



Guide du débutant

## *Chapitre 13*

### *Débuter avec les macros*

*Utiliser l'Enregistreur de Macro... et au-delà*



## Droits d'auteur

---

Ce document est protégé par Copyright © 2017 par l'Équipe de Documentation de LibreOffice. Les contributeurs sont nommés ci-dessous. Vous pouvez le distribuer et/ou le modifier sous les termes de la Licence Publique Générale GNU (<http://www.gnu.org/licenses/gpl.html>), version 3 ou ultérieure, ou de la Licence Creative Commons Attribution (<http://creativecommons.org/licenses/by/4.0/>), version 4.0 ou ultérieure.

Toutes les marques déposées citées dans ce guide appartiennent à leurs légitimes propriétaires.

## Contributeurs

Andrew Pitonyak  
Ron Faile Jr.

Peter Schofield  
Hazel Russman

Martin Fox  
Olivier Hallot

## Traducteur

Christian Chenal

Jean-Luc Vandemeulebroucke

Relecteurs : François OLIVIER

## Retours

Veuillez adresser tout commentaire ou suggestion concernant ce document à la liste de diffusion de l'Équipe de Documentation : [doc@fr.libreoffice.org](mailto:doc@fr.libreoffice.org)

Remarque : tout ce que vous envoyez à la liste de diffusion, y compris votre adresse mail et toute autre information personnelle incluse dans le message, est archivé publiquement et ne peut pas être effacé.

## Remerciements

Ce document est adapté et mis à jour à partir du Chapitre 13 de *Getting Started with OpenOffice.org 3.3*. Ont contribué à son édition :

Andrew Pitonyak

Jean Hollis Weber

## Traduction

Cette traduction s'appuie sur le *Guide du Débutant – Chapitre 13, Débuter avec les macros*, version 4.0 de LibreOffice.

Traducteur :

Relecteurs : Pierre-Yves Samyn

## Date de publication et version du logiciel

Publié le 26 avril 2018. Basé sur LibreOffice 5.2.

Le système d'exploitation utilisé pour les copies d'écran est Windows 10 et le jeu d'icônes est Galaxy.

## Utiliser LibreOffice sur un Mac

---

Sur Mac, certaines touches et certains éléments de menu sont différents de ceux utilisés sous Windows ou Linux. Le tableau ci-dessous donne quelques substitutions courantes pour les instructions de ce chapitre. Pour une liste plus détaillée, voyez l'Aide de l'application.

<b>Windows ou Linux</b>	<b>Équivalent Mac</b>	<b>Effet</b>
Sélection du menu <b>Outils &gt; Options</b>	<b>LibreOffice &gt; Préférences</b>	Accès aux options de configuration
Clic droit	<i>Control+clic</i> ou <i>clic droit</i> selon la configuration de l'ordinateur	Ouvre un menu contextuel
<i>Ctrl</i> ( <i>Control</i> )	⌘ ( <i>Command</i> )	Utilisé avec d'autres touches
<i>F5</i>	<i>Shift+⌘+F5</i>	Ouvre le Navigateur
<i>F11</i>	⌘+T	Ouvre la fenêtre Styles et Formatage

## Table des matières

---

<b>Droits d'auteur</b> .....	<b>3</b>
Contributeurs.....	3
Traducteur.....	3
Retours.....	3
Remerciements.....	3
Traduction.....	3
Date de publication et version du logiciel.....	3
<b>Utiliser LibreOffice sur un Mac</b> .....	<b>4</b>
<b>Introduction</b> .....	<b>7</b>
<b>Vos premières macros</b> .....	<b>7</b>
Ajouter une macro.....	7
Enregistrer une macro.....	10
Exécuter une macro.....	10
Afficher et éditer des macros.....	11
Commentaires REM.....	12
Définir des sous-routines avec SUB.....	12
Définir des variables avec DIM.....	12
Explication du code de la macro.....	12
<b>Une macro plus complexe</b> .....	<b>13</b>
Exemple.....	14
Enregistrement de la macro.....	14
Exécuter rapidement une macro.....	17
<b>Limites de l'enregistrement de macro</b> .....	<b>18</b>
Schéma de transmission des commandes.....	18
Comment l'enregistreur de macro utilise le schéma de transmission des commandes.....	18
Autres options.....	18
<b>Organisation des macros</b> .....	<b>19</b>
Où les macros sont-elles enregistrées ?.....	20
Importer des macros.....	20
<b>Comment exécuter une macro ?</b> .....	<b>22</b>
Barres d'outils, éléments de menus et raccourcis clavier.....	22
Événements.....	23
<b>Écrire des macros sans enregistrement</b> .....	<b>25</b>



## Introduction

---

Une macro est une séquence enregistrée de commandes ou de saisies au clavier qui sont conservées pour une utilisation ultérieure. Un exemple de macro simple serait une macro qui “saisit” votre adresse. Le langage de macro LibreOffice est très souple et permet l'automatisation de tâches simples ou complexes. Les macros sont particulièrement utiles pour répéter une suite d'actions exactement de la même façon, plusieurs fois au cours du temps.

LibreOffice prend en charge les langages suivants :

- LibreOffice Basic
- JavaScript
- BeanShell
- Python

Le langage LibreOffice Basic est le plus couramment utilisé pour écrire des macros dans LibreOffice. C'est ce langage qui est utilisé pour les exemples figurant dans ce chapitre.

Vous pouvez apprendre le langage LibreOffice Basic afin de pouvoir écrire des macros de A à Z, mais cela nécessite de posséder ou d'acquérir de bonnes compétences en programmation informatique. Si vous êtes débutant, il est plus facile d'utiliser des macros écrites par d'autres, ou d'utiliser l'enregistrement de macros, qui permet de mémoriser les touches frappées et d'enregistrer cette séquence pour une utilisation ultérieure.

La plupart des tâches dans LibreOffice sont accomplies en “dispatchant” (envoyant) une commande, qui est interceptée et utilisée. L'enregistrement de macros fonctionne en enregistrant les commandes qui sont dispatchées (voir « **Schéma de transmission des commandes** » page 18).

## Vos premières macros

---

### Ajouter une macro

La première étape dans l'apprentissage de la programmation des macros est de savoir retrouver et utiliser des macros existantes. Dans la suite de cette section, vous devez disposer d'une macro que vous voulez utiliser. Pour cet exemple, utilisez la macro simple du Listing 1, qui permet d'afficher le message « Bonjour ! ». Vous devez créer une bibliothèque et un module pour contenir votre macro. Voyez « **Organisation des macros** » page 19 pour plus d'informations.

*Listing 1. Macro simple qui affiche bonjour*

```
Sub MacroBonjour
  MsgBox "Bonjour !"
End Sub
```

Suivez les étapes suivantes pour créer et exécuter la macro du Listing 1 :

- 1) Choisissez **Outils > Macros > Gérer les macros > LibreOffice Basic** pour ouvrir la boîte de dialogue *Macros LibreOffice Basic* (Figure 1).
- 2) Cliquez sur **Gérer** pour ouvrir la boîte de dialogue *Gestionnaire de macros LibreOffice Basic* et sélectionnez l'onglet *Bibliothèques* (Figure 2).
- 3) Si ce n'est déjà fait, choisissez l'emplacement *Mes macros et boîtes de dialogue*, qui est l'emplacement par défaut.
- 4) Cliquez sur **Nouveau** pour ouvrir la boîte de dialogue *Nouvelle bibliothèque*.
- 5) Saisissez le nom de la nouvelle bibliothèque, **BibTest** par exemple, et cliquez sur **OK**.
- 6) Sélectionnez l'onglet *Modules*.

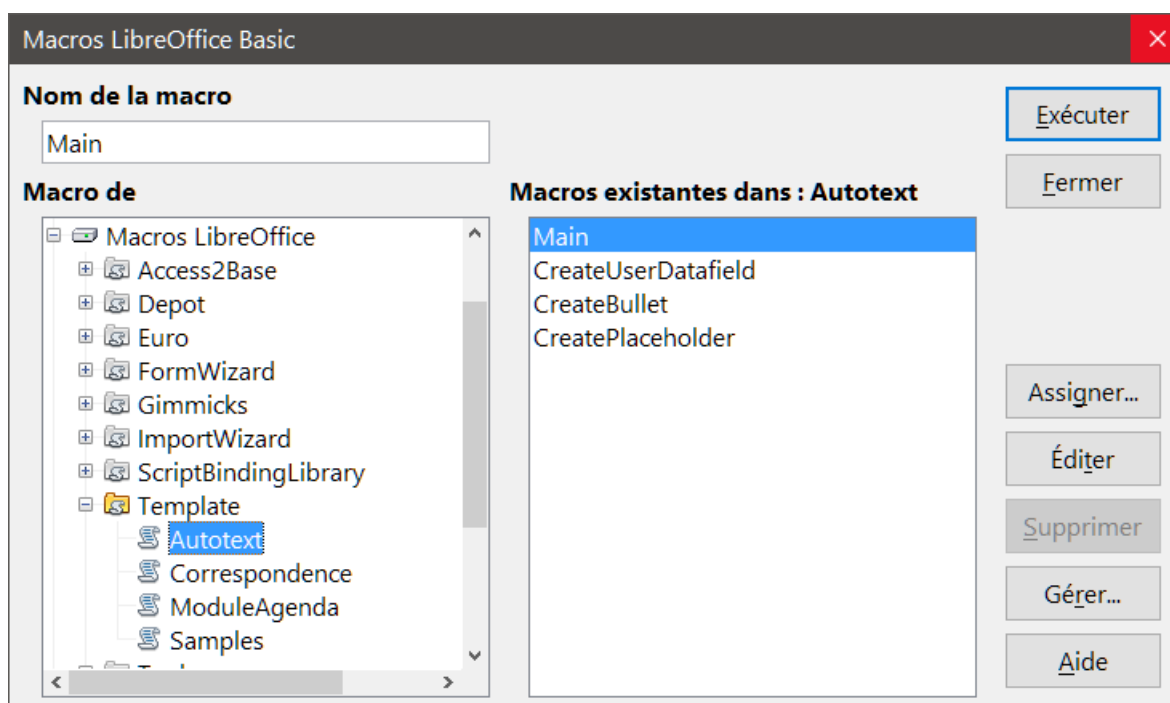


Figure 1 : Boîte de dialogue Macros LibreOffice Basic

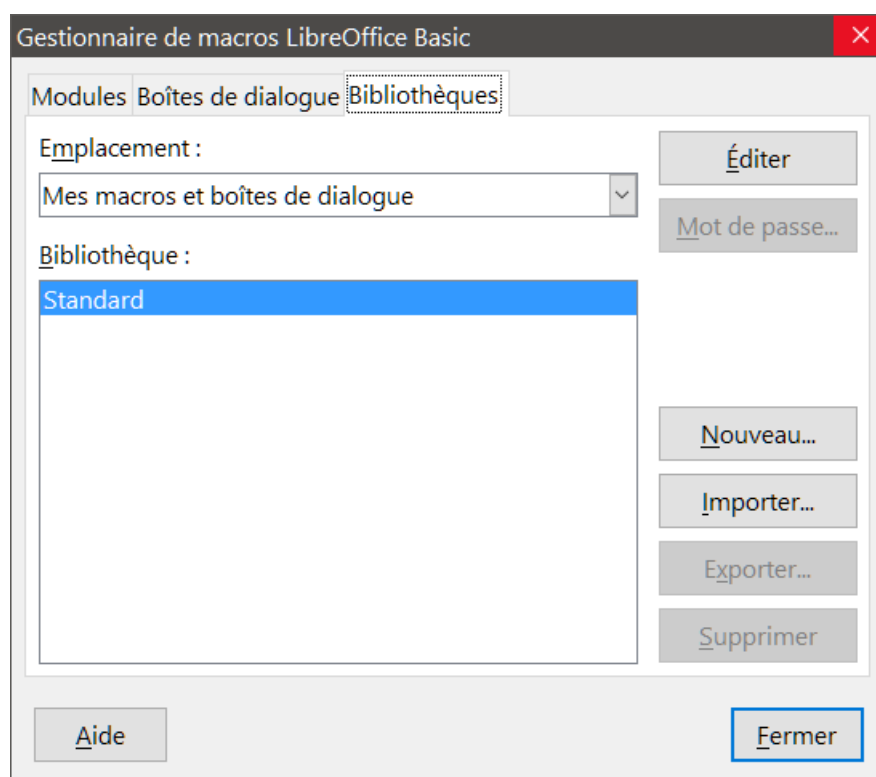


Figure 2 : Boîte de dialogue Gestionnaire de macros LibreOffice Basic

- 7) Dans la liste des modules, étendez *Mes macros* et sélectionnez votre nouvelle bibliothèque. Un module nommé *Module1* existe déjà et peut contenir votre macro. Vous pouvez également cliquer sur **Nouveau** pour créer un nouveau module dans votre bibliothèque.



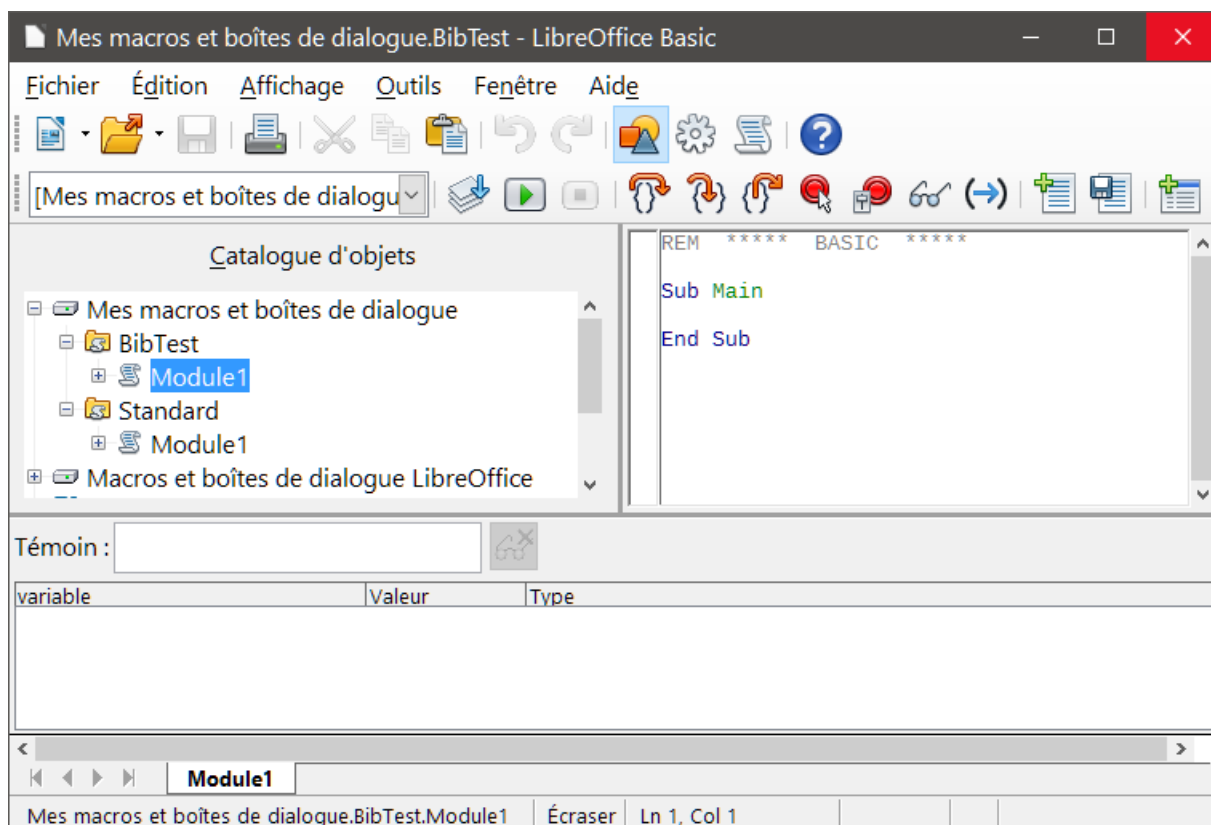


Figure 3 : Environnement de Développement Intégré LibreOffice Basic

- 8) Sélectionnez le module et cliquez sur **Éditer** pour ouvrir l'Environnement de Développement Intégré (Figure 3). Cet EDI est un éditeur de texte inclus dans LibreOffice qui vous permet de créer et de modifier des macros.
- 9) Lorsqu'un nouveau module est créé, il contient un commentaire et une macro vide nommée Main, qui ne fait rien.
- 10) Ajoutez la nouvelle macro avant Sub Main ou après End Sub. Le Listing 2 montre la nouvelle macro insérée après End Sub.

Listing 2. Module1 après l'ajout de la macro

```



REM ***** BASIC *****

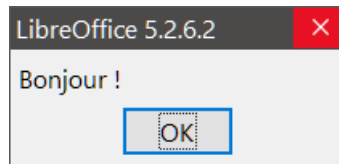
Sub Main


End Sub

Sub MacroBonjour
  MsgBox "Bonjour !"
End Sub

```

- 11) Cliquez sur l'icône **Compiler**  de la barre d'outils Macro pour compiler la macro.
- 12) Placez le curseur à l'intérieur de la sous-routine MacroBonjour et cliquez sur l'icône **Exécuter le BASIC**  de la barre d'outils Macro, ou appuyez sur la touche F5, pour exécuter la macro MacroBonjour du module. Une petite boîte de dialogue s'ouvre en affichant le mot Bonjour.



- 13) Cliquez sur **OK** pour refermer cette boîte de dialogue.
- 14) Si le curseur ne se trouve pas à l'intérieur d'une sous-routine ou d'une fonction, la boîte de dialogue *Macros LibreOffice Basic* s'ouvre et vous demande de sélectionner la macro à exécuter. L'icône **Sélectionner une macro**  de la barre d'outils *Standard* de l'EDI et la commande **Outils > Macros > Gérer les macros > LibreOffice Basic** ont le même effet.
- 15) Sélectionnez une macro puis cliquez sur le bouton **Exécuter le BASIC**.

## Enregistrer une macro

Si vous devez effectuer des opérations de manière répétée, comme la saisie d'informations fixes par exemple, vous pouvez créer une macro qui mémorise ces opérations. La façon la plus simple de le faire est d'utiliser l'enregistrement de macro.

- 1) Vérifiez que l'enregistrement de macro est bien actif en allant dans **Outils > Options > LibreOffice > Avancé** et en cochant l'option **Activer l'enregistreur de macros**. Par défaut, cette fonction est désactivée lors de l'installation de LibreOffice sur votre ordinateur.
- 2) Choisissez **Outils > Macros > Enregistrer une macro** dans la barre de menus principale pour démarrer l'enregistrement de la macro. Une petite boîte de dialogue *Enregistrer une macro* apparaît, indiquant que l'enregistrement de macro est en cours.
- 3) Effectuez la séquence d'opérations que vous désirez. Par exemple, saisissez votre nom.
- 4) Cliquez sur **Terminer l'enregistrement** dans la boîte de dialogue *Enregistrer une macro* pour arrêter l'enregistrement. La boîte de dialogue *Macros LibreOffice Basic* s'ouvre (Figure 1).
- 5) Étendez le conteneur de bibliothèques Mes macros et trouvez la bibliothèque Standard. Notez que chaque conteneur de bibliothèques contient une bibliothèque nommée Standard.
- 6) Sélectionnez la bibliothèque Standard et cliquez sur **Nouveau Module** pour créer le nouveau module qui va contenir la macro. La boîte de dialogue *Nouveau module* s'ouvre.
- 7) Saisissez un nom significatif pour le nouveau module, **Enregistrements** par exemple, et cliquez sur **OK** pour le créer. Le nouveau module figure désormais dans la bibliothèque Standard dans la boîte de dialogue *Macros LibreOffice Basic*.
- 8) Dans la zone **Nom de la macro**, saisissez le nom de la macro que vous venez d'enregistrer, **SaisieNom** par exemple.
- 9) Cliquez sur **Enregistrer** pour enregistrer la macro et fermer la boîte de dialogue *Macros LibreOffice Basic*.

Si vous avez suivi toutes les étapes précédentes, la bibliothèque Standard contient maintenant un module appelé *Enregistrements* et celui-ci possède une macro nommée *SaisieNom*.



### Remarque

Quand LibreOffice crée un nouveau module, il y ajoute automatiquement la macro appelée *Main*.

---

## Exécuter une macro

- 1) Choisissez **Outils > Macros > Exécuter la macro** dans la barre de menus principale pour ouvrir la boîte de dialogue *Sélecteur de macro* (Figure 4).

- 2) Déroulez l'arborescence des bibliothèques, sélectionnez la macro désirée, SaisieNom par exemple, et cliquez sur **Exécuter**.

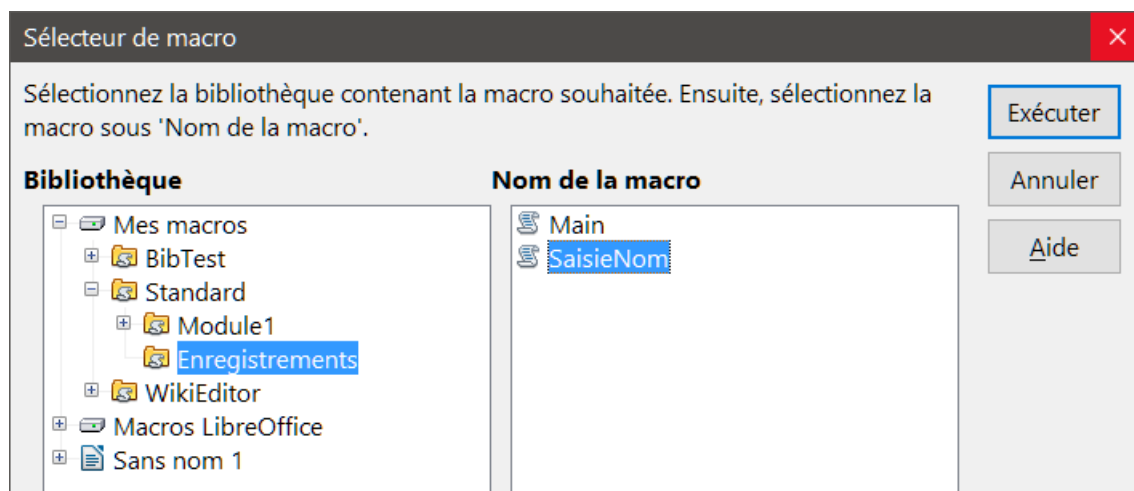


Figure 4 : Boîte de dialogue Sélecteur de macro

Vous pouvez également choisir **Outils > Macros > Gérer les macros > LibreOffice Basic** dans la barre de menus principale pour ouvrir la boîte de dialogue *Macros LibreOffice Basic* (Figure 1), sélectionner la macro et cliquer sur **Exécuter**.

## Afficher et éditer des macros

Pour afficher ou éditer les macros que vous avez créées :

- 1) Choisissez **Outils > Macros > Gérer les macros > LibreOffice Basic** pour ouvrir la boîte de dialogue *Macros LibreOffice Basic*.
- 2) Déroulez l'arborescence des bibliothèques, sélectionnez la macro désirée, SaisieNom par exemple, et cliquez sur **Éditer** pour ouvrir l'EDI Basic. La macro SaisieNom est affichée Listing 3.

Listing 3. Macro enregistrée SaisieNom

```
REM ***** BASIC *****
Sub Main
End Sub

sub SaisieNom
rem -----
rem define variables
dim document as object
dim dispatcher as object
rem -----
rem get access to the document
document = ThisComponent.CurrentController.Frame
dispatcher = createUnoService("com.sun.star.frame.DispatchHelper")
rem -----
dim args1(0) as new com.sun.star.beans.PropertyValue
args1(0).Name = "Text"
args1(0).Value = "Jean Dupont"
```

```
dispatcher.executeDispatch(document, ".uno :InsertText", "", 0,  
args1())  
end sub
```

La macro SaisieNom n'est pas aussi compliquée qu'elle apparaît au premier abord. La connaissance de quelques notions permet de comprendre l'essentiel des macros.

### Commentaires REM

Toutes les séquences du code de la macro qui commencent par REM contiennent des commentaires. Tout le texte situé après REM et se trouvant sur la même ligne est ignoré. Vous pouvez employer également le caractère apostrophe (') comme raccourci pour débiter un commentaire.

Les mots-clés de LibreOffice Basic ne sont pas sensibles à la casse : REM, Rem et rem peuvent donc être tous utilisés pour débiter un commentaire.

### Définir des sous-routines avec SUB

Chacune des macros est stockée dans une sous-routine. Une sous-routine commence avec le mot-clé SUB, et se termine avec les mots END SUB. Le code commence par la définition de la sous-routine appelée Main, qui est vide et ne fait rien. La sous-routine suivante, SaisieNom, contient le code généré pour votre macro.

Les points suivants, qui font partie des fonctionnalités avancées et ne seront pas développés dans ce guide, sont cités pour votre information :

- Vous pouvez écrire une macro de sorte que des valeurs soient transmises à la sous-routine. Ces valeurs sont appelées arguments. Cependant, les macros enregistrées n'admettent pas d'arguments.
- Il existe une autre forme de sous-routine, appelée fonction, qui renvoie une valeur. Les fonctions sont définies par le fait qu'elles commencent par le mot-clé FUNCTION. Cependant, l'enregistrement de macro crée toujours des sous-routines qui ne sont pas des fonctions.

### Définir des variables avec DIM

Vous pouvez écrire une information sur un bout de papier pour vous en souvenir pour plus tard. Une variable, comme un bout de papier, contient une information qui peut être modifiée et lue. Le mot-clé DIM est à l'origine une abréviation de Dimension et était utilisé pour définir les dimensions d'un tableau. L'instruction DIM utilisée dans la macro SaisieNom revient à mettre de côté un bout de papier destiné à recueillir un message ou une note.

Dans la macro SaisieNom, les variables *document* et *dispatcher* sont définies avec le type *object* (objet). Il existe d'autres types de variables, comme *string* (chaîne de caractères), *integer* (nombre entier) ou *date* (date). Une troisième variable, nommée *args1*, est un tableau de valeurs de propriétés. Une variable de type *array* (tableau) permet à une variable de contenir des valeurs multiples, comme un livre qui contient plusieurs pages. Les valeurs d'un tableau sont habituellement numérotées à partir de zéro. Le nombre entre parenthèses indique le nombre le plus haut utilisable pour accéder à une valeur. Dans cet exemple, il y a uniquement une valeur, et elle porte le numéro zéro.

### Explication du code de la macro

Voici une explication du code de la macro SaisieNom. Vous ne comprendrez peut-être pas tous les détails, mais l'explication de chaque ligne de code vous donnera une idée de la façon dont la macro fonctionne.

```
sub SaisieNom
```

Définit le début de la macro

```
dim document as object
```

Définit la variable

```
dim dispatcher as object
```

Définit la variable

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

ThisComponent fait référence au document en cours.

CurrentController est une propriété qui fait référence au service qui contrôle le document. Par exemple, lorsque vous faites une saisie, c'est CurrentController qui mémorise ce que vous tapez. CurrentController transmet alors les modifications au cadre du document.

Frame est une propriété du document qui renvoie le cadre principal du document.

Par conséquent, document est une variable qui fait référence au cadre du document, qui reçoit les commandes transmises.

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

La plupart des tâches de LibreOffice sont accomplies en transmettant des commandes. LibreOffice comprend un service DispatchHelper qui effectue le plus gros du travail pour transmettre les commandes des macros. La méthode createUnoService comprend le nom d'un service et essaie de créer une instance de ce service. En fin de compte, la variable dispatcher fait référence au DispatchHelper.

```
dim args1(0) as new com.sun.star.beans.PropertyValue
```

Déclare un tableau de propriétés. Chaque propriété comporte un nom et une valeur. Le tableau créé comporte une seule propriété à l'index zéro.

```
args1(0). Name = "Text"  
args1(0). Value = "Jean Dupont"
```

Donne à la propriété le nom "Text" et la valeur "Jean Dupont", qui est le texte qui est inséré lorsque la macro est exécutée.

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

C'est là que tout se passe. Le DispatchHelper transmet au cadre du document (enregistré dans la variable document) la commande .uno:InsertText. Les deux arguments suivants, le nom du cadre et les flags de recherche, vont au-delà du propos de ce guide. Le dernier argument est le tableau de propriétés à utiliser lors de la commande InsertText.

```
end sub
```

Dernière ligne de code qui termine la sous-routine.

## Une macro plus complexe

Lors de la création, il est important de se poser deux questions avant de commencer l'enregistrement :

- Est-ce que cette tâche consiste en un ensemble simple de commandes ?

- Est-ce que les différentes étapes peuvent être agencées de manière à ce que chaque commande laisse le curseur disponible pour la commande suivante ou la saisie de texte ou de données dans le document ?

## Exemple

Vous pouvez avoir besoin de copier des lignes et des colonnes de données à partir d'un site web et de les formater afin d'obtenir un tableau dans un document texte. Un bon exemple figure dans la documentation de l'API de LibreOffice concernant le groupe de constantes FontWeight, qui caractérise l'épaisseur de la police d'un caractère, et qui se trouve sur la page web <http://api.libreoffice.org/docs/common/ref/com/sun/star/awt/FontWeight.html>.

- 1) Copiez les données depuis le site web dans le presse-papiers.
- 2) Pour éviter les formats et les polices étranges, copiez le texte du presse-papiers dans un document Writer en tant que texte non formaté (Figure 5).

DONTKNOW	The font weight is not specified/known.
THIN	specifies a 50% font weight.
ULTRALIGHT	specifies a 60% font weight.
LIGHT	specifies a 75% font weight.
SEMILIGHT	specifies a 90% font weight.
NORMAL	specifies a normal font weight.
SEMIBOLD	specifies a 110% font weight.
BOLD	specifies a 150% font weight.
ULTRABOLD	specifies a 175% font weight.
BLACK	specifies a 200% font weight.

Figure 5 : Texte copié

Vous pouvez utiliser la commande **Tableau > Convertir > Texte en tableau** pour obtenir un tableau, mais le résultat peut ne pas vous convenir : certains caractères inutiles sont conservés (espaces à la fin de la description) et vous souhaitez avoir une première colonne qui comporte la valeur numérique de l'épaisseur, par exemple.

## Enregistrement de la macro

En gardant à l'esprit les deux questions préliminaires ci-dessus, examinez le texte pour voir si une macro peut être enregistrée pour formater le texte. Le nom de la constante est suivi par un espace et une tabulation. La description est suivie de deux espaces.

Une conversion automatique peut être effectuée pour chaque ligne, sauf pour les lignes DONTKNOW et NORMAL qui ne comportent pas de valeur numérique, et pour lesquelles une saisie manuelle sera nécessaire.

Les données peuvent être nettoyées de plusieurs façons. L'exemple ci-dessous utilise les touches du clavier.

- 1) Vérifiez que l'enregistrement de macro est bien actif en sélectionnant **Outils > Options > LibreOffice > Avancé** et en cochant l'option **Activer l'enregistrement de macro**. Par défaut, cette fonction est désactivée lors de l'installation de LibreOffice sur votre ordinateur.
- 2) Positionnez le curseur juste avant le mot THIN.
- 3) Choisissez **Outils > Macros > Enregistrer une macro** pour démarrer l'enregistrement de la macro.
- 4) Appuyez sur *Ctrl+flèche droite* pour déplacer le curseur au début de "specifies".
- 5) Appuyez sur la touche *Retour arrière* deux fois pour supprimer la tabulation et l'espace.

- 6) Appuyez sur la touche *Tab* pour ajouter une tabulation sans espace après le nom de la constante.
- 7) Appuyez sur la touche *Suppr* pour supprimer le s minuscule, puis appuyez sur *Maj+S* pour ajouter un S majuscule.
- 8) Appuyez sur *Ctrl+flèche droite* deux fois pour déplacer le curseur au début du nombre.
- 9) Appuyez sur *Ctrl+Maj+flèche droite* pour sélectionner le nombre et déplacer le curseur avant le signe %.
- 10) Appuyez sur *Ctrl+C* pour copier le texte sélectionné dans le presse-papiers.
- 11) Appuyez sur la touche *Fin* pour déplacer le curseur à la fin de la ligne.
- 12) Appuyez sur la touche *Retour arrière* deux fois pour supprimer les deux derniers espaces.
- 13) Appuyez sur la touche *Début* pour déplacer le curseur au début de la ligne.
- 14) Appuyez sur *Ctrl+V* pour coller le nombre sélectionné au début de la ligne.
- 15) Coller une valeur ajoute un espace supplémentaire, donc appuyez sur la touche *Retour arrière* pour le supprimer.
- 16) Appuyez sur la touche *Tab* pour ajouter une tabulation entre le nombre et le nom.
- 17) Appuyez sur la touche *Début* pour déplacer le curseur au début de la ligne.
- 18) Appuyez sur la touche *flèche bas* pour déplacer le curseur sur la ligne suivante.
- 19) Arrêtez l'enregistrement et enregistrez la macro (voir "Enregistrer une macro" page 10).

Cela prend plus longtemps de lire et d'écrire les différentes étapes que d'enregistrer la macro. Avancez lentement et réfléchissez bien aux étapes au fur et à mesure que vous les effectuez. Avec la pratique, ceci deviendra plus naturel.

Le code généré de la macro du Listing 4 a été modifié, afin de faire apparaître dans les commentaires le numéro de chaque étape ci-dessus.

*Listing 4. Copie de la valeur numérique au début de chaque ligne*

```

sub FormatLigneTableau
rem -----
rem define variables
dim document as object
dim dispatcher as object
rem -----
rem get access to the document
document = ThisComponent.CurrentController.Frame
dispatcher = createUnoService("com.sun.star.frame.DispatchHelper")

rem (4) Appuyez sur Ctrl+flèche droite pour déplacer le curseur
rem au début de "specifies"
dispatcher.executeDispatch(document, ".uno:GoToNextWord", "", 0,
Array())

rem (5) Appuyez sur la touche Retour arrière deux fois pour
rem supprimer la tabulation et l'espace.
dispatcher.executeDispatch(document, ".uno:SwBackspace", "", 0,
Array())
dispatcher.executeDispatch(document, ".uno:SwBackspace", "", 0,
Array())

rem (6) Appuyez sur la touche Tab pour ajouter une tabulation sans
rem espace après le nom de la constante.
dim args4(0) as new com.sun.star.beans.PropertyValue
args4(0).Name = "Text"

```

```

args4(0). Value = CHR$(9)

dispatcher.executeDispatch(document, ".uno:InsertText", "", 0,
args4())

rem (7) Appuyez sur la touche Suppr pour supprimer le s minuscule...
dispatcher.executeDispatch(document, ".uno>Delete", "", 0, Array())

rem (7)... puis appuyez sur Maj+S pour ajouter un S majuscule.
dim args6(0) as new com.sun.star.beans. PropertyValue
args6(0). Name = "Text"
args6(0). Value = "S"

dispatcher.executeDispatch(document, ".uno:InsertText", "", 0,
args6())

rem (8) Appuyez sur Ctrl+flèche droite deux fois pour déplacer le
rem curseur au début du nombre.
dispatcher.executeDispatch(document, ".uno:GoToNextWord", "", 0,
Array())
dispatcher.executeDispatch(document, ".uno:GoToNextWord", "", 0,
Array())

rem (9) Appuyez sur Ctrl+Maj+flèche droite pour sélectionner le
rem nombre et déplacer le curseur avant le signe %.
dispatcher.executeDispatch(document, ".uno:WordRightSel", "", 0,
Array())

rem (10) Appuyez sur Ctrl+C pour copier le texte sélectionné
rem dans le presse-papiers.
dispatcher.executeDispatch(document, ".uno:Copy", "", 0, Array())

rem (11) Appuyez sur la touche Fin pour déplacer le curseur
rem à la fin de la ligne.
dispatcher.executeDispatch(document, ".uno:GoToEndOfLine", "", 0,
Array())

rem (12) Appuyez sur la touche Retour arrière deux fois pour
rem supprimer les deux derniers espaces.
dispatcher.executeDispatch(document, ".uno:SwBackspace", "", 0,
Array())
dispatcher.executeDispatch(document, ".uno:SwBackspace", "", 0,
Array())

rem (13) Appuyez sur la touche Début pour déplacer le curseur
rem au début de la ligne.
dispatcher.executeDispatch(document, ".uno:GoToStartOfLine", "", 0,
Array())

rem (14) Appuyez sur Ctrl+V pour coller le nombre sélectionné
rem au début de la ligne.
dispatcher.executeDispatch(document, ".uno:Paste", "", 0, Array())

rem (15)... appuyez sur la touche Retour arrière pour le supprimer.

```



```

dispatcher.executeDispatch(document, ".uno:SwBackspace", "", 0,
Array())

rem (16) Appuyez sur la touche Tab pour ajouter une tabulation
rem entre le nombre et le nom.
dim args17(0) as new com.sun.star.beans.PropertyValue
args17(0). Name = "Text"
args17(0). Value = CHR$(9)

dispatcher.executeDispatch(document, ".uno:InsertText", "", 0,
args17())

rem (17) Appuyez sur la touche Début pour déplacer le curseur
rem au début de la ligne.
dispatcher.executeDispatch(document, ".uno:GoToStartOfLine", "", 0,
Array())

rem (18) Appuyez sur la touche flèche bas pour déplacer le curseur
rem sur la ligne suivante.
dim args19(1) as new com.sun.star.beans.PropertyValue
args19(0). Name = "Count"
args19(0). Value = 1
args19(1). Name = "Select"
args19(1). Value = false

dispatcher.executeDispatch(document, ".uno:GoDown", "", 0,
args19())


end sub

```

Les mouvements de curseur sont utilisés pour toutes les opérations. Si vous exécutez la macro au début de la ligne DONTKNOW, le mot "weight" est copié au début de la ligne et le mot "The" est changé en "She". Ce n'est pas parfait et vous ne devriez pas exécuter sur les lignes qui ont un format différent. Vous devrez formater ces lignes manuellement.

## Exécuter rapidement une macro

Il est fastidieux de répéter l'exécution de la macro en utilisant **Outils > Macros > Exécuter la macro** depuis la barre de menus principale. Faites-le depuis l'EDI Basic. C'est plus facile et rapide, surtout pour les macros temporaires qui ne seront utilisées que quelques fois puis abandonnées.

- 1) Dans le document Writer, placez le curseur au début de la première ligne à traiter.
- 2) Choisissez **Outils > Macros > Gérer les macros > LibreOffice Basic** pour ouvrir la boîte de dialogue *Macros LibreOffice Basic*.
- 3) Déroulez l'arborescence des bibliothèques, sélectionnez la macro désirée et cliquez sur **Éditer** pour ouvrir l'EDI Basic.
- 4) Dans l'EDI Basic, placez le curseur à un endroit quelconque de la macro FormatLigneTableau.
- 5) Cliquez sur l'icône **Exécuter le BASIC**  de la barre d'outils Macro, ou appuyez sur la touche F5, pour exécuter la macro, autant de fois que vous le souhaitez.

## Limites de l'enregistrement de macro

---

L'enregistrement de macro peut parfois ne pas fonctionner complètement. La compréhension des mécanismes internes de LibreOffice permet d'expliquer quand et pourquoi l'enregistrement de macro ne fonctionne pas.

### Schéma de transmission des commandes

Le but du schéma de transmission des commandes (dispatch framework) est de fournir un accès uniforme aux composants (documents) pour les commandes, qui correspondent habituellement à des éléments de menus. La commande **Fichier > Enregistrer** depuis la barre de menus, le raccourci clavier *Ctrl+S* ou le clic sur l'icône **Enregistrer** sont des commandes qui sont traduites dans la même « commande transmise ».

Le schéma de transmission des commandes est également utilisé pour renvoyer des commandes à l'interface utilisateur. Par exemple, après l'enregistrement d'un document, l'icône **Enregistrer** est désactivée. Dès qu'une modification est effectuée dans le document, l'icône est à nouveau disponible.

Une commande transmise peut être identifiée par un texte, comme `.uno:InsertObject` ou `.uno:GoToStartOfLine`. Cette commande est envoyée au cadre du document et transmise jusqu'à ce qu'un objet soit trouvé pour la prendre en compte.

### Comment l'enregistreur de macro utilise le schéma de transmission des commandes

L'enregistrement de macro mémorise la séquence des commandes transmises telles qu'elles sont générées. Cela marche la plupart du temps. Le problème est que certaines commandes transmises ne sont pas complètes. Par exemple, l'insertion d'un objet génère la ligne de code suivante :

```
dispatcher.executeDispatch(document, ".uno:InsertObject", "", 0, Array())
```

Il n'est pas possible de spécifier quel type d'objet est créé ou inséré. Si un objet est inséré depuis un fichier, il n'est pas possible de spécifier quel fichier doit être inséré.

Si, lors de l'enregistrement d'une macro, vous utilisez la commande **Outils > Options** pour ouvrir et modifier des éléments de configuration, la macro générée n'enregistrera aucune modification. En fait, le code généré est mis en commentaire, et donc il ne sera jamais exécuté.

```
rem dispatcher.executeDispatch(document, ".uno : OptionsTreeDialog », "", 0, Array())
```

Si une boîte de dialogue est ouverte, une commande pour ouvrir cette boîte sera probablement générée et tout ce qui est effectué à l'intérieur de la boîte ne sera habituellement pas enregistré. C'est le cas par exemple des boîtes de dialogue d'organisation des macros, d'insertion de caractères spéciaux ou d'autres. D'autres problèmes possibles lors de l'enregistrement de macro affectent notamment l'insertion de formule, le paramétrage des données utilisateur, les filtres de Calc, les actions sur les formulaires de base de données et l'export des documents vers des PDF chiffrés. Vous ne pouvez jamais être sûr du bon fonctionnement avant de l'essayer. Par exemple, les actions dans la boîte de dialogue de recherche sont correctement enregistrées.

### Autres options

Si l'enregistrement de macro n'est pas capable de résoudre un problème particulier, la solution est d'écrire directement le code en utilisant les objets de LibreOffice. Cependant, cela représente un niveau d'expertise supplémentaire. Il est recommandé de commencer avec des exemples simples

et d'augmenter le périmètre des macros au fur et à mesure de votre apprentissage. La lecture des macros générées est un bon début.

## Organisation des macros

Dans LibreOffice, les macros sont regroupées en modules, les modules sont regroupés en bibliothèques, et les bibliothèques sont regroupées dans des conteneurs de bibliothèques. Une bibliothèque est habituellement utilisée comme principal niveau de regroupement pour contenir toute une catégorie de macros ou toute une application. Les modules sont généralement utilisés pour regrouper des macros selon leurs fonctionnalités, comme l'interface avec les utilisateurs ou les calculs. Chaque macro peut être une sous-routine ou une fonction. La Figure 6 montre un exemple de structure hiérarchique de bibliothèques de macros dans LibreOffice.

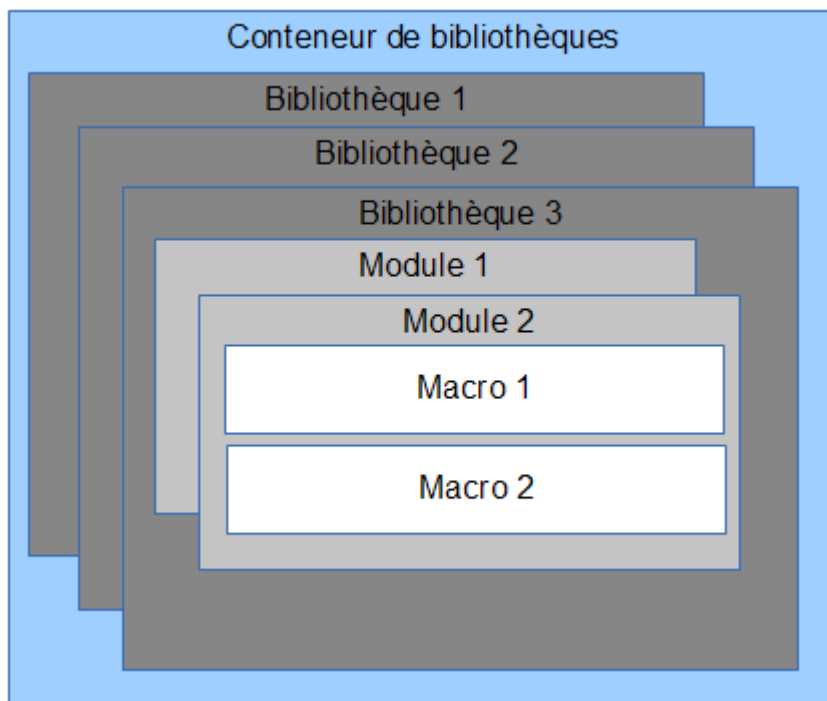


Figure 6 : Hiérarchie des bibliothèques de macros

Choisissez **Outils > Macros > Gérer les macros > LibreOffice Basic** pour ouvrir la boîte de dialogue Macros LibreOffice Basic (Figure 1). Tous les conteneurs de bibliothèques disponibles sont affichés dans la zone *Macro de*. Chaque document est un conteneur de bibliothèques, qui peut contenir plusieurs bibliothèques. L'application elle-même comporte deux conteneurs de bibliothèques, l'un appelé Macros LibreOffice qui contient des macros livrées avec LibreOffice, et l'autre appelé Mes macros destiné à contenir les macros personnelles.

Les macros de Macros LibreOffice sont stockées avec le code exécutable de l'application et peuvent donc ne pas être modifiables si vous n'êtes pas administrateur de l'ordinateur. Ceci permet de protéger ces macros qui ne devraient pas être modifiées. Il est également fortement déconseillé de stocker vos propres macros dans ce conteneur.

À moins que vos macros ne soient applicables qu'à un et un seul document, vous les stockerez probablement dans le conteneur Mes macros. Ce conteneur est stocké dans votre zone utilisateur ou dans votre dossier par défaut.

Si une macro est contenue dans un document, une macro enregistrée va alors essayer de travailler avec ce document, à travers l'usage de "ThisComponent" pour ses actions.

Chaque conteneur de bibliothèques contient une bibliothèque nommée *Standard*. Il est préférable de créer vos propres bibliothèques et de leur donner des noms significatifs, plutôt que d'utiliser la

bibliothèque Standard. Non seulement les noms significatifs sont plus faciles à utiliser, mais ils peuvent également être importés dans d'autres conteneurs de bibliothèques, alors que ce n'est pas possible avec la bibliothèque Standard.



## Attention

LibreOffice vous permet d'importer des bibliothèques dans un conteneur de bibliothèques, mais pas d'écraser la bibliothèque nommée Standard si elle n'est pas vide. Par conséquent, si vous stockez vos macros dans la bibliothèque Standard, vous ne pouvez pas les importer dans un autre conteneur de bibliothèques.

De même qu'il est préférable de donner des noms significatifs à vos bibliothèques, il est prudent de faire de même pour vos modules. Par défaut, LibreOffice utilise les noms Module1, Module2, et ainsi de suite.

Lorsque vous créez vos macros, vous devez décider où les stocker. Enregistrer une macro dans un document est utile si le document est destiné à être partagé et si vous voulez que la macro suive ce document. Cependant, les macros stockées dans le conteneur de bibliothèques Mes macros sont disponibles dans tous les documents.

Les macros ne sont pas disponibles jusqu'à ce que la bibliothèque qui les contient soit chargée. Cependant, la bibliothèque Standard est automatiquement chargée. Une bibliothèque s'affiche différemment selon le fait qu'elle soit chargée ou non : dans l'exemple de la Figure 7, La bibliothèque Standard est chargée, alors que la bibliothèque BibTest ne l'est pas. Pour charger une bibliothèque et les modules qu'elle contient, double-cliquez sur la bibliothèque.

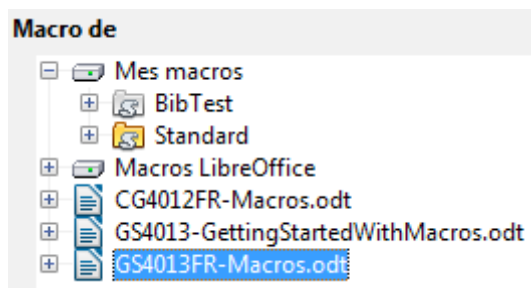


Figure 7 : Bibliothèques chargées ou non

## Où les macros sont-elles enregistrées ?

LibreOffice enregistre les données particulières à un utilisateur dans un sous-dossier du dossier principal de cet utilisateur. Cet emplacement dépend du système d'exploitation. Vous pouvez en avoir un aperçu en choisissant **Outils > Options > LibreOffice > Chemins** et en regardant où les autres données de configuration, les données d'AutoCorrection par exemple, sont enregistrées. Par exemple, sous Windows 10, il s'agit du chemin **C:\Users\. Les macros du conteneur Mes macros sont stockées dans le dossier **LibreOffice\4\user\basic**. Chaque bibliothèque est stockée dans son propre dossier.**

Il n'est pas fondamental de comprendre où les macros sont stockées pour pouvoir les utiliser. Cependant, si vous le savez, vous pouvez les sauvegarder et les partager.

## Importer des macros

La boîte de dialogue *Gestion des macros de LibreOffice* (Figure 2) vous permet d'importer des bibliothèques de macros dans votre document, ainsi que de créer, de supprimer et de renommer des bibliothèques, des modules et des boîtes de dialogue.



## Remarque

Vous ne pouvez pas importer la bibliothèque nommée Standard.

---

- 1) Dans l'onglet *Bibliothèques*, choisissez le conteneur de bibliothèques où vous voulez effectuer l'import des macros et cliquez sur le bouton **Importer**.
- 2) Dans la boîte de dialogue *Importer des bibliothèques*, naviguez jusqu'à l'emplacement contenant la bibliothèque à importer.
  - Si la bibliothèque se trouve dans votre système de fichiers, ce qui est le cas notamment pour celles qui se trouvent dans le conteneur Mes macros, allez dans le dossier portant le nom de cette bibliothèque et choisissez l'un des deux fichiers qu'il contient, à savoir `dialog.xlb` ou `script.xlb`. Peu importe que vous sélectionnez l'un ou l'autre de ces fichiers.
  - Vous pouvez importer plusieurs bibliothèques d'un conteneur en choisissant le fichier `dialog.xlc` ou `script.xlc` se trouvant au niveau du dossier de ce conteneur.
  - Si la bibliothèque se trouve dans un document, sélectionnez ce document.



## Astuce

Selon votre système d'exploitation, il est possible que vous ne puissiez accéder à l'emplacement désiré en naviguant dans la boîte de dialogue *Importer des bibliothèques*. Dans ce cas, saisissez directement le chemin de l'emplacement dans la zone.

---

- 3) Sélectionnez le fichier et cliquez sur le bouton Ouvrir pour ouvrir la boîte de dialogue *Importer des bibliothèques* (Figure 8).

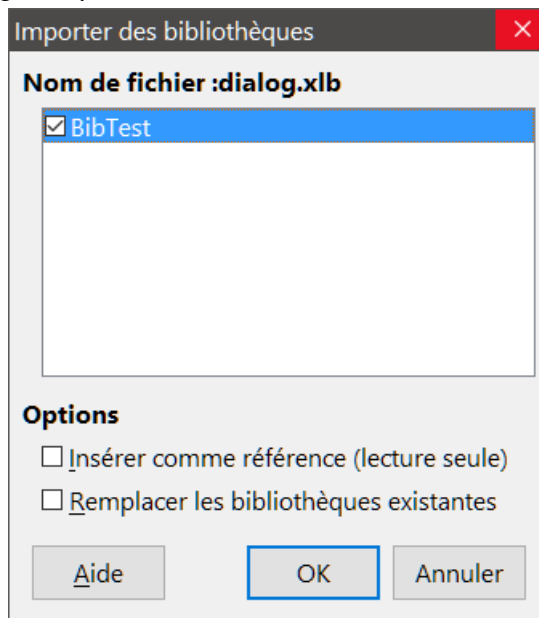


Figure 8 : Choix des options d'import

- 4) Choisissez l'une des options suivantes pour importer les bibliothèques :
  - Si aucune option n'est sélectionnée, les bibliothèques sont copiées dans votre conteneur. Cependant, s'il existe déjà une bibliothèque ayant un nom identique dans le conteneur de destination, cette bibliothèque ne sera pas copiée.
  - Cochez **Remplacer les bibliothèques existantes** si vous voulez écraser les bibliothèques existantes dans le conteneur de destination.

- Cochez **Insérer comme référence (lecture seule)** si vous voulez utiliser la bibliothèque comme référence, mais ne pas l'importer dans votre document. Dans ce cas, la bibliothèque reste à son emplacement initial et est en lecture seule.
- 5) Cliquez sur OK pour importer les bibliothèques de macros que vous avez sélectionnées.

## Comment exécuter une macro ?

Vous pouvez utiliser **Outils > Macros > Exécuter la macro** pour exécuter toutes les macros, mais ceci peut ne pas être très efficace si vous les utilisez fréquemment.

Une technique répandue pour les macros fréquemment utilisées est d'attribuer une macro à une icône de barre d'outils, à un élément de menu, à un raccourci clavier ou à un bouton contenu dans le document. Pour choisir une méthode, vous devez vous poser les questions suivantes :

- La macro doit-elle être disponible uniquement dans un document, ou globalement ?
- Est-ce que la macro est spécifique à un type de document, comme les documents Calc par exemple ?
- À quelle fréquence la macro sera-t-elle utilisée ?

Les réponses vont déterminer où ranger la macro et comment la rendre disponible. Par exemple, vous n'incorporez pas une macro rarement utilisée dans une barre d'outils. Pour vous aider à choisir, voir Tableau 1.

Tableau 1. Méthodes pour appeler une macro

Type	LibreOffice	Type de document	Document
Barre d'outils	Non	Oui	Oui
Menu	Non	Oui	Oui
Raccourci	Oui	Oui	Non
Événement	Oui	Non	Oui

## Barres d'outils, éléments de menus et raccourcis clavier

Pour ajouter un élément de menu, un raccourci clavier ou une icône de barre d'outils qui appelle une macro, utilisez la boîte de dialogue *Personnaliser* (Figure 9) et choisissez l'onglet correspondant. Pour ouvrir cette boîte de dialogue, choisissez **Outils > Personnaliser** depuis la barre de menus ou faites un clic droit sur une barre d'outils et choisissez **Personnaliser la barre d'outils** dans le menu contextuel.

Pour associer une macro à un élément, étendez la catégorie Macros LibreOffice et sélectionnez le module désiré. La liste des macros du module apparaît dans la zone Commandes.

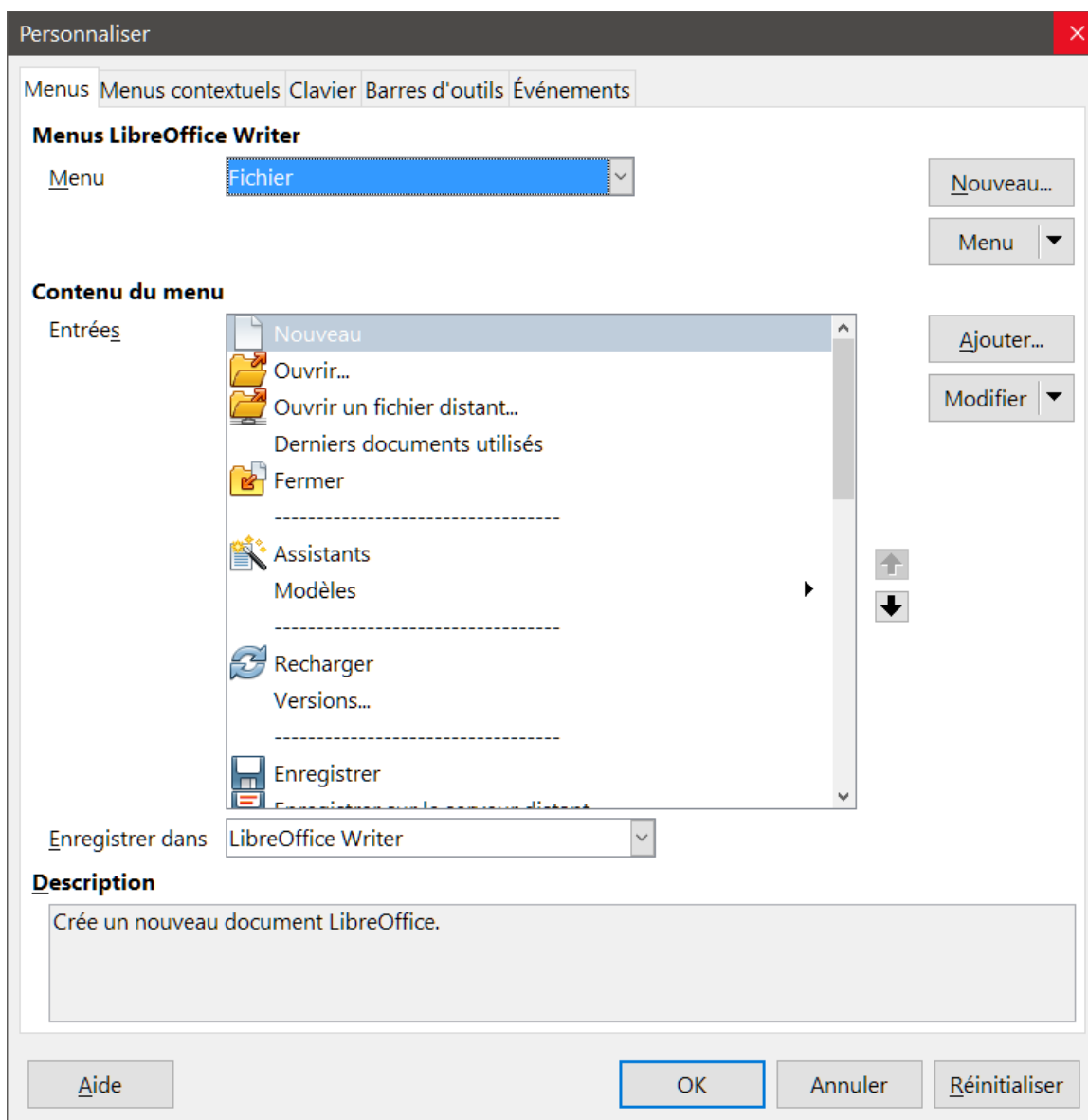


Figure 9 : Boîte de dialogue Personnaliser

## Événements

Dans LibreOffice, quand quelque chose se passe, cela signifie qu'un événement s'est produit : par exemple, l'ouverture d'un document, l'appui d'une touche ou le déplacement du pointeur de la souris. LibreOffice permet que des éléments puissent déclencher l'appel d'une macro ; la macro est alors appelée un gestionnaire d'événement.



### Attention

Faites attention quand vous configurez un gestionnaire d'événement. Par exemple, si vous avez configuré un gestionnaire d'événement appelé à chaque fois qu'une touche est appuyée et si vous appuyez cette touche par erreur, la macro sera tout de même exécutée, ce qui peut produire un résultat que vous ne vouliez pas.

- 1) Choisissez **Outils > Personnaliser** dans la barre de menus pour ouvrir la boîte de dialogue Personnaliser et sélectionner l'onglet *Événements*.

- 2) Dans la liste déroulante *Enregistrer dans*, choisissez si le gestionnaire d'événement concerne tous les documents de LibreOffice ou uniquement le document en cours.
- 3) Une personnalisation souvent paramétrée est d'assigner une macro à l'événement Ouvrir le document, afin d'effectuer un certain nombre de tâches d'initialisation pour un document. Sélectionnez l'événement désiré et cliquez sur le bouton **Macro** pour ouvrir la boîte de dialogue Sélecteur de macro (Figure 4).
- 4) Sélectionnez la macro désirée et cliquez sur OK pour assigner la macro à l'événement. L'onglet *Événements* est alors mis à jour (Figure 10).

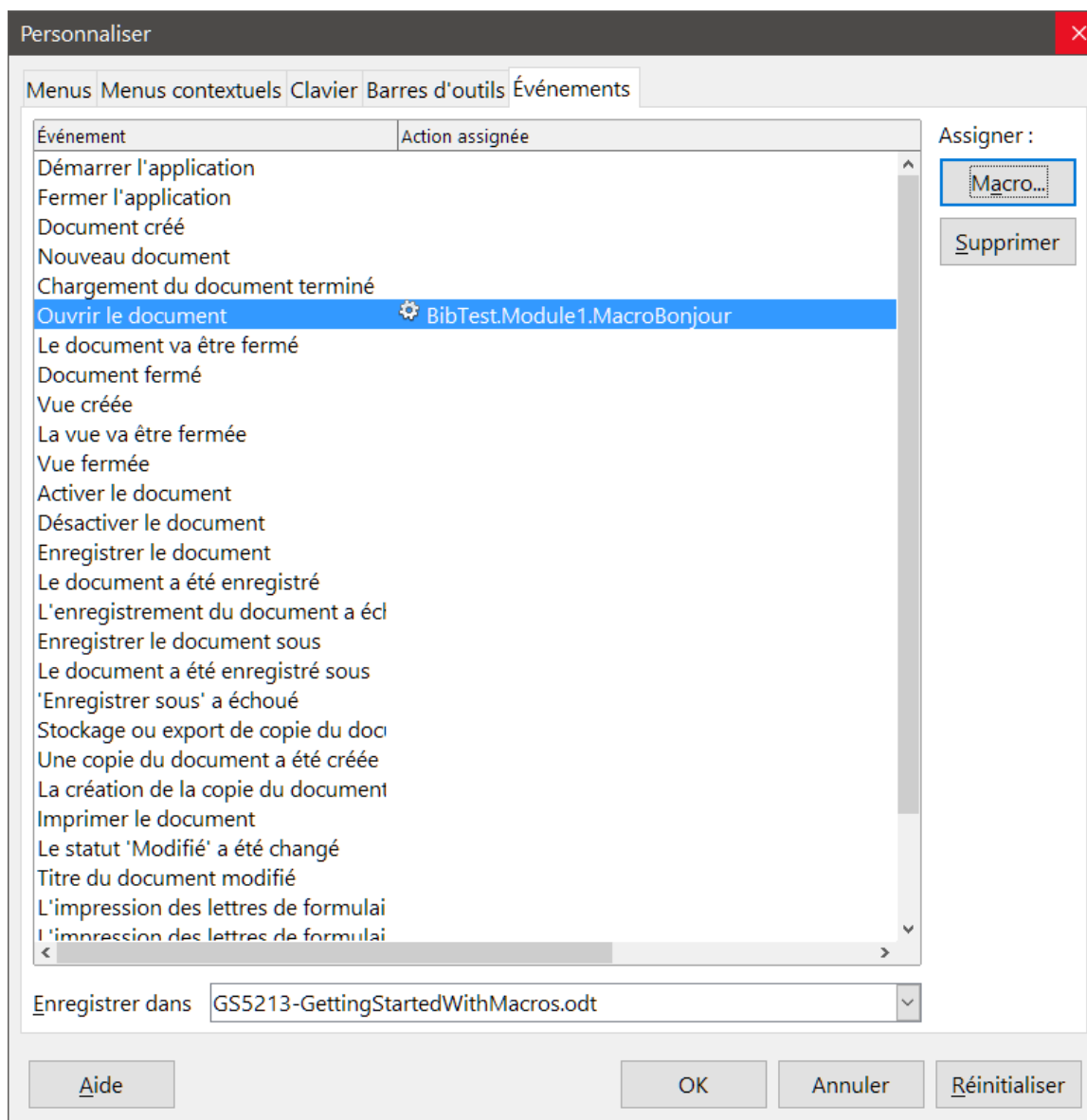


Figure 10 : Onglet *Événements* de la boîte de dialogue *Personnaliser*

Un certain nombre d'objets dans un document peuvent être paramétrés pour appeler une macro quand un événement se produit. Voir les guides de chaque composant pour plus de détails. La façon la plus ordinaire est d'ajouter un contrôle, comme un bouton par exemple, qui, lorsque vous le cliquez, appelle une macro.



## Écrire des macros sans enregistrement

---

Les exemples précédents ont été créés en utilisant l'enregistrement de macro. Vous pouvez également écrire des macros qui accèdent directement aux objets de LibreOffice si vous vous sentez capables d'écrire du code informatique. En d'autres termes, vous pouvez créer une macro qui manipule directement le document.

La manipulation directe des objets internes de LibreOffice est une fonction avancée qui va au-delà du propos de ce chapitre. Voici toutefois un exemple simple pour illustrer ce fonctionnement.

*Listing 5. Ajout du texte "Bonjour" à la fin du document en cours*

```
Sub AjoutBonjour
Dim mDoc
Dim sTextService$
Dim mCurs

REM ThisComponent est le document en cours
mDoc = ThisComponent

REM Vérifie que c'est un document texte
sTextService = "com.sun.star.text.TextDocument"
If NOT mDoc.supportsService(sTextService) Then
MsgBox "Cette macro ne fonctionne qu'avec un document texte"
Exit Sub
End If

REM Récupère le curseur en cours
mCurs = mDoc.currentController.getViewCursor()

REM Déplace le curseur à la fin du document
mCurs.gotoEnd(False)

REM Insère le texte "Bonjour" à la fin du document
mCurs.Text.insertString(mCurs, "Bonjour", False)
End Sub
```