



# Introduction aux macros et au Basic

---

## Sommaire

Introduction.....	2
Présentation.....	3
Qu'est ce qu'une macro ?.....	3
Organisation des macros.....	4
Ma première macro.....	7
Enregistrement de la macro.....	8
Stockage des macros.....	9
Exécution de la macro.....	10
Assignation d'un raccourci clavier.....	11
Edition de la macro.....	12
Ma seconde macro.....	15
Enregistrement de la macro.....	15
Assignation d'un bouton en barre d'outils.....	17
Personnaliser la macro.....	19
Conclusion.....	22

## Introduction

Les tutoriels « *Formation Express* » vous permettent de vous familiariser rapidement avec des concepts ou des fonctions de la suite bureautique libre **OpenOffice.org** version 2.0 minimum.

Chaque tutoriel « *Formation Express* » aborde un thème précis et vous donne les informations de base qui vous seront nécessaires pour utiliser efficacement le module concerné, le tout en un minimum de pages.

Ce tutoriel est libre de droit pour une utilisation dans un cadre privé ou non commercial. Vous ne devez pas le modifier sans l'autorisation écrite de son auteur. Pour un usage dans un but commercial, reportez-vous aux [conditions générales d'utilisation](#).

Toute mise à disposition de ce tutoriel sur un autre site que le [forum francophone de OpenOffice.org](#) est soumise à autorisation de l'auteur.

De la même manière, je n'assure aucune aide directe, ni support sur des questions d'utilisation ou de compréhension de ce tutoriel. Il est donc inutile de me contacter, je vous invite plutôt à vous reporter directement sur le [forum francophone de OpenOffice.org](#) en indiquant que vous demandez de l'aide à propos du manuel « Introduction aux macros et au Basic ».

Les marques et noms de société éventuellement cités dans ce support sont déposés par leurs propriétaires respectifs.

Je ne suis lié avec aucun éditeur ou constructeur informatique.

Ce tutoriel a été entièrement réalisé avec la suite bureautique libre **OpenOffice.org** 2.0 (disponible gratuitement sur [le site officiel du projet OOo](#)) qui permet d'exporter nativement en PDF.

### Avertissement complémentaire :

Les éléments (données ou formulaires) éventuellement inclus dans ce support vous sont fournis à titre d'exemple uniquement. Leur utilisation peut avoir, dans certains cas, des conséquences matériels et juridiques importantes qui peuvent varier selon le sujet dont ils traitent. Il est recommandé d'être assisté par une personne compétente en informatique ou de consulter un conseiller juridique ou financier avant de les utiliser ou de les adapter à votre activité.

## Présentation

Ce manuel vous donne des informations de base concernant les macros et son langage de programmation le Basic sous **OpenOffice.org**. Vous apprendrez ce qu'est une macro et comment les créer et les modifier.



### Note :

J'utilise **OOo** sous Windows XP. Toutes les copies écran sont donc issues de cette plateforme. Je vous invite également à vous procurer **la dernière version** car de nombreuses bogues sont corrigées au fur et à mesure.

Pour des raisons de lisibilité, j'emploie indifféremment le sigle **OOo** en lieu et place de la terminologie **OpenOffice.org**.

### *Qu'est ce qu'une macro ?*

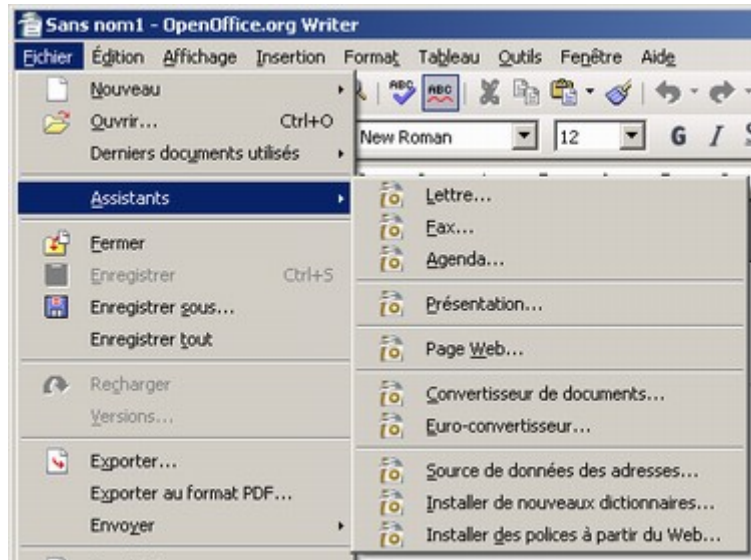
Si l'on devait définir ce qu'est une macro, on pourrait dire qu'il s'agit d'une méthode visant à grouper des tâches en une seule commande facile à utiliser. Le terme *macro* est utilisé dans le même contexte qu'en économie où on parle souvent de macro-économie. Ainsi une macro se réfère à quelque chose de grand ou de volumineux.

Ce terme est donc utilisé dans ce sens car une macro rassemble une série de petites fonctions de base au sein d'une opération beaucoup plus grande. On appelle cela une macro-commande.

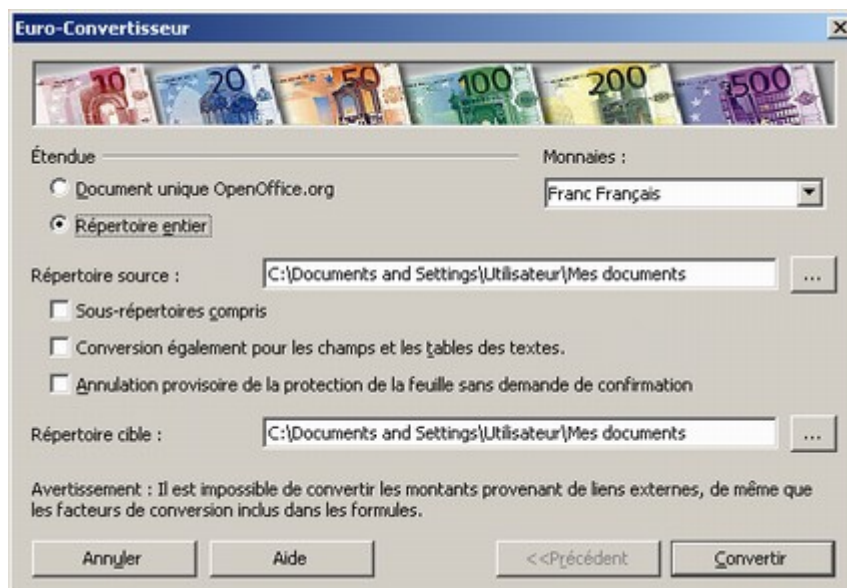
On peut également considérer que OOo est lui-même un programme composé de macro-commandes. C'est une définition un peu simpliste mais les différentes routines qu'appellent les menus ou les boutons ne sont en fait rien de plus que des macros internes au logiciel.

## Organisation des macros

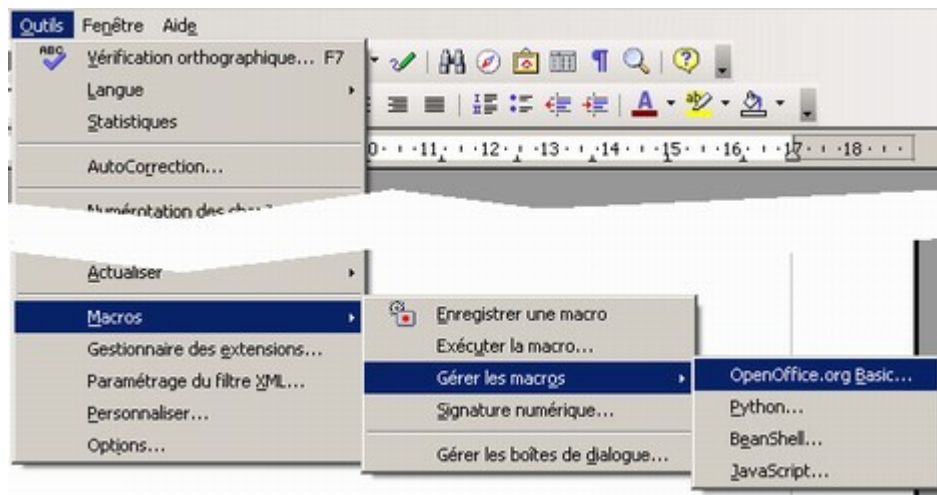
Prenons l'exemple des différents assistants proposés dans Writer comme celui de l'Euro-convertisseur :



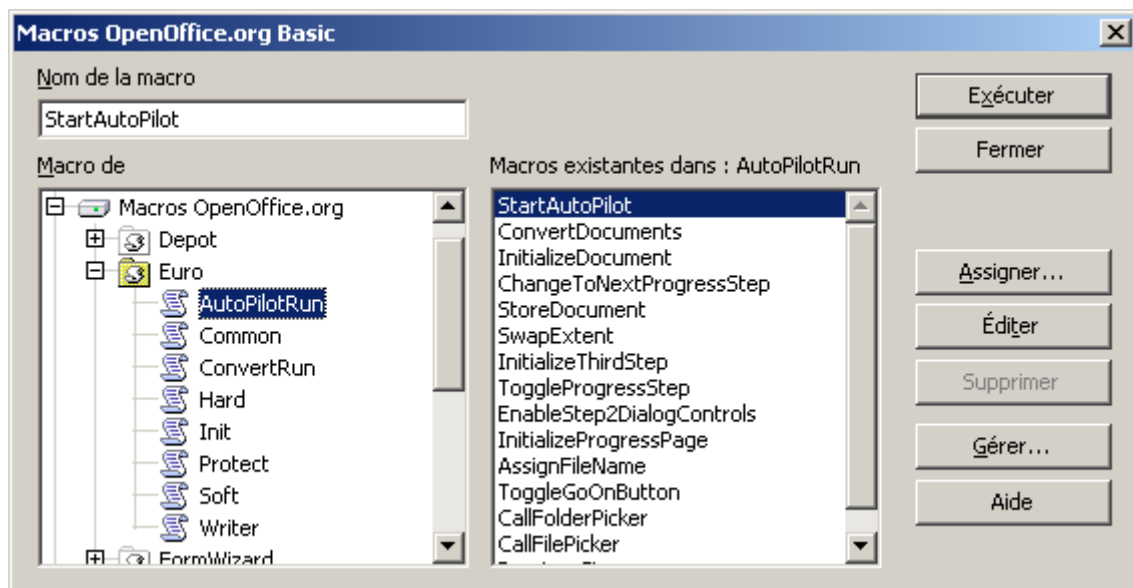
L'illustration ci-dessous vous montre le lancement de la boîte de dialogue de l'assistant Euro-convertisseur.



Cet assistant correspond à un ensemble de macros que vous pouvez retrouver dans la liste des macros internes à OoO. Lancez pour cela, le gestionnaire de macros par le menu **Outils / Macros / Gérer les macros / OpenOffice.org Basic** :



Dans la boîte de dialogue ci-dessous, vous voyez par exemple l'assistant Euro-convertisseur. Celui-ci se trouve dans le container « Macros OpenOffice.org », dans le dossier « Euro ». Il se compose de différentes bibliothèques et chacune d'entre elles contient des macros.



Si vous sélectionnez dans la bibliothèque « AutoPilotRun », la macro « StartAutoPilot » et que vous cliquez sur le bouton « Exécuter », vous obtenez la même boîte de dialogue que précédemment.

Je reviendrai plus tard sur cette organisation entre container, bibliothèque et macro.

On peut classer les macros en trois grandes catégories :

1. **Les macros d'automatisation.** Elles permettent de faciliter l'accès à des commandes très fréquemment utilisées. On les exécute souvent par une combinaison de touches. Par exemple, le changement de casse d'un mot (basculement majuscule / minuscule).
2. **Les macros d'extension.** Elles augmentent les capacités fonctionnelles de OOo. Il s'agit de nos fameux assistants comme par exemple le convertisseur de documents.
3. **Les macros applicatives.** Ce sont des programmes complets destinés à réaliser des tâches complexes. Par exemple, le logiciel [FactOOor](#) permet d'assurer et d'automatiser sa facturation en s'appuyant sur le tableur.

Bien sûr, ce classement n'est pas étanche. On peut très bien trouver des macros d'extension qui réalisent des tâches automatiques et des macros d'automatisation qui ajoutent de nouvelles fonctions à OOo.

## Ma première macro

Nous allons maintenant passer à l'acte en créant notre première macro. Il existe deux manières de créer une macro : l'écrire soi-même en tapant le code ou passer par l'enregistreur.

1. Lorsque vous créez une macro en écrivant son code, vous devez taper manuellement toutes les instructions requises en langage Basic afin que OOo puisse l'exécuter. Si vous débutez, inutile de vous dire qu'il faut être agueri et connaître parfaitement la syntaxe pour utiliser cette méthode.
2. Enregistrer une macro est donc la façon la plus simple car elle ne nécessite aucune connaissance en programmation. Il vous suffit de déclencher la procédure et d'exécuter votre série d'action (saisie, appel de touche, etc). A la fin, l'enregistreur vous restitue l'ensemble de ces actions sous la forme d'une macro.

### Limitation de l'enregistreur

Je vous rappelle que vous ne pouvez pas enregistrer les mouvements de la souris. Par contre, vous avez la possibilité d'enregistrer certain résultat comme une sélection à la souris.

Cependant, si vous aviez l'habitude de vous servir d'une célèbre suite bureautique concurrente, vous risquez d'être très fortement déçu par l'enregistreur de macro de OOo. Celui-ci souffre en effet de sévères lacunes :

1. Il n'est disponible que dans Writer et Calc. Les habitués des bases de données risquent donc de souffrir avec le module Base.
2. Il ne sait pas tout enregistrer. Certaines actions passeront inaperçues notamment celles faisant appel aux boîtes de dialogue et à leur contenu.
3. Le code produit est imbuvable. Chaque action fait appel à un dispatcheur qui ne reflète pas la véritable correspondance avec le langage Basic.

Malgré tout, cette enregistreur vous permettra de démarrer et de comprendre quelques trucs afin d'organiser vos premières macros. Une fois rompu à cela, vous pourrez envisager de passer à une programmation en direct.



#### Note :

Le Basic n'est pas le seul langage possible. OOo permet également l'utilisation du Python, du Javascript et du BeanShell. Je ne traite bien sûr pas de ces derniers dans ce support.

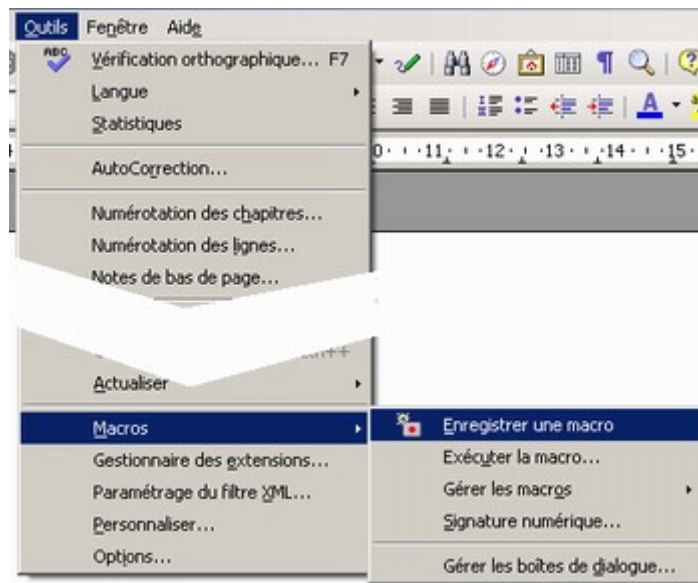
## **Enregistrement de la macro**

Pour notre mise en jambe, nous allons tenter de créer une macro d'automatisation pour le traitement de textes Writer. Celle-ci va permettre de remettre dans l'ordre deux lettres injustement inversées.

Ouvrez un document vierge et saisissez volontairement le mot **bonjoru**.

Le curseur du texte doit clignoter juste derrière la lettre **u**.

Maintenant, sélectionnez le menu **Outils / Macros / Enregistrer une macro**

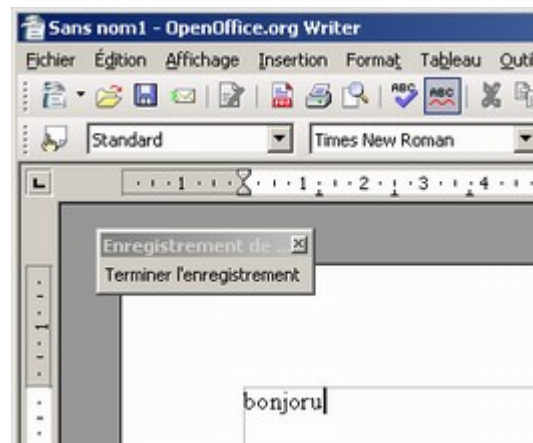


Une petite fenêtre apparaît comme dans l'illustration de droite. A partir de ce point, tout ce que vous faites sera enregistré.

Maintenez la touche Maj. Enfoncée et appuyez sur la touche flèche pour sélectionner le **u**.

Appuyez sur « Ctrl + X » pour couper le caractère.

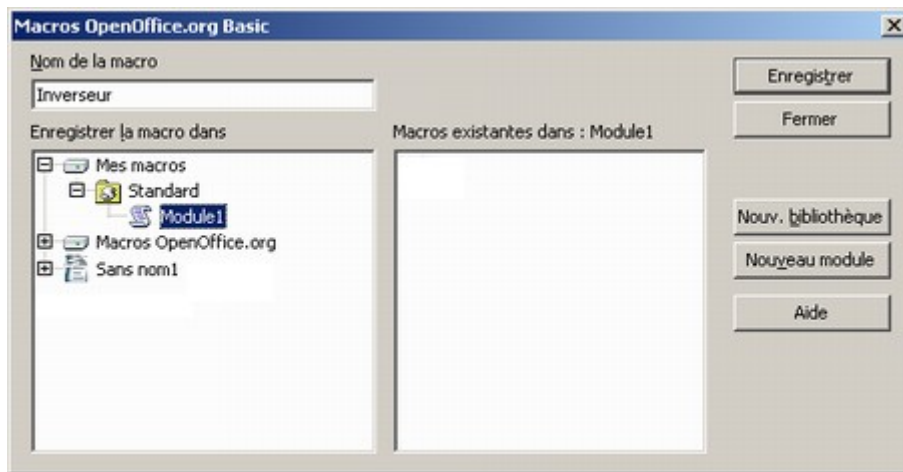
Déplacer le curseur derrière la lettre **r** et appuyez sur « Ctrl + V » pour coller la lettre **u**.



La macro est maintenant finie, nous sommes arrivé au résultat voulu. Vous pouvez cliquer sur le bouton « Terminer l'enregistrement ».



Une boîte apparaît pour achever l'opération.



Dans le champ « Nom de la macro », saisissez « Inverseur » puis sélectionnez le container « Mes macros », répertoire Standard et bibliothèque « Module1 ». Cliquez sur le bouton « Enregistrer ».

Vous venez de créer votre première macro.

### **Stockage des macros**

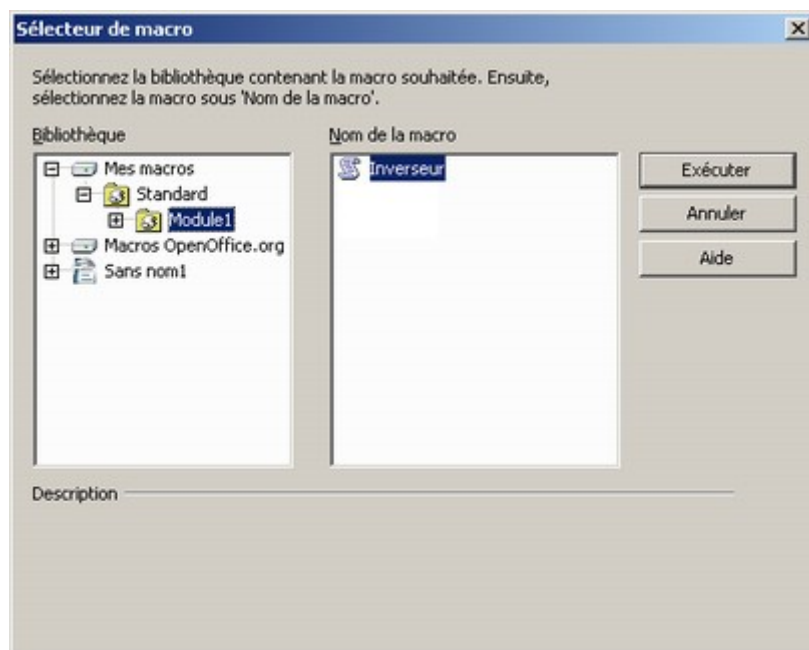
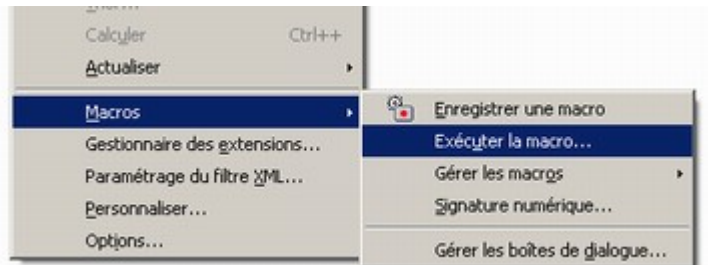
Comme je vous l'avais précisé dans le chapitre précédent, il existe trois emplacements où vont pouvoir être stockées les macros :

1. **Mes macros** Il s'agit du container propre à vos macros et qui seront disponible pour l'ensemble de vos documents.
2. **Macros OpenOffice.org** Il s'agit du container dédié aux macros fournis avec OOo. Vous ne pouvez pas enregistrer de macro dans ce container.
3. Enfin, il existe un container pour le document chargé. La macro ne sera bien sûr disponible que si celui-ci est actif.

## Exécution de la macro

Retapez à nouveau le mot **bonjour** en faisant volontairement une inversion de lettres et laissez le curseur à la bonne place.

Sélectionnez le menu **Outils / Macros / Exécuter la macro**

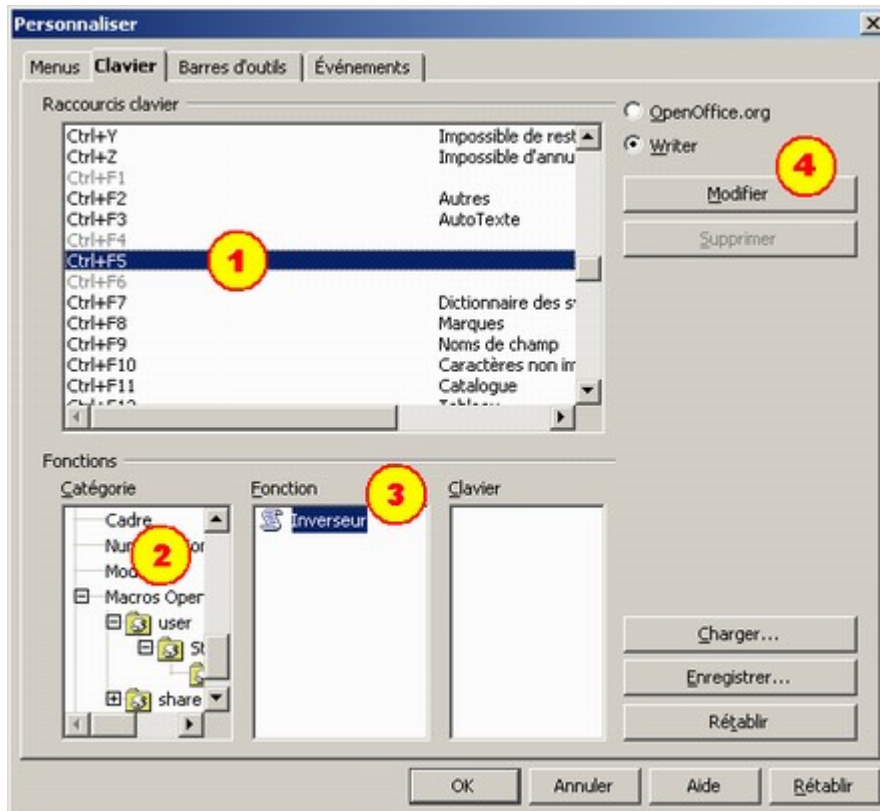


Sélectionnez votre macro et cliquez sur le bouton « Exécuter ». Le mot apparaît correctement orthographié.

## Assignment d'un raccourci clavier

Bien sûr la méthode d'exécution est peu pratique. Il serait donc plus efficace d'avoir un raccourci clavier qui permettrait de réaliser l'opération rapidement.

Sélectionnez le menu **Outils / Personnaliser** et choisissez l'onglet **Clavier**.



1. Choisissez le raccourci clavier à affecter, par exemple « Ctrl + F5 » qui est disponible.
2. Sélectionnez la catégorie Macros OpenOffice.org / user / Standard / Module1.
3. Sélectionnez votre macro.
4. Laissez l'option Writer afin de ne rendre ce raccourci disponible uniquement dans le traitement de texte. Cliquez sur le bouton « Modifier ».

Le raccourci est maintenant affecté. Cliquez sur « OK » pour quitter le dialogue. Recommencez l'opération d'exécution de la macro en utilisant cette fois le raccourci clavier « Ctrl + F5 », vous constatez que cela fonctionne.

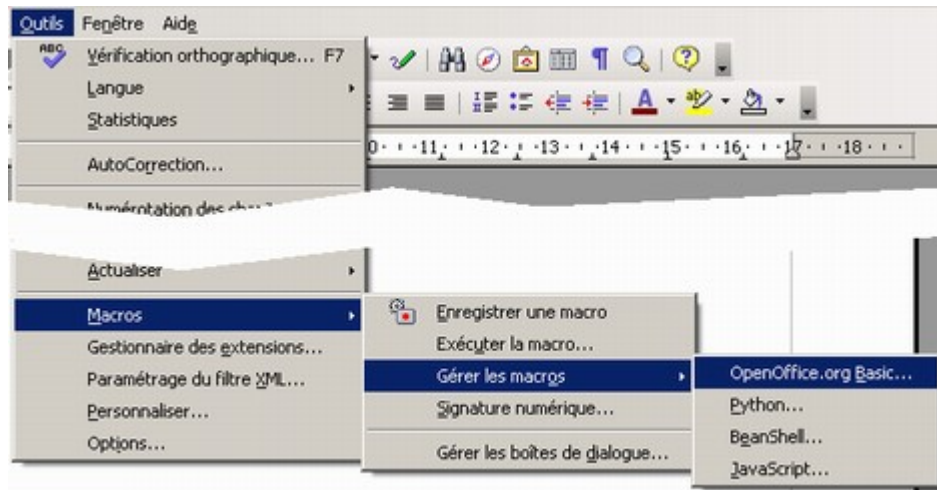


### Note :

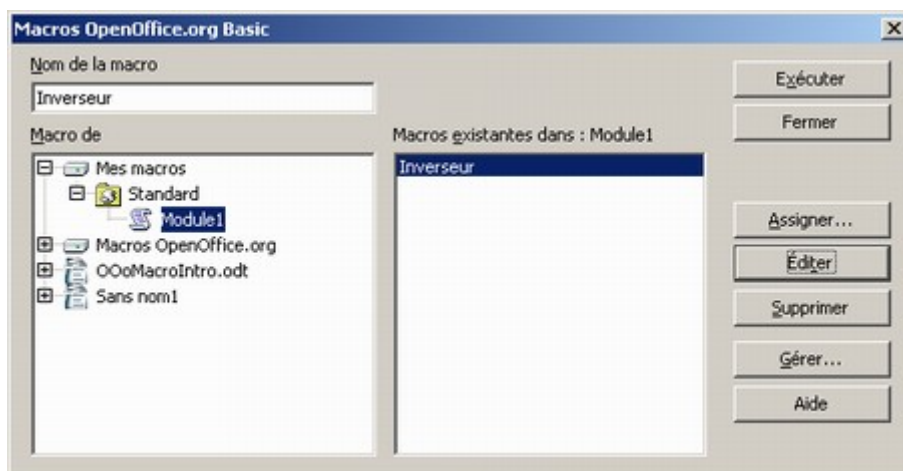
Certains raccourcis ne peuvent être affectés et sont grisés parce qu'ils sont réservés au système d'exploitation.

## ***Édition de la macro***

Voyons ensemble ce que donne notre macro. Sélectionnez le menu **Outils / Macros / Gérer les macros / OpenOffice.org Basic** :

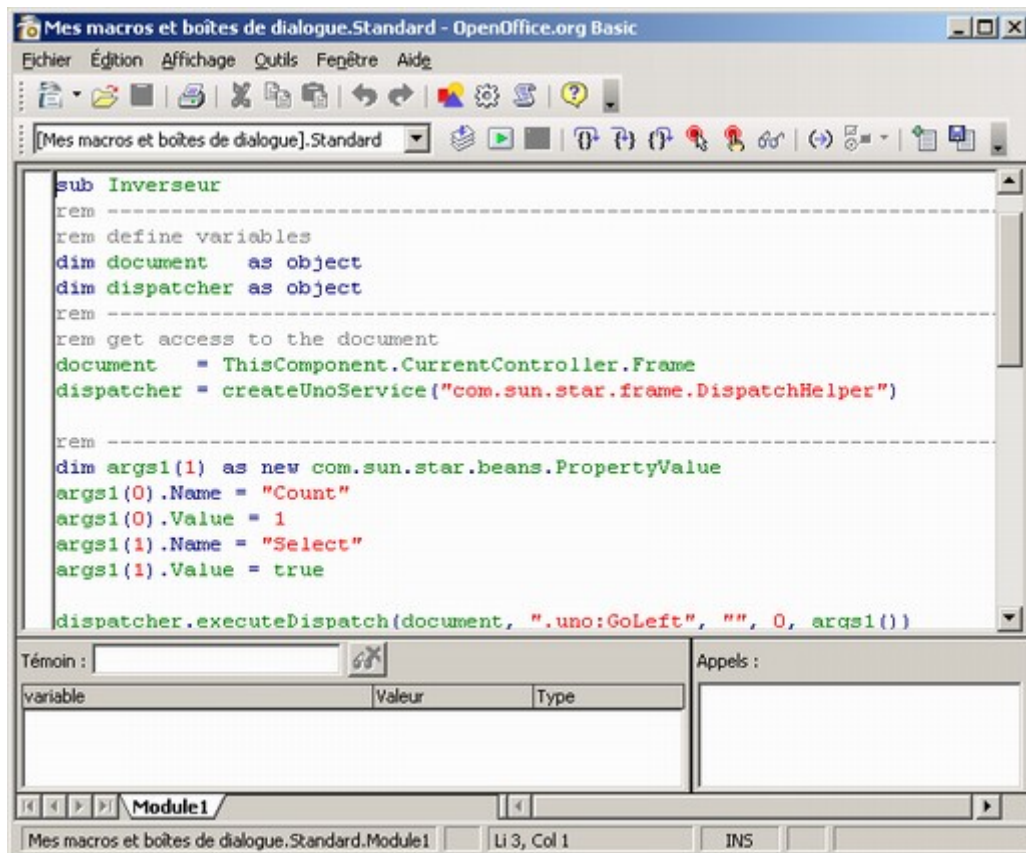


Sélectionnez votre macro comme nous l'avons vu précédemment.



Cliquez sur le bouton « Éditer ».

L'éditeur Basic s'ouvre avec le contenu de votre macro.



Examinons sommairement le code généré.

```
sub Inverseur
```

Terme énigmatique qui apparaît au début de chaque macro afin d'indiquer que celle-ci soit considéré comme une sous-routine. Il indique donc le début de la macro suivi de son nom et qui sera exécuté jusqu'au terme de fin de routine « `end sub` ».

Nous avons également des lignes grisées qui commencent par « `rem` ». Ce sont des lignes de remarque qui servent à commenter votre code.

```
dim document as object
dim dispatcher as object
```

Le terme « `dim` » permet la déclaration des variables permettant de réserver des ressources en mémoire pour le stockage des informations. Le terme « `as object` » permet de spécifier le type de variable. Il existe d'autres types comme boolean (booléen), long (entier long), date, etc.

```
document = ThisComponent.CurrentController.Frame
```

On affecte l'intégralité du document courant dans la variable.

```
dispatcher = createUnoService("com.sun.star.frame.DispatchHelper")
```

On appelle le service du dispatcheur qui va permettre de manipuler les commandes.

```
dim args1(1) as new com.sun.star.beans.PropertyValue  
args1(0).Name = "Count"  
args1(0).Value = 1  
args1(1).Name = "Select"  
args1(1).Value = true
```

On déclare une variable qui va dériver de ce que l'on appelle un modèle de donnée. Celle-ci est définie dans un tableau composé de deux variables dont l'index démarre à zéro. Chaque variable possède une propriété Name (nom) et Value (valeur).

```
dispatcher.executeDispatch(document, ".uno:GoLeft", "", 0, args1())
```

On exécute le service du dispatcheur sur le document en lui passant les paramètres qui vont bien : ici, aller vers la gauche en sélectionnant et en comptant un caractère.

Le reste du code reprend le même principe, je ne détaille donc pas plus.

## Ma seconde macro

Nous allons maintenant réaliser une macro pour Calc qui permet de trier une colonne rapidement.



Vous allez me dire qu'il existe déjà deux boutons pour faire du tri rapide dans la barre d'outils.

Le problème, c'est que ces boutons trient toute une colonne. Par exemple, si je veux trier la colonne Ville...

	A	B	C
1	Nom	Prénom	Ville
2	Depardieu	Gérard	Paris
3	Rocheftort	Jean	Lyon
4	Perez	Vincent	Marseille
5	Bouchez	Elodie	Paris

	A	B	C
1	Bouchez	Elodie	Paris
2	Depardieu	Gérard	Paris
3	Nom	Prénom	Ville
4	Perez	Vincent	Marseille
5	Rocheftort	Jean	Lyon

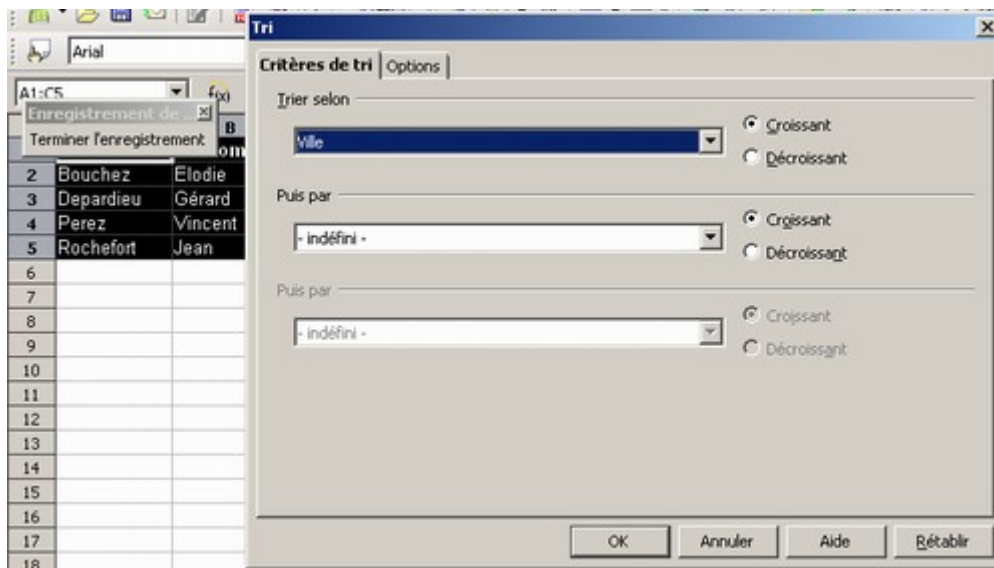
Le résultat ne sera pas cohérent car l'entête de la colonne est pris en compte.

## Enregistrement de la macro

Nous allons donc créer une macro d'automatisation pour réaliser un tri rapide en tenant compte de l'entête de colonne dans le tableur Calc.

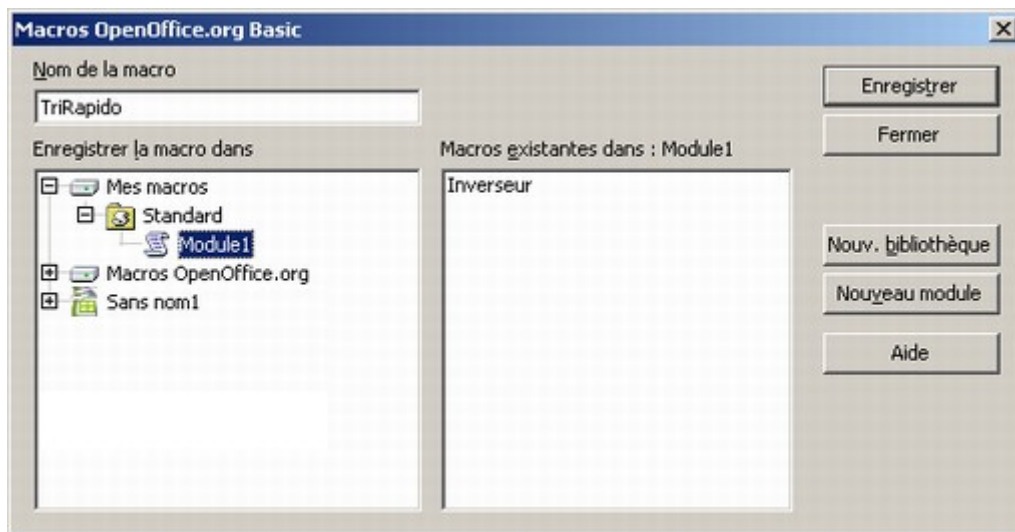
Ouvrez un classeur vierge et saisissez données comme dans l'exemple ci-dessus.

Lancez l'enregistreur et choisissez le menu **Données / Trier**. Dans le dialogue, sélectionnez la colonne « Ville » et validez.



Formation express / Introduction aux macros et au Basic

Terminez l'enregistrement et sauvegardez votre macro au même endroit que la première mais en lui donnant bien sûr un autre nom.

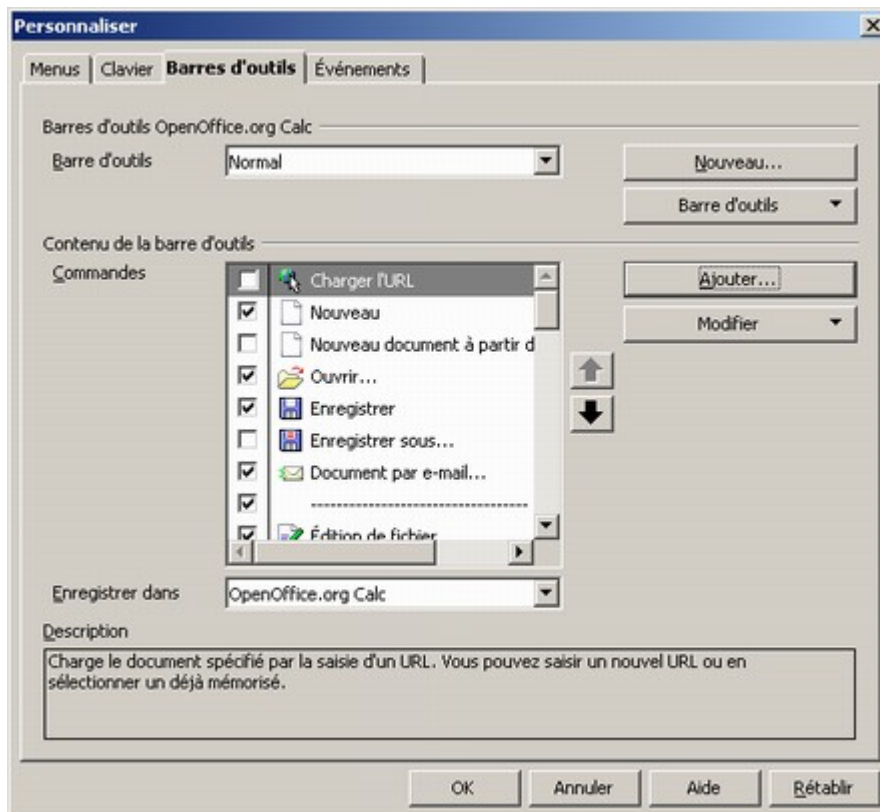


Dans notre exemple, nous appellerons notre macro « TriRapido ».

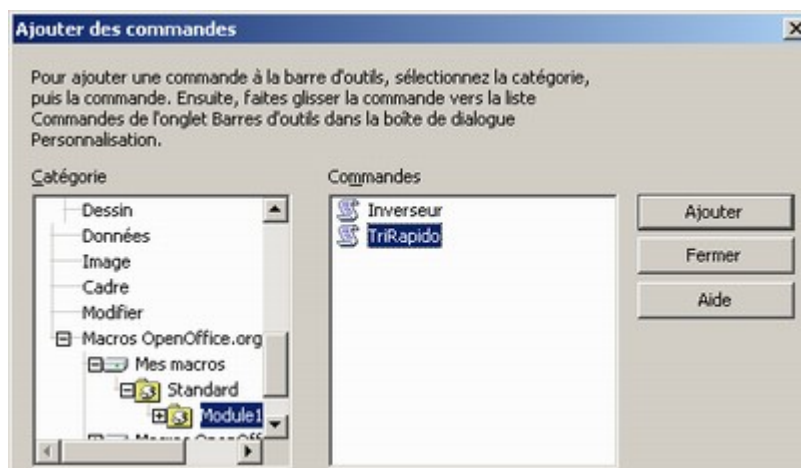


## Assignment d'un bouton en barre d'outils

Cette fois, nous allons créer un bouton en barre d'outils pour appeler rapidement cette macro. Sélectionnez le menu **Outils / Personnaliser** et placez-vous sur l'onglet « Barre d'outils ».

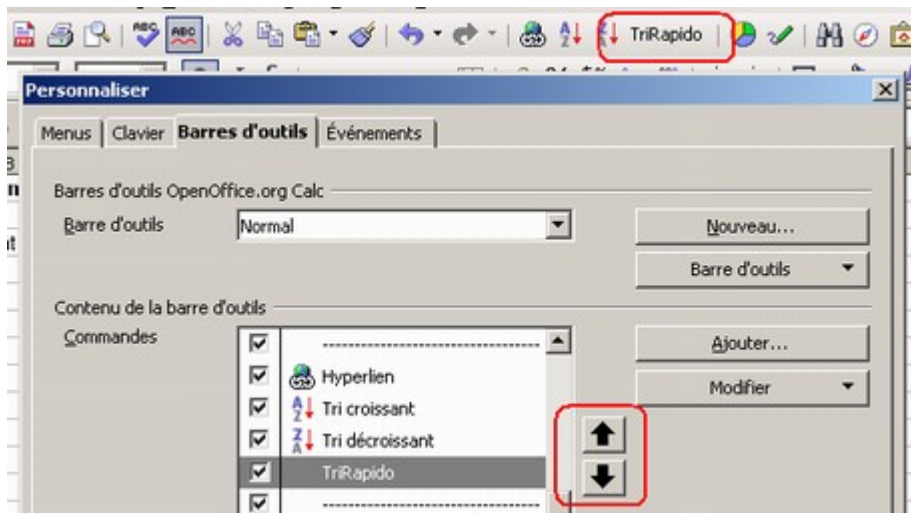


Cliquez sur le bouton « Ajouter ».

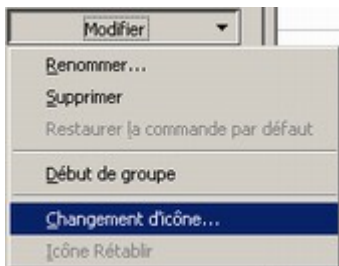


Sélectionnez depuis la zone « Catégorie », votre bibliothèque et votre macro puis cliquez sur « Ajouter ».

## Formation express / Introduction aux macros et au Basic



A l'aide des boutons en forme de flèches, placez votre bouton à l'endroit désiré comme dans l'illustration ci-dessus.



Cliquez sur le bouton « Modifier » et choisissez le menu « Changement d'icône ».

Choisissez dans la liste, un dessin symbolisant au mieux votre macro.



Validez les deux dialogues par le bouton « OK ».

## Personnaliser la macro

Le problème de cette macro, c'est qu'elle ne fait que ce que l'on a enregistré : elle ne trie donc que la colonne Ville. Or ce qui serait mieux, c'est qu'elle trie la colonne sur laquelle le curseur se trouve.

Voyons ensemble ce que contient notre macro. Sélectionnez le menu **Outils / Macros / Gérer les macros / OpenOffice.org Basic** et éditez « TriRapido ».

L'éditeur Basic s'ouvre. Pour des raisons de lisibilité, j'ai enlevé toutes les lignes de remarque.

```
sub TriRapido

dim document as object
dim dispatcher as object
document = ThisComponent.CurrentController.Frame
dispatcher = createUnoService("com.sun.star.frame.DispatchHelper")

dim args1(6) as new com.sun.star.beans.PropertyValue
args1(0).Name = "ByRows"
args1(0).Value = true
args1(1).Name = "HasHeader"
args1(1).Value = true
args1(2).Name = "CaseSensitive"
args1(2).Value = false
args1(3).Name = "IncludeAttribs"
args1(3).Value = true
args1(4).Name = "UserDefIndex"
args1(4).Value = 0
args1(5).Name = "Coll"
args1(5).Value = 3
args1(6).Name = "Ascending1"
args1(6).Value = true

dispatcher.executeDispatch(document, ".uno:DataSort", "", 0, args1())

end sub
```

Nous allons détailler la partie qui nous intéresse. C'est à dire le tableau d'arguments servant à lancer l'action « `uno:DataSort` » du dispatcheur.

```
args1(0).Name = "ByRows"
args1(0).Value = true
```

Direction du tri par ligne (de haut en bas).

```
args1(1).Name = "HasHeader"  
args1(1).Value = true
```

La page de tri contient des étiquettes de colonne.

```
args1(2).Name = "CaseSensitive"  
args1(2).Value = false
```

Respect de la casse.

```
args1(3).Name = "IncludeAttribs"  
args1(3).Value = true
```

Conservation du format actif des cellules.

```
args1(4).Name = "UserDefIndex"  
args1(4).Value = 0
```

Précise si une liste de tri doit être utilisée.

```
args1(5).Name = "Col1"  
args1(5).Value = 3
```

Numéro de la colonne à trier. C'est donc sur ce paramètre qu'il faut agir.

```
args1(6).Name = "Ascending1"  
args1(6).Value = true
```

Tri dans l'ordre croissant.

Il nous faut donc récupérer la colonne sur laquelle le curseur se trouve. Pour cela, nous allons d'abord déclarer deux variables pour stocker le numéro de la colonne et les valeurs de la cellule courante.

```
Dim numCol as integer
```

La variable est un entier.

```
Dim oCell as object
```

La variable est un objet.

Nous allons maintenant stocker les valeurs de la cellule.

```
oCell = ThisComponent.CurrentSelection.getRangeAddress
```

On prend le fichier dans son ensemble et on extrait la sélection courante dont on stocke les coordonnées de la cellule.

```
numCol = oCell.StartColumn + 1
```

On récupère la colonne de départ de la cellule. Comme l'indexage des colonnes démarre à zéro, on ajoute 1.

Formation express / Introduction aux macros et au Basic

Il ne nous reste plus qu'à modifier la ligne qui comporte l'argumentaire du numéro de colonne.

```
args1(5).Name = "Col1"  
args1(5).Value = numCol
```

C'est tout !

Si vous testez maintenant votre macro TriRapido. Elle va fonctionner quelque soit la colonne utilisée.

## Conclusion

Voilà ! Nous venons de couvrir l'essentiel de ce qu'il faut savoir pour démarrer avec les macros et commencer à créer vos propres automatismes avec **OOo**.

Je vous rappelle que vous pouvez trouver de nombreux exemples de macro dans la section dédiée du [Forum francophone de OpenOffice.org](#).

Bon courage dans vos développements.