

LibreOffice

Aide-mémoire LibreOffice

# LibreOffice Basic

## Calc

v. 2.0 - 06/04/2021

**Débutant**

Rédigé avec LibreOffice v. 7.0.5 - Plateforme : Toutes

### Documents LibreOffice

#### Document courant

```
Dim oDoc As Object
oDoc = ThisComponent
```

#### Ouvrir un document existant

##### Mode visible

```
Dim oDoc As Object, Chemin As String
Dim Props() 'ici, ce tableau n'est pas initialisé
Chemin = ConvertToURL("C:\Chemin\Vers\UnClasseur.ods")
oDoc = StarDesktop.loadComponentFromURL(Chemin, "_blank", 0, Props())
```

##### Mode invisible

```
Dim oDoc As Object, Chemin As String
Dim Props() As New com.sun.star.beans.PropertyValue
Chemin = ConvertToURL("C:\Chemin\Vers\UnClasseur.ods")
Props().Name = "Hidden" 'le document va être ouvert "caché"
Props().Value = True
oDoc = StarDesktop.loadComponentFromURL(Chemin, "_blank", 0, Props())
```

#### Rendre visible a posteriori

```
oDoc.CurrentController.Frame.ContainerWindow.Visible = True
oDoc.CurrentController.Frame.ContainerWindowToFront()
```

#### Créer un nouveau document Calc

À partir (1) du modèle par défaut ou (2) d'un modèle spécifique.

```
Dim oDoc As Object, Modele As String
Dim Props() 'ici, ce tableau n'est pas initialisé
Modele = "private:factory/scalc" ' (1)
'ou bien
Modele = "C:\Chemin\Vers\UnModeleDeClasseur.ots" ' (2)
oDoc = StarDesktop.loadComponentFromURL(Modele, "_blank", 0, Props())
```

#### Enregistrer un document

##### Le document existe déjà

(équivalent à Fichier > Enregistrer)

Utilisez la méthode store de l'objet document. Ex: ThisComponent.store

##### Le document n'a pas encore été enregistré

(équivalent à Fichier > Enregistrer sous)

```
Dim oDoc As Object, Chemin As String
Dim Props() 'les propriétés d'enregistr. (vide)
Chemin = ConvertToURL("C:\Chemin\Vers\UnClasseur.ods")
oDoc.storeAsURL(Chemin, Props())
```

☞ Si copie, elle **devient** le document actif.

##### Enregistrer une copie

Comme ci-dessus mais avec oDoc.storeToURL(Chemin, Props())

☞ La copie ne devient **pas** le document actif.

##### Enregistrer avec mot de passe (tous modes)

Ajoutez une propriété Password et donnez-lui la valeur requise.

```
Props(n).Name = "Password"
Props(n).Value = "mypwd"
```

#### Fermer un document

Utilisez la méthode close de l'objet document : ThisComponent.close(True)

#### Informations sur le document

##### L'objet document expose des propriétés

Location Le répertoire de stockage du document.  
 DocumentProperties (Object) Des propriétés complémentaires (ci-dessous).  
 ☞ Chaîne vide si pas encore enregistré.

##### DocumentProperties

Author	Nom de l'auteur.	ModifyDate	Date de la dernière modification
CreationDate	Date de création.	Subject	Rubrique Sujet (String).
Description	Rubrique Commentaires.	Title	Rubrique Titre (String).
ModifiedBy	Nom de l'utilisateur ayant modifié le document.	UserDefinedProperties	Propriétés personnalisées (Object).

##### Est-ce un document Calc ?

L'objet oDoc pointe vers le document (ex : oDoc = ThisComponent). CalcOK = oDoc.SupportsService("com.sun.star.sheet.SpreadsheetDocument")

### Calc - Fonctionnalités générales

L'objet oDoc pointe vers le document (ex : oDoc = ThisComponent).

#### Calcul automatique

```
Activé ? (Boolean) Auto = oDoc.isAutomaticCalculationEnabled
Activer/Désactiver oDoc.enableAutomaticCalculation(True) 'False
Forcer le recalcul oDoc.calculate '(formules pas à jour seulement)
oDoc.calculateAll '(tout)
```

#### Protéger le classeur

```
Classeur protégé ? Test = oDoc.isProtected
Protéger le classeur oDoc.protect(MotPasse) [vide possible]
Déprotéger le classeur oDoc.unprotect(MotPasse)
```

### Feuilles (sheets)

L'objet oDoc pointe vers le document (ex : oDoc = ThisComponent).

#### Accéder aux feuilles

On peut travailler avec les objets feuilles (index base 0) :

La feuille active	oFeuil = oDoc.CurrentController.ActiveSheet
Liste des feuilles	oLesFeuilles = oDoc.Sheets
Nombre de feuilles	NbFeuilles = oDoc.Sheets.Count
Objet feuille (par son index)	oFeuil = oDoc.Sheets(index)
Objet feuille (par son nom)	oFeuil = oDoc.Sheets.getByIndex(NomFeuille)
Vérifier existence (nom)	Exist = oDoc.Sheets.hasByName(NomFeuille)
Index d'une feuille	Index = oFeuil.RangeAddress.Sheet

#### Modifier les feuilles

p est la position dans le classeur (base 0).

```
Ajouter une feuille nommée Nom oDoc.Sheets.insertNewByName(Nom, p)
Supprimer une feuille oDoc.Sheets.removeByName(NomFeuille)
Dupliquer une feuille oDoc.Sheets.copyByName(NomSource, NomCible, p)
Déplacer une feuille oDoc.Sheets.moveByName(NomFeuille, p)
```

#### Gérer les feuilles

Soit oFeuil un objet feuille.

Activer une feuille	oDoc.CurrentController.ActiveSheet = oFeuil
Masquer/montrer une feuille	oFeuil.IsVisible = False 'True
Tester la protection	Protegee = oFeuil.IsProtected
Protéger une feuille	oFeuil.protect(MotPasse) '(peut être vide)
Déprotéger une feuille	oFeuil.unprotect(MotPasse)
Colorier l'onglet	oFeuil.tabColor = RGB(255, 255, 0)

#### Lier une feuille

Voir les constantes com.sun.star.sheet.SheetLinkMode.XXX

```
Lier à un fichier oFeuil.link(URL, "", "Text - txt - csv (StarCalc)", _
(ex: CSV) Filtre, com.sun.star.sheet.SheetLinkMode.VALUE)
Libérer un lien oFeuil.setLinkMode(com.sun.star.sheet.SheetLinkMode.NONE)
```

#### Trouver la dernière ligne/colonne utilisée

oFeuil est l'objet feuille à explorer. Ligne et Col sont les informations recherchées.

```
Dim oCur As Object ' curseur sur la cellule
Dim oPlage As Object ' la plage utilisée
Dim Ligne As Long, Col As Long
oCur = oFeuil.createCursorByRange(oFeuil.getCellRangeByName("A1"))
oCur.gotoEndOfUsedArea(True)
oPlage = oFeuil.getCellRangeByName(oCur.AbsoluteName)
Ligne = oPlage.RangeAddress.EndRow
Col = oPlage.RangeAddress.EndColumn
```

### Cellules (cells)

Ci-dessous oCell est un objet cellule.

#### Accéder aux cellules

Soit oFeuil un objet feuille. On peut accéder à un objet cellule :

Parsez coordonnées	oCell = oFeuil.getCellRangeByName("A4")
Par son nom	oCell = oFeuil.getCellRangeByName("TVA")
Parsez coordonnées en X, Y	oCell = oFeuil.getCellByPosition(0,3)
	Avec X=0 (col.A); Y=3 (lig.4)

#### Accéder à la cellule active

Soient oDoc un objet document et ActiveCell l'objet cellule active recherché.

```
If oDoc.currentSelection.supportsService _
("com.sun.star.sheet.SheetCell") Then
'la sélection est bien une cellule
ActiveCell = oDoc.currentSelection
End If
```

#### Sélectionner une cellule

ThisComponent.CurrentController.select(oCell)

#### Coordonnées d'une cellule

Coordonnées (Object)	Coord = oCell.CellAddress
Rang de la feuille (Integer)	NumFeuil = oCell.CellAddress.Sheet
Rang de la colonne (Long)	NumCol = oCell.CellAddress.Column
Rang de la ligne (Long)	NumLig = oCell.CellAddress.Row
Objet feuille conteneur	oFeuil = oCell.Spreadsheet
Coordonnées absolues (String)	Coord = oCell.AbsoluteName

#### (Dé)Protéger des cellules

La propriété oCell.CellProtection peut prendre les valeurs booléennes :

Interdire les modifications	oCell.CellProtection.IsLocked = True
Cacher la formule	oCell.CellProtection.IsFormulaHidden = True
Cacher la cellule	oCell.CellProtection.IsHidden = True
Ne pas imprimer la cellule	oCell.CellProtection.IsPrintHidden = True

#### Accéder au contenu d'une cellule

##### Propriétés

Lire un contenu texte	MonTexte = oCell.String
Lire un contenu numérique	UnNombre = oCell.Value
Lire une formule (US)	LaFormule = oCell.Formula
Lire une formule (langue locale)	LaFormule = oCell.FormulaLocal
Connaître le type de contenu	LeType = oCell.Type 'voir ci-dessous
Vider une cellule	oCell.String = ""

##### Types de contenus (propriété Type)

Les constantes com.sun.star.table.CellContentType.XXX permettent de connaître le type d'information que portent les cellules (oCell.Type, ci-dessus) :

EMPTY	Cellule vide	VALUE	Contenu valeur numérique
TEXT	Contenu texte	FORMULA	Contenu formule

#### Écrire dans une cellule

Remplacer le <b>texte</b> existant	oCell.String = "Coucou !"
Remplacer la <b>valeur</b> existante	oCell.Value = 1,234
Remplacer la formule existante	oCell.Formula = "=AND(A1="OUI";A2="OK")"
Remplacer la formule existante	oCell.FormulaLocal = "=ET(A1="OUI";A2="OK")"

## Plages (ranges)

Plage = ensemble de cellules (y compris une seule) : Dim Pl As Object

### Accéder aux plages

Soit oFeuil un objet feuille. On récupère un objet plage oPlage :

```
Par ses coordonnées oPlage = oFeuil.getCellRangeByName("C2:G14")
Par son nom oPlage = oFeuil.getCellRangeByName("NomDePlage")
Par ses coordonnées oPlage = oFeuil.getCellRangeByPosition(2, 1, 6, 13)
(X1, Y1, X2, Y2)
Arbitrairement oPlage = ThisComponent.Sheets.getCellRangeByPosition(
(ex 3e feuille classeur) 2, 1, 6, 13, 2)
```

### Accéder à la plage active

Comme pour accéder à la cellule active, mais vérifiez  
"com.sun.star.sheet.SheetCellRange" ou "[...]SheetCellRanges".

### Sélectionner une plage

ThisComponent.CurrentController.select(oPlage) où oPlage est un objet plage.

### Coordonnées d'une plage

Les index des lignes/col. sont des Long. Ceux des feuilles des Integer.  
Coordonnées (Object) Coord = oPlage.RangeAddress  
Rang de la feuille Rang = oPlage.RangeAddress.Sheet  
Rang de la colonne (ht/gauche) NumCHG = oPlage.RangeAddress.StartColumn  
Rang de la ligne (ht/gauche) NumLHG = oPlage.RangeAddress.StartRow  
Rang de la colonne (bas/droite) NumCBD = oPlage.RangeAddress.EndColumn  
Rang de la ligne (bas/droite) NumLBD = oPlage.RangeAddress.EndRow  
Objet feuille conteneur oFeuil = oPlage.Spreadsheet  
Coordonnées absolues (String) Coord = oPlage.AbsoluteName

### Plages nommées

L'objet oDoc pointe vers le document. Avec Dim oPlages As Object  
Les plages nommées oPlages = oDoc.NamedRanges  
Nombre (Long) Nb = oPlages.Count  
Accéder à une plage (par index) oPlage = oPlages(index)  
Vérifier existence (pour un nom) Existe = oPlages.hasByName(Nom)  
Accéder à une plage (par nom) oPlage = oPlages.getByName(Nom)  
Ajouter une plage oPlages.addNewByName(Nom, Coord, \_  
Coord : coordonnées de la plage CellRef.CellAddress, 0)  
CellRef : objet cellule de référence.  
Supprimer (par nom) oPlages.removeByName(Nom)

### Effacer une plage

Effacer le contenu de oPlage oPlage.clearContents(ModeEff)  
ModeEff est une valeur qui détermine le type d'effacement. Les constantes  
com.sun.star.sheet.CellFlags.XXX permettent ce choix (combinaison avec +):  
ANNOTATION Commentaires | STRING Texte  
DATETIME Nombres au format date-heure | VALUE Nombres (sauf date-heure)  
FORMULA Formules

### Accéder au contenu des cellules d'une plage

oPlage.DataArray est un tableau des valeurs des cellules pour oPlage.

### Recopier le contenu d'une plage vers une autre

Soient deux plages Source et Cible, de mêmes dimensions.  
Recopier le contenu (valeurs) de Source Cible.DataArray = Source.DataArray  
dans Cible.

### Écrire des valeurs dans une plage

Soient oPlage un objet plage et Tablo un tableau, de mêmes dimensions, dont les valeurs  
sont à transférer dans la plage.

```
Dim Tablo As Variant
Tablo = oPlage.DataArray 'Tablo prend les dim de la plage
!'(donner des valeurs aux éléments du tableau)
oPlage.DataArray = Tablo
```

oPlage.DataArray est un tableau imbriqué : utilisez oPlage.DataArray(i)(j)

### Parcourir les cellules d'une plage

À partir d'une collection (Plages.Cells) créez une enumeration. Celle-ci est parcourue en  
appelant ses propriétés hasMoreElements et NextElement (oDoc est le classeur) :

```
Dim oPlages As Object, oEnum As Object
oPlages = oDoc.CreateInstance("com.sun.star.sheet.SheetCellRanges")
oPlages.insertByName("", oPlage)
oEnum = oPlages.Cells.CreateEnumeration
Do While oEnum.hasMoreElements
  Cellule = oEnum.NextElement
  'appliquer un traitement à partir de l'objet Cellule
Loop
```

Les cellules vides ne sont pas parcourues !

### Plages : divers

Fusionner les cellules de oPlage oPlage.Merge

### Types de plages

Selon le mode d'accès à une plage, celle-ci implémente l'un des services :

- com.sun.star.sheet.SheetCell
- com.sun.star.table.CellRange
- com.sun.star.sheet.NamedRange
- com.sun.star.sheet.SheetCellRange
- com.sun.star.sheet.SheetCellRanges

En fonction du service implémenté, les plages devront être utilisées différemment.  
Testez au moyen de leur méthode supportsService() (ex. ci-dessous)

### Plage ou cellule ?

Pour connaître le type d'un objet, testez l'objet plage/cellule avec supportsService() :  
If MonObj.supportsService(nom\_du\_service) Then ...

Remplacez nom\_du\_service par :

Cellule ? "com.sun.star.sheet.SheetCell" ①  
Plage simple ? "com.sun.star.sheet.SheetCellRange" ②  
Plage multiple ? "com.sun.star.sheet.SheetCellRanges" ③

Test cellule toujours avant plage simple car une cellule est aussi une plage simple !

## Lignes/Colonnes (rows/columns)

Lignes et colonnes sont des propriétés des objets Sheet et Range.

### Généralités

Lignes (Objet oLignes) oLignes = oPlage.Rows  
Colonnes (Objet oCols) oCols = oPlage.Columns  
Nombre NbLig = oPlage.Rows.Count  
NbCol = oPlage.Columns.Count  
Une ligne (Objet oLigne) (base 0) oLigne = oPlage.Rows(index)  
Une colonne (Objet oCol) (base 0) oCol = oPlage.Columns(index)

### Propriétés des lignes/colonnes

S'appliquent à des objets Ligne ou Lignes (resp. Colonne ou Colonnes).  
Visible/Masquée (Boolean) IsVisible = True  
Largeur optimale ou non (Boolean) OptimalWidth = True

### Insérer/supprimer des lignes/colonnes

Soit un objet oLougC. PremPos et Nombre sont les positions de début et le nombre de lignes  
(resp. colonnes) à ajouter/supprimer (Long).

Insérer oLougC.insertByIndex(PremPos, Nombre)  
Effacer oLougC.clearContents(ModeEff) [ModeEff : voir Effacer une plage]  
Supprimer oLougC.removeByIndex(PremPos, Nombre)

### Fixer lignes/colonnes

Uniquement sur un classeur visible.  
Utiliser l'objet Contrôleur : oContrlr = ThisComponent.CurrentController  
Y en a-t-il ? Fixes = oContrlr.hasFrozenPanels  
Fixer (X, Y) oContrlr.freezeAtPosition(1, 2)  
Supprimer oContrlr.freezeAtPosition(0, 0)

### Appeler une fonction Calc

Utilisez le service "com.sun.star.sheet.FunctionAccess"

#### Principe

```
Dim FCalc As Object, Resultat As (selon contexte)
Dim Params As (selon contexte), NonFonction As String
FCalc = CreateUnoService("com.sun.star.sheet.FunctionAccess")
Resultat = FCalc.callFunction(NomFonction, Params)
```

Son nom, les arguments et le type du résultat dépendent de la fonction choisie.

Le nom de la fonction doit être le nom anglais de celle-ci.  
Pour le connaître, basculez temporairement l'affichage des noms de fonctions dans Calc,  
par Outils > Options > LibreOffice Calc > Formule, Utiliser les noms [...] anglais.

#### Exemple 1 (fonction SOMME())

```
Dim FCalc As Object, Resultat As Long
FCalc = CreateUnoService("com.sun.star.sheet.FunctionAccess")
Resultat = FCalc.callFunction("SUM", Array(1, 55, 321, 8))
```

#### Exemple 2 (fonction ARRONDI())

```
Dim FCalc As Object, Resultat As Double
Dim Params(1) As Variant
Params(0) = 1,2345 'nb à arrondir
Params(1) = 3 '3 décimales
FCalc = CreateUnoService("com.sun.star.sheet.FunctionAccess")
Resultat = FCalc.callFunction("ROUND", Params())
```

### Créer une fonction Calc

#### Création

Exemple : calcul de la surface d'un trapèze (S = ((B + b) / 2) × H)

```
Function SurfaceTrapeze(GB As Double, PB As Double, H As Double) As Double
  SurfaceTrapeze = ((GB + PB) / 2) * H
End Function
```

#### Utilisation dans Calc

Si A2 est la grande base, A3 la petite base et A4 la hauteur, la surface du trapèze s'obtient en insérant la formule suivante dans une cellule : =SURFACETRAPEZE (A2 ; A3 ; A4)

La macro reçoit les valeurs des arguments et non pas l'objet cellule émetteur.

La macro renvoie une valeur. Elle ne peut pas agir sur une cellule.

La fonction doit être placée dans une bibliothèque accessible au moment de son utilisation (ex : Standard du document ou de l'utilisateur) (sinon, erreur #VALEUR!)

#### Crédits

**Auteur :** Jean-François Nifenecker – [jean-francois.nifenecker@laposte.net](mailto:jean-francois.nifenecker@laposte.net)

*Nous sommes comme des nains assis sur des épaules de géants. Si nous voyons plus de choses et plus lointaines qu'eux, ce n'est pas à cause de la perspicacité de notre vue, ni de notre grandeur, c'est parce que nous sommes élevés par eux. (Bernard de Chartres [attr.])*

#### Historique

Version	Date	Commentaires
2.0	06/04/2021	Révision de la mise en forme ; corrections.

#### Licence

Cet aide-mémoire est placé sous licence

**Creative Commons BY-SA v3 (fr)**

Informations :

<https://creativecommons.org/licenses/by-sa/3.0/fr/>

