en bas de votre écran, vous verrez un bouton **STOP** apparaître dès le début, cliquez dessus pour arrêter l'enregistrement.

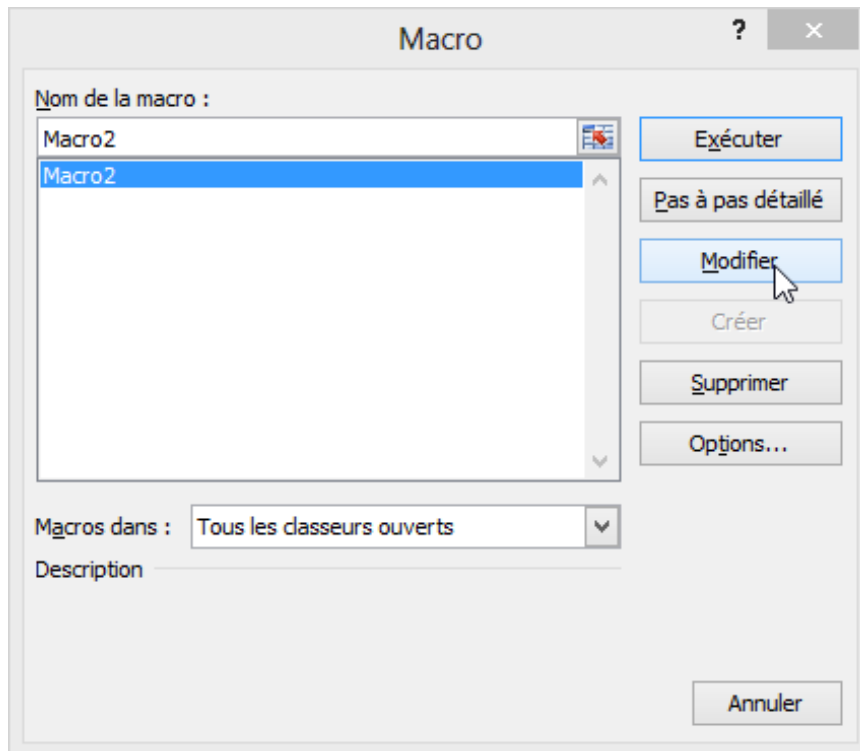


Voilà, la première partie de votre macro est créée. Allons maintenant la modifier.

## Modification d'une macro enregistrée avec l'enregistreur de macro



Pour cela, vous cliquerez sur le bouton **Macros** qui se trouve près du bouton pour lancer l'enregistrement (toujours dans l'onglet **Développeur**), sélectionnez la *Macro2* que vous venez de créer et cliquer sur **Modifier**.



Excel lance alors **Visual Basic** et vous voyez apparaître le code de votre *Macro 2*. C'est ce code que nous allons maintenant améliorer pour obtenir la validation *OUI* ou *NON* que nous souhaitons mettre en place.

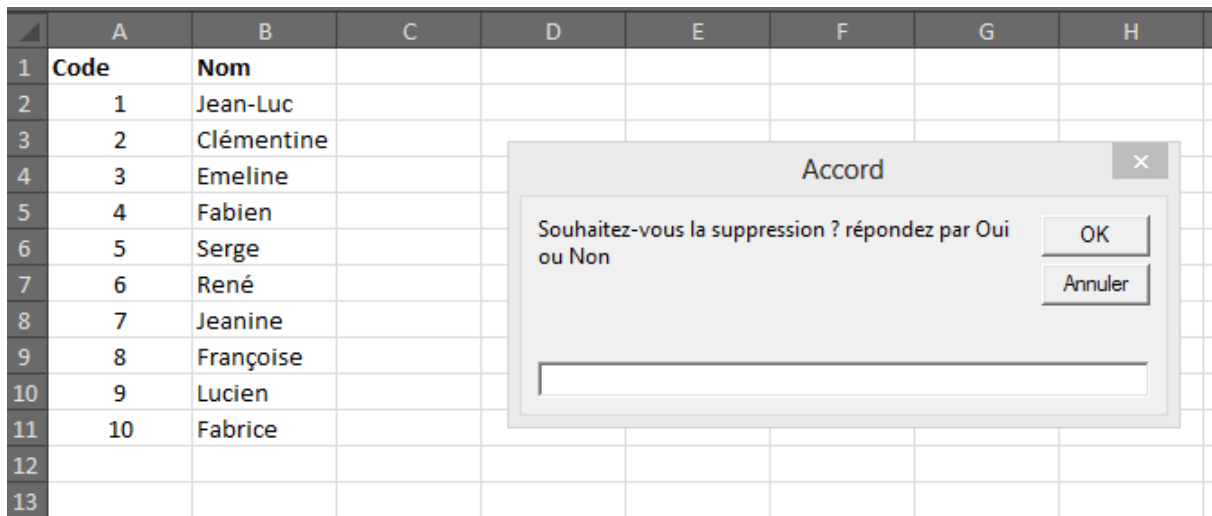
```
(Général)
Sub Macro2 ()
'
' Macro2 Macro
'
'
'
Range ("B2:B11") .Select
Selection.ClearContents
End Sub
```



Votre macro est désormais terminée. Vous remarquerez la ligne qui commence par **IF** et qui demande si la variable est égale à *OUI*, alors il exécute le code. Cette tranche de code se termine obligatoirement par un **END IF** qui indique au code que les conditions sont finies.

### Lancement d'une macro Excel

Il vous reste à la tester. Revenez alors sur votre feuille Excel et cliquez sur le bouton **Macros**. Au lieu de sélectionner **Modifier**, cliquez sur exécuter. Vous obtiendrez alors une boîte de dialogue dans laquelle vous entrerez *Oui* ou *Non*.



À vous de tester les deux possibilités pour valider votre macro.

Dernière précision, au moment d'enregistrer votre fichier, il vous faudra sélectionner un nouveau type de fichier .xlsm qui définit le format des macros.

Et voilà, vous avez créé votre première macro en VBA et j'espère vous avoir amusé, et qui sait, rendu curieux pour en découvrir encore plus.

Retrouvez d'autres tutos concernant Excel sur [Excel-Plus !](http://Excel-Plus.fr) et suivez Thierry sur les réseaux :

- Facebook : <https://www.facebook.com/ExcelPlus.fr>
- Twitter : <https://twitter.com/ExcelPlus>

**Tutoriel réalisé avec Excel 2013**

**[Voir la version vidéo de cet article](#)**