Programmation en LibreOffice Basic – première partie

Samuel TOUBON

2A 2017-2018

ENSAI
École nationale de la statistique et de l'analyse de l'information

Organisation du cours

Volumétrie :

- 1h30 de cours
- 16h30 de TP (5 $\times$ 3h + 1h30)

Évaluation :

- 1 examen final

Introduction

- LibreOffice est un **fork** de OpenOffice.org
- Il hérite de son langage de macros : OpenOffice.org Basic qu'il renomme LibreOffice Basic (LOB).
- Ces langages font partie de la famille Basic, au même titre que VBA, et sont donc très proches.
- LibreOffice, contrairement à Microsoft Office, est ouvert à d'autres langages d'automatisation des tâches : Python, BeanShell, JavaScript, Java, C++.
- Le plus simple reste Basic, il est de plus mieux intégré, avec un éditeur plus sympathique.

ENSAI École nationale de la statistique et de l'analyse de l'information

Introduction

LibreOffice Basic peut être divisé en quatre composants :

- Le **langage** LibreOffice Basic.
- La **bibliothèque d'exécution** qui fournit des fonctions standard ne faisant pas directement référence à LibreOffice.
- L'**API** LibreOffice qui permet de manipuler les documents.
- L'**éditeur de boîtes de dialogues**.

Les deux premiers éléments sont compatibles avec VBA, à quelques détails près.

ENSAI École nationale de la statistique et de l'analyse de l'information

Sommaire de la première partie

Le langage LibreOffice Basic

Idée générale par rapport à Java :

- Seuls des éléments de syntaxe changent, mais il n'y a **aucun concept nouveau**.
- Dans ce cours, on manipulera des objets issus de l'API, mais on ne définira pas ses propres classes.

LOB :

- est un langage **interprété**,
- ne peut être utilisé en dehors de LibreOffice.

ENSAI École nationale de la statistique et de l'analyse de l'information

Généralités

Lignes de programme

- Cas général : une instruction = une ligne
- Pas de ;
- Possibilité d'utiliser le _ pour écrire une instruction sur plusieurs lignes
- Possibilité d'utiliser le : pour écrire plusieurs instructions par ligne

Commentaires

- Introduits par le symbole '
- Ou le mot clé `Rem`

ENSAI École nationale de la statistique et de l'analyse de l'information

Noms des variables

Noms de variables

- LOB insensible à la casse
- Règles habituelles : pas de caractères spéciaux, le premier caractère doit être une lettre...

Notation hongroise

- Souvent rencontrée dans les langages Basic
- Utilisation non obligatoire
- On rencontrera dans les exemples `o` (Object), `s` (String)

ENSAI École nationale de la statistique et de l'analyse de l'information

Variables

Comportement par défaut

- Déclaration **implicite** des variables

```
1 a = b + c
```

Combien de variables déclarées ? Que valent-elles ?

Variables

Le problème :

```
1 Sub Main
2     Dim cellule as Integer
3     cellule = 0
4     celule = cellule + 1
5     MsgBox cellule
6 End Sub
```

Qu'est-ce qui est affiché ?

Variables

La solution

- Imposer une déclaration **explicite** des variables avec `Option Explicit`.

```
1 Option Explicit
2
3 Sub Main
4     Rem TODO
5 End Sub
```

`Option Explicit` doit se trouver sur la première ligne de chaque module.

ENSAI École nationale de la statistique et de l'analyse de l'information

Variables – chaînes de caractères

```
1 Dim MyString As String
2 MyString = " This is a test"
```

```
1 Dim MyString As String
2 MyString =    "This string is so long that it " + _
3               "has been split over two lines."
```

```
1 Dim MyString As String
2 MyString = "a ""-quotation mark."
```

Variables – numériques

5 types des bases pour les numériques :

- `Integer` pour les entiers de - 32 768 à 32 767
- `Long` pour les entiers de - 2 147 483 648 à 2 147 483 647
- `Single` pour les réels positifs ou négatifs $1,401298 \times 10^{-45}$ à $3,402823 \times 10^{38}$
- `Double` pour les réels positifs ou négatifs de $4,94065645841247x10^{-324}$ à $1,79769313486232x10^{308}$
- `Currency` pour les valeurs monétaires entre -922 337 203 685 477,5808 et +922 337 203 685 477,5807

Il est possible d'utiliser la notation exponentielle : `A = 1.43E2`

ENSAI École nationale de la statistique et de l'analyse de l'information

Variables – autres

Booléens

Il existe un type `Boolean` qui prend `True` ou `False`.
Si on stocke un entier dans une variable booléenne, celle-ci vaut `False` si l'entier vaut 0, `True` dans les autres cas.

Dates et heures

On peut déclarer une variable de type `Date`. Les comparaisons et les opérations arithmétiques entre dates sont alors possibles.

Constantes

On peut déclarer des constantes grâce à la syntaxe suivante :

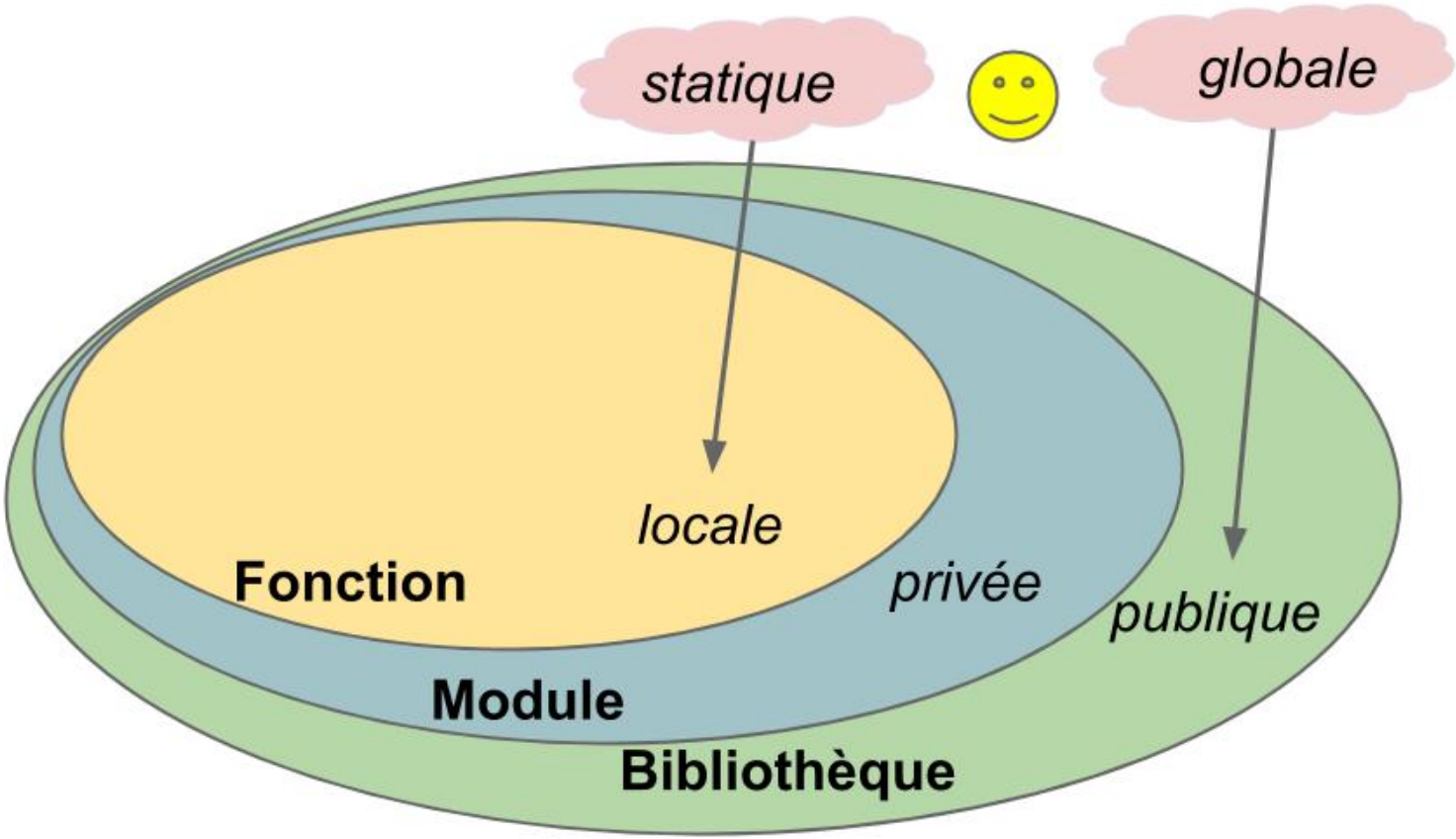
```
Const B As Double = 10.
```

ENSAI École nationale de la statistique et de l'analyse de l'information

Variables – type Variant

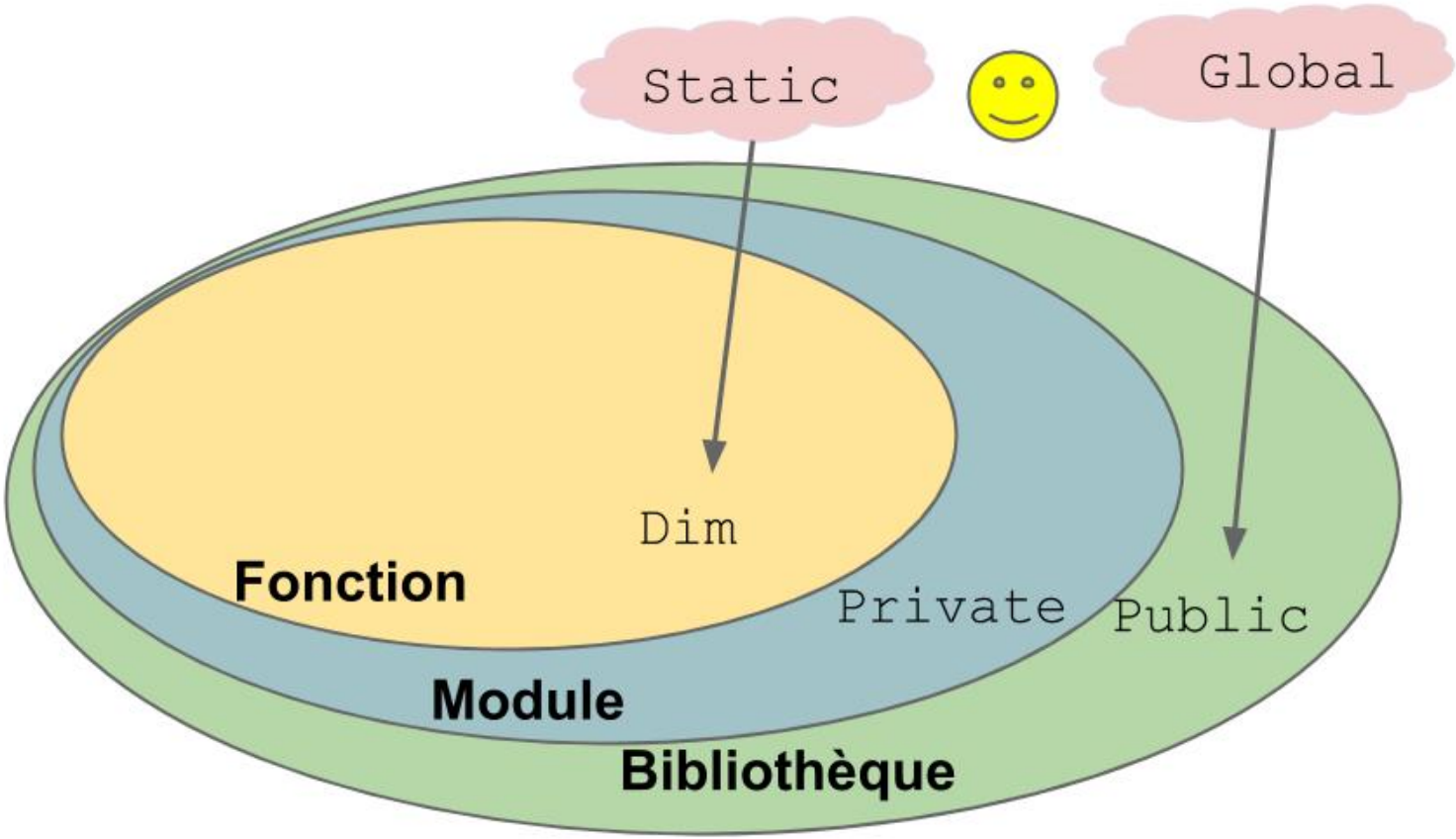
```
1 Dim MyVar As Variant
2 MyVar = "Hello World"
3 MyVar = 1
4 MyVar = 1.0
5 MyVar = True
```

Portée des variables *



ENSAI École nationale de la statistique et de l'analyse de l'information

Portée des variables *



ENSAI École nationale de la statistique et de l'analyse de l'information

Tableaux

Dimensions

`Dim MyInteger(3) As Integer` déclare un tableau de 4 entiers indexés de 0 à 3.

Index

On peut modifier le champ des index à la déclaration :
`Dim MyInteger(5 To 10) As Integer`. Des bornes négatives sont possibles.

Indice de départ

Si on préfère compter à partir de 1, on peut utiliser `Option Base 1`. La commande `Dim MyInteger(3) As Integer` initialise toujours un tableau de 4 entiers dans ce cas, indicés de 1 à 4.

VBA

En VBA, lorsqu'on modifie l'indice de départ, le nombre d'éléments créés n'est pas conservé. `Dim MyInteger(3) As Integer` initialise un tableau de taille 3. Pour que LibreOffice se comporte comme VBA, on peut utiliser `Option Compatible`.

Matrices

Déclaration

On peut déclarer une matrice de dimension 2 ainsi :
`Dim MyIntArray(5, 5) As Integer`. Celle-ci possède 36 cellules.

Dimensions

Le nombre de dimensions d'une matrice n'est limitée que par la mémoire de la machine.

ENSAI École nationale de la statistique et de l'analyse de l'information

Redimensionnement

Redimensionnement

On peut redimensionner dynamiquement un tableau ou une matrice grâce à la commande `ReDim MyArray(10) As Integer`.

Préservation des données

Si l'on souhaite redimensionner un tableau ou une matrice sans écraser les données contenues, il faut utiliser l'option `Preserve` :

```
ReDim Preserve MyArray(20) As Integer
```

En VBA, seule la limite supérieure de la dernière dimension d'une matrice peut être modifiée en préservant les données.

ENSAI École nationale de la statistique et de l'analyse de l'information

Déterminer les dimensions

Fonctions LBound et UBound

Elles permettent de récupérer les indices.

```
1 Dim MyArray(10) As Integer
2 '... des instructions redimensionnent le tableau
3 MsgBox(LBound(MyArray)) ' affiche la borne basse (0)
4 MsgBox(UBound(MyArray)) ' affiche la borne haute (10)
```

En multi-dimensionnel :

```
1 Dim MyArray(10, 13 to 28) As Integer
2 MsgBox(LBound(MyArray,2)) ' affiche 13
3 MsgBox(UBound(MyArray,2)) ' affiche 28
```

ENSAI École nationale de la statistique et de l'analyse de l'information

Lecture et écriture

```
1 Dim MyArray(10) As Integer
2 MyArray(0) = 57
3 MsgBox(MyArray(0))
```

En multi-dimensionnel :

```
1 Dim MyArray(10, 13 To 28) As Integer
2 MyArray(0,13) = 57
3 MsgBox(MyArray(0,13))
```

ENSAI École nationale de la statistique et de l'analyse de l'information

Opérateurs

Arithmétiques

- `+` : nombres, dates, chaînes
- `-` : idem
- `*` : nombres
- `/` : nombres
- `^` : nombres (élévation à la puissance)
- `Mod` : nombres (opération modulo)

Logiques

- `And` : et
- `Or` : ou
- `Xor` : ou exclusif
- `Not` : négation
- `Eqv` : les deux éléments ont la valeur `True` ou `False`
- `Imp` : si la première expression est vraie, alors la seconde l'est également

Opérateurs

Comparaison

- = : nombres, dates, chaînes
- <> : idem
- < : idem
- <= : idem
- > : idem
- >= : idem

VBA

VBA introduit l'opérateur de comparaison `Like` qui n'est pas disponible en LOB.

If... Then... Else

```
1 If A = 0 Then
2   B = 0
3 ElseIf A < 3 Then
4   B = 1
5 Else
6   B = 2
7 End If
```

Select... Case

```
1  Select Case DayOfWeek
2    Case 1:
3      NameOfWeekday = "Sunday"
4    Case 2:
5      NameOfWeekday = "Monday"
6    Case 3:
7      NameOfWeekday = "Tuesday"
8    Case 4:
9      NameOfWeekday = "Wednesday"
10   Case 5:
11     NameOfWeekday = "Thursday"
12   Case 6:
13     NameOfWeekday = "Friday"
14   Case 7:
15     NameOfWeekday = "Saturday"
16 End Select
```

ENSAI École nationale de la statistique et de l'analyse de l'information

Select... Case

```
1  Select Case Var
2    Case 1 To 5
3      ' ... Var is between the numbers 1 and 5
4    Case 6, 7, 8
5      ' ... Var is 6, 7 or 8
6    Case > 8
7      ' ... Var is greater than 8
8    Case Else
9      ' ... all other instances
10 End Select
```

For... Next

```
Dim I
For I = 1 To 10
  ' ... Inner part of loop
Next I
```

```
Dim I
For I = 1 To 10 Step 0.5
  ' ... Inner part of loop
Next I
```

```
Dim I
For I = 10 To 1 Step -1
  ' ... Inner part of loop
Next I
```

For Each

```
1 Const d1 = 2
2 Const d2 = 3
3 Const d3 = 2
4 Dim a(d1, d2, d3)
5 For Each i In a()
6     ' ... Inner part of loop
7 Next i
```

Do... Loop

Quelle ménagerie !

En LOB, il existe **cinq** façons d'écrire une boucle `While`. En voici deux.

```
1 Do While A > 10
2     ' ... loop body
3 Loop
```

```
1 Do
2   ' ... loop body
3 Loop While A > 10
```

Procédures

```
1 Sub Test
2   ' ... here is the actual code of the procedure
3 End Sub
```

Fonctions

```
1 Function Test ' cette fonction retourne un Variant
2   Test = 12
3   ' ...
4   Test = 123 ' valeur retournee
5 End Function
```

```
1 Function Test As Integer
2   ' ... here is the actual code of the function
3 End Function
```

Passage par valeur ou par référence

Passage par référence par défaut

Les paramètres sont normalement passés par référence dans LibreOffice Basic. Les modifications apportées aux variables sont conservées lorsque la procédure ou la fonction se termine !

Passage par valeur

On peut forcer un passage par valeur avec la syntaxe suivante :

```
1 Sub ChangeValue(ByVal TheValue As Integer)
2   TheValue = 20
3 End Sub
```

En VBA, les arguments sont passés par valeur par défaut.

ENSAI École nationale de la statistique et de l'analyse de l'information

Paramètres facultatifs (=optionnels)

```
1  Sub Test(A As Integer, Optional B As Integer)
2    Dim B_Local As Integer
3    ' Check whether B parameter is actually present
4    If Not IsMissing (B) Then
5      B_Local = B       ' B parameter present
6    Else
7      B_Local = 0       ' B parameter missing -> default value 0
8    End If
9    ' ... Start the actual function
10 End Sub
```

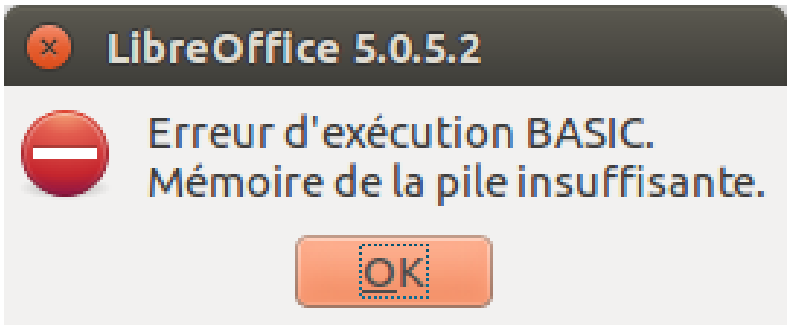
Le mot-clé ParamArray proposé par VBA n'est pas pris en charge en LOB.

ENSAI École nationale de la statistique et de l'analyse de l'information

Récursivité

Récursivité

- L'appel récursif des fonctions est possible en LOB.
- La taille de la pile dépend du système d'exploitation et de la machine. À l'Ensai, quelque part entre 1 500 et 2 000, avec un plantage en cas de dépassement [1].
- Sur un PC avec Windows 7, 8 Go de RAM, LibreOffice 5.2, quelque part entre 5 000 et 6 000, avec un message d'erreur poli.
- Sur un PC avec Ubuntu 15.10, 2 Go de RAM, LibreOffice 5.0, quelque part entre 9 000 et 9 500, avec un message d'erreur poli.



ENSAI École nationale de la statistique et de l'analyse de l'information

1. C'était vrai en juin 2017 avec LibreOffice 4, à retester maintenant.

Récursivité

```
1  Sub Main
2      MsgBox(Test(9500)) 'entre 9 000 et 9 500
3  End Sub
4
5  Function Test(A As Integer)
6      If A>0 Then
7          Test = 1 + Test(A-1)
8      Else
9          Test = 0
10     End IF
11 End Function
```

Instruction On Error

Un mécanisme de gestion des erreurs est inclus dans LOB.

```
1 Sub Test
2   On Error Goto ErrorHandler
3   ' ... undertake task during which an error may occur
4   Exit Sub
5   ErrorHandler:
6     ' ... individual code for error handling
7 End Sub
```

Instruction Resume

```
1 ErrorHandler:
2   ' ... individual code for error handling
3   Resume Next
```

```
1 ErrorHandler:
2   ' ... individual code for error handling
3   Resume Proceed
4
5 Proceed:
6   ' ... the program continues here after the error
```

On Error Resume Next

```
1 Sub Test
2   On Error Resume Next
3   ' ... perform task during which an error may occur
4 End Sub
```

Avertissement

La commande On Error Resume Next doit être utilisée avec précaution, car elle a un effet global.

Information sur les erreurs

```
1 MsgBox "Error " & Err & ": " & Error$ & " (line : " & Erl & ")"
```

- La variable `Err` le numéro de l'erreur qui est survenue.
- La variable `Error$` contient une description de l'erreur.
- La variable `Erl` contient le numéro de la ligne où l'erreur s'est produite.

Portée

Les informations de statut restent valables jusqu'à ce que le programme rencontre une commande `Resume` ou `On Error`, qui les réinitialise. On peut utiliser `On Error Goto 0` pour réinitialiser ces informations.

VBA regroupe les messages d'erreur dans un objet nommé Err. La méthode Err.Clear() de l'objet Err réinitialise le statut d'erreur.

Sommaire

ENSAI École nationale de la statistique et de l'analyse de l'information

Conversion implicite

```
1 Dim A As String
2 Dim B As Integer
3 Dim C As Integer
4
5 B = 1
6 C = 1
7 A = B + C
```

Que vaut A ?

Conversion explicite

- `CStr(Var)` : convertit tout type de données en chaîne de caractères.
- `CInt(Var)` : convertit tout type de données en valeur entière.
- `CLng(Var)` : convertit tout type de données en valeur longue.
- `CSng(Var)` : convertit tout type de données en valeur simple.
- `CDbl(Var)` : convertit tout type de données en valeur double.
- `CBool(Var)` : convertit tout type de données en valeur booléenne.
- `CDate(Var)` : convertit tout type de données en date.

ENSAI École nationale de la statistique et de l'analyse de l'information

Fonctions de test

- `IsNumeric(Value)` : détermine si une valeur est un nombre.
- `IsDate(Value)` : détermine si une valeur est une date.
- `IsArray(Value)` : détermine si une valeur une liste.

Booléens

Il n'existe pas de fonction similaire pour les booléens, mais vous pouvez la construire vous-mêmes.

Programmer IsBoolean.

Cas limite

Conversion impossible

- Si une chaîne contenant une valeur non numérique est assignée à un nombre, LOB ne génère pas d'erreur, mais transmet la valeur 0 à la variable.
- Ceci est encore vrai si l'on utilise la fonction de conversion explicite.

VBA aurait généré une erreur.

Accès à une partie de la chaîne

```
1 Dim MyString As String
2 Dim MyResult As String
3 Dim MyLen As Integer
4
5 MyString = "This is a small test"
6 MyResult = Left(MyString,5)       ' Provides the string "This "
7 MyResult = Right(MyString, 5)     ' Provides the string " test"
8 MyResult = Mid(MyString, 8, 5)    ' Provides the string " a sm"
9 MyLen = Len(MyString)             ' Provides the value 21
```

Formatage des chaînes

Cinq substituants :

- 0
- #
- .
- ,
- $

```
1    MyFormat = "#,##0.00 \€"
2    MyString = Format ( -1579.8 , MyFormat ) ' -1 579,80 €"
3    MyString = Format (1579.8 , MyFormat ) ' 1 579,80 €"
4    MyString = Format (0.4 , MyFormat ) ' 0,40 €"
5    MyString = Format (0.434 , MyFormat ) ' 0,43 €"
```

ENSAI École nationale de la statistique et de l'analyse de l'information

Assignation des dates

Formalisme conseillé pour les dates (année, mois, jour) :

```
1 Dim MyVar As Date
2 MyVar = DateSerial (2001, 1, 24)
```

Pour les temps (heure, minute, seconde) :

```
1 Dim MyVar As Date
2 MyVar =  TimeSerial(11, 23, 45)
```

Autre formalisme possible :

```
1 Dim MyDate As Date
2 MyDate = "24.1.2002"
```

Extraction de données

- `Day(MyDate)` : retourne le jour du mois de MyDate.
- `Month(MyDate)` : retourne le mois de MyDate.
- `Year(MyDate)` : retourne l'année de MyDate.
- `Weekday(MyDate)` : retourne le numéro du jour de la semaine de MyDate.
- `Hour(MyTime)` : retourne les heures de MyTime.
- `Minute(MyTime)` : retourne les minutes de MyTime.
- `Second(MyTime)` : retourne les secondes de MyTime.

Heure et date du système :

- `Date()` : retourne la date actuelle.
- `Time()` : retourne l'heure actuelle.
- `Now()` : retourne le point présent dans le temps (la date et l'heure combinées dans une seule valeur).

ENSAI École nationale de la statistique et de l'analyse de l'information

Formatage des dates

```
1 Sub Main
2     Dim myDate as Date
3     myDate = "01/06/98"
4     TestStr = Format(myDate, "mm-dd-yyyy") ' 01-06-1998
5     MsgBox TestStr
6 End Sub
```

Affichage des messages

```
1 MsgBox "This is a piece of information!"
2 MsgBox "This is a piece of information!", MB_OK
```

```
1 MsgBox "Do you want to continue?",  292
2 MsgBox "Do you want to continue?",  _
3    MB_YESNO + MB_DEFBUTTON2 + MB_ICONQUESTION
```

```
1 MsgBox "Do you want to continue?",  292, "Box Title"
```

Affichage des messages

- 0, MB_OK : bouton OK ;
- 1, MB_OKCANCEL : boutons OK et Annuler ;
- 2, MB_ABORTRETRYIGNORE : boutons Abandonner, Réessayer et Ignorer ;
- 3, MB_YESNOCANCEL : boutons Oui, Non et Annuler ;
- 4, MB_YESNO : boutons Oui et Non ;
- 5, MB_RETRYCANCEL : boutons Réessayer et Annuler.

- 0, MB_DEFBUTTON1 : le premier bouton est sélectionné par défaut ;
- 256, MB_DEFBUTTON2 : le deuxième bouton est sélectionné par défaut ;
- 512, MB_DEFBUTTON3 : le troisième bouton est sélectionné par défaut.

- 16, MB_ICONSTOP : signe stop ;
- 32, MB_ICONQUESTION : point d'interrogation ;
- 48, MB_ICONEXCLAMATION : point d'exclamation ;
- 64, MB_ICONINFORMATION : icône Astuce.

ENSAI École nationale de la statistique et de l'analyse de l'information

Valeurs de retour

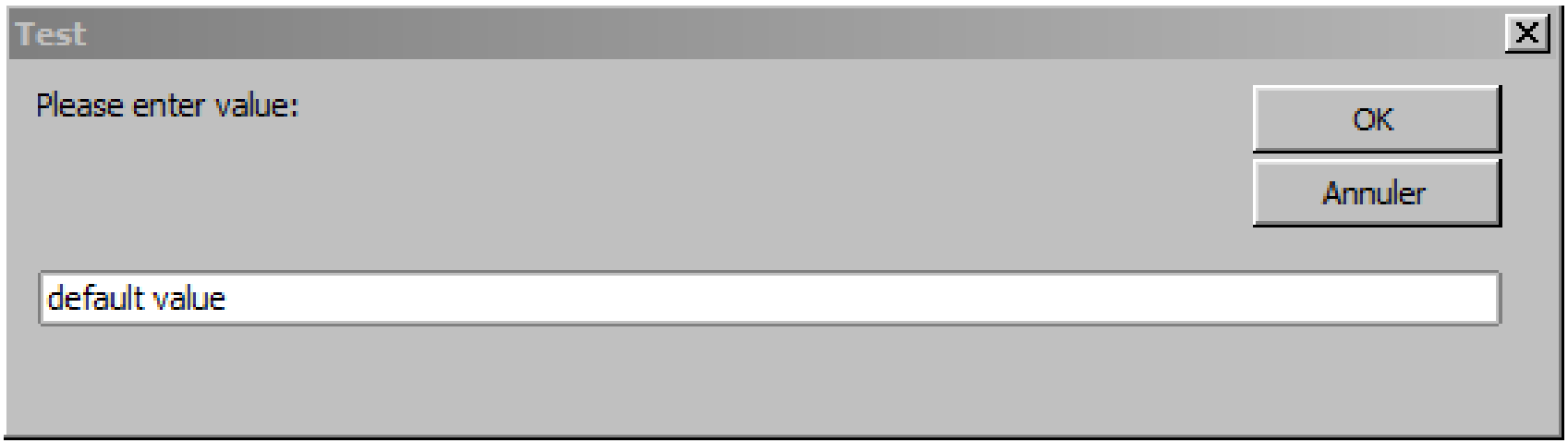
```
1 Dim iBox As Integer
2 iBox = MB_YESNO + MB_DEFBUTTON2 + MB_ICONQUESTION
3 If MsgBox ("Do you want to continue?",  iBox) = IDYES Then
4   ' Yes button pressed
5 Else
6   ' No button pressed
7 End If
```

- 1, IDOK : OK ;
- 2, IDCANCEL : Annuler ;
- 3, IDABORT : Abandonner ;
- 4, IDRETRY : Réessayer ;
- 5 : Ignorer ;
- 6, IDYES : Oui ;
- 7, IDNO : Non.

ENSAI École nationale de la statistique et de l'analyse de l'information

Zone de saisie

```
1 InputVal = InputBox("Please enter value:", _
2       "Test", "default value")
```



Programmation en LibreOffice Basic – seconde partie

Samuel TOUBON

2A 2017-2018

ENSAI
École nationale de la statistique et de l'analyse de l'information

Avertissement

- Cette partie n'est en aucun cas exhaustive. Elle vous présente quelques fonctions et quelques modèles pour que vous puissiez commencer rapidement.
- Vous pourrez vous référer à la documentation (cf. bibliographie), votre moteur de recherche favori, `stackoverflow.com` et toute source pertinente pour obtenir des informations plus complètes.
- Lorsque c'est nécessaire, les TP vous souffleront les parties de documentation les plus pertinentes.

ENSAI École nationale de la statistique et de l'analyse de l'information

Sommaire de la seconde partie

ENSAI École nationale de la statistique et de l'analyse de l'information

Accès au classeur

Classeur contenant la macro :

```
1 Dim oDoc As Object
2 oDoc = ThisComponent
```

Le **composant** actif (ce peut être un classeur ou non) :

```
1 Dim oDoc As Object
2 oDoc = CurrentComponent
```

Accès à la feuille de calcul

Par son numéro :

```
1 Dim Doc As Object
2 Dim Sheet As Object
3
4 Doc = ThisComponent
5 Sheet = Doc.Sheets (0)
```

Par son nom :

```
1 Dim Doc As Object
2 Dim Sheet As Object
3
4 Doc = ThisComponent
5 Sheet = Doc.Sheets.getByName("Sheet 1")
```

ENSAI École nationale de la statistique et de l'analyse de l'information

Accès à la feuille de calcul

Feuille active :

```
1 ThisComponent.getCurrentController.getActiveSheet
```

Parcours des feuilles :

```
1 Dim oSheet As Object
2 Dim eSheets As Object
3 eSheets = ThisComponent.getSheets.createEnumeration
4
5 Do While eSheets.hasMoreElements
6         oSheet = eSheets.nextElement()
7         ' here you can work your sheet
8         MsgBox "Next sheet name is " & oSheet.getName & "."
9 Loop
```

Création d'une feuille de calcul

```
1  Dim Doc As Object
2  Dim Sheet As Object
3
4  Doc = StarDesktop.CurrentComponent
5  Sheet = Doc.Sheets(0)
6
7  If Doc.Sheets.hasByName("MySheet") Then
8     Sheet = Doc.Sheets.getByName("MySheet")
9  Else
10    Sheet = Doc.createInstance("com.sun.star.sheet.Spreadsheet")
11    Doc.Sheets.insertByName("MySheet", Sheet)
12 End If
```

ENSAI École nationale de la statistique et de l'analyse de l'information

Création et suppression d'une feuille de calcul

On peut aussi utiliser `insertNewByName` dont le deuxième argument est la position :

```
1 Doc.Sheets.insertNewByName("MySheet", 2)
```

La suppression peut se faire ainsi :

```
1 Doc.Sheets.removeByName("MySheet")
```

Sélection de lignes ou colonnes

```
1  Dim Doc As Object
2  Dim Sheet As Object
3  Dim FirstRow As Object
4  Dim FirstCol As Object
5
6  Doc = StarDesktop.CurrentComponent
7  Sheet = Doc.Sheets(0)
8
9  FirstCol = Sheet.Columns(0)
10 FirstRow = Sheet.Rows(0)
```

Insertion de lignes ou colonnes

```
1 Dim Doc As Object
2 Dim Sheet As Object
3 Dim NewColumn As Object
4
5 Doc = StarDesktop.CurrentComponent
6 Sheet = Doc.Sheets(0)
7
8 Sheet.Columns.insertByIndex(3, 1) ' insert 1 colonne en 4e pos.
9 Sheet.Columns.removeByIndex(5, 1) ' efface la colonne 6
```

ENSAI École nationale de la statistique et de l'analyse de l'information

Sélection de cellule

```
1 Dim Doc As Object
2 Dim Sheet As Object
3 Dim Cell As Object
4
5 Doc = StarDesktop.CurrentComponent
6 Sheet = Doc.Sheets(0)
7
8 Cell = Sheet.getCellByPosition(0, 0)
9 Cell.String = "Test"
```

Ou bien :

```
1 Cell = Sheet.getCellRangeByName("A1")
```

Contenu d'une cellule

```
1 Cell = Sheet.getCellByPosition(0, 0)
2 Cell.Value = 100
3
4 Cell = Sheet.getCellByPosition(0, 1)
5 Cell.String = "Test"
6
7 Cell = Sheet.getCellByPosition(0, 2)
8 Cell.Formula = "=A1"
```

Sélection de plage de cellules

```
1 oRange = oSheet.getCellRangeByName( "B2:C3" )
```

Ou bien :

```
1 oRange = oSheet.getCellRangeByPosition(1,1,2,2)
```

L'ordre est left, top, right, bottom.

Opérations sur les plages *

```
1 Sheet = Doc.Sheets.getByName("Sheet 1")
2 CellRange = Sheet.getCellRangeByName("A1:C3")
3 MsgBox CellRange.computeFunction(com.sun.star.
4     sheet.GeneralFunction.AVERAGE)
```

- SUM : somme de toutes les valeurs numériques.
- COUNT : nombre total de valeurs (y compris les valeurs non numériques).
- COUNTNUMS : nombre total de valeurs numériques.
- AVERAGE : moyenne de toutes les valeurs numériques.
- MAX : valeur numérique la plus élevée.
- MIN : valeur numérique la plus petite.
- PRODUCT : produit de toutes les valeurs numériques.
- STDEV : écart-type.
- VAR : variance.
- STDEVP : écart-type calculé sur la base de la population totale.
- VARP : variance calculée sur la base de la population totale.

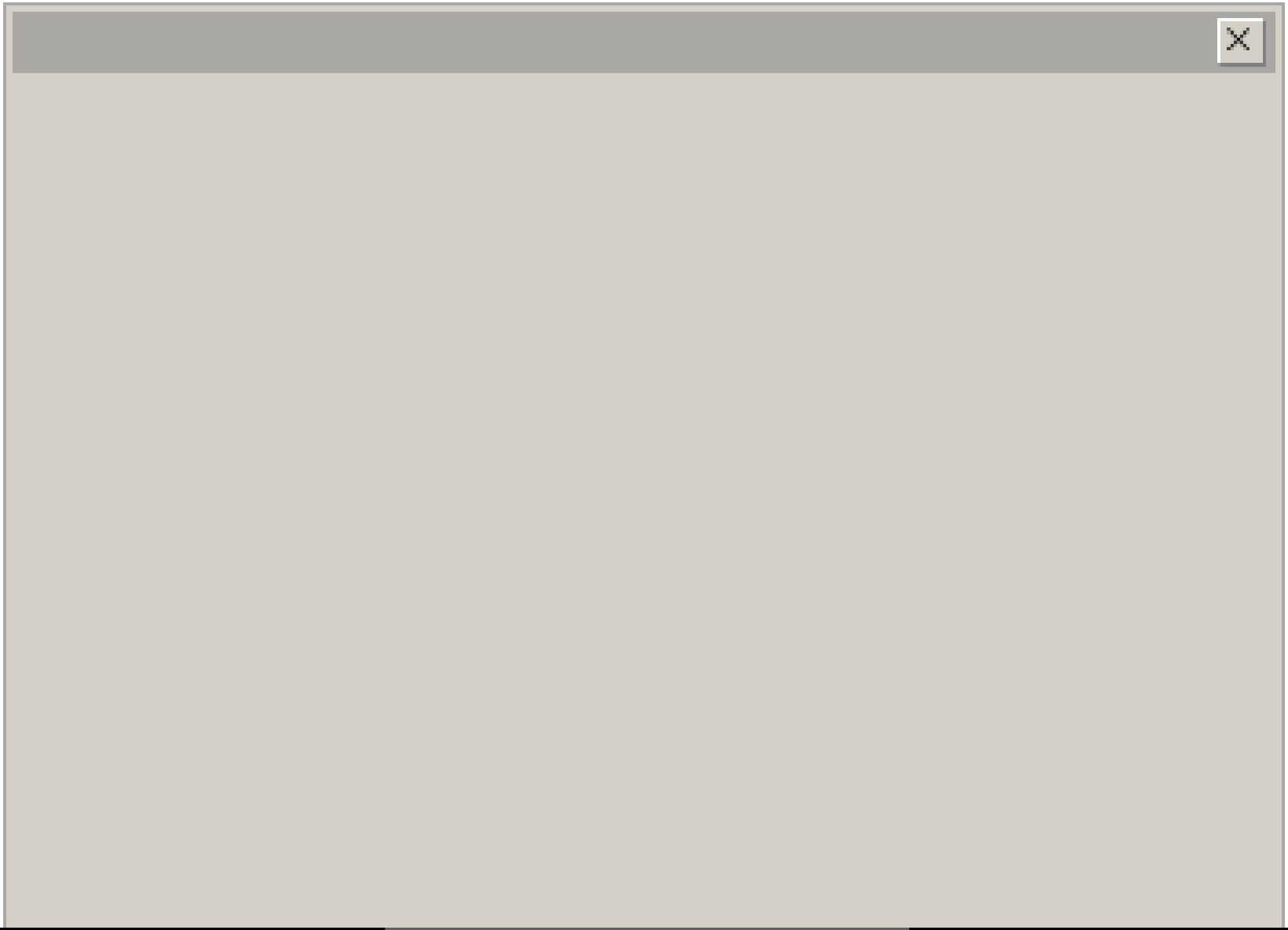
Nous verrons au TP4 comment utiliser n'importe quelle fonction de LibreOffice pour faire des calculs sur les plages, et pas seulement les précédentes.

ENSAI École nationale de la statistique et de l'analyse de l'information

Sommaire

ENSAI École nationale de la statistique et de l'analyse de l'information

Assistant graphique de création



Ouverture d'une boîte de dialogue créée avec l'assistant

```
1  Dim Dlg As Object
2
3  DialogLibraries.LoadLibrary("Standard")
4  Dlg = CreateUnoDialog(DialogLibraries.Standard.DlgDef)
5
6  'Affiche la boite de dialogue
7  Dlg.Execute()
8
9  'Libere les ressources
10 Dlg.dispose()
```

ENSAI École nationale de la statistique et de l'analyse de l'information

Éléments de contrôle

- boutons submit
- images
- cases à cocher
- bouton radio
- zone de texte
- liste
- combobox
- barre de progression
- champ de date / d'heure
- champ numérique / de devises
- sélecteur de fichier
- ...

Accès aux éléments de contrôle

```
1 Dim Ctl As Object
2
3 Ctl = Dlg.getControl("MyButton")
4 Ctl.Label = "New Label"
```

Le nom des éléments de contrôle est sensible à la casse.

Accès aux éléments de contrôle

De nombreuses propriétés des éléments de contrôle (nom, titre, hauteur, position, ...) se trouvent dans son `Model` :

```
1  Controls = Dlg.Controls
2
3  I = 0
4  A = ""
5
6  For Each cControl In Controls
7          I = I + 1
8          A = A & cControl.getModel().Name
9          ' To get back the name of cControl.
10 Next cControl
```

Et bien plus en TP...

- Diagrammes
- Utilisation de formules LO dans LOB
- Structures de données complexes
- Connexion aux bases de données
- Et plein d'autres choses pour impressionner votre chef

Autour de thèmes « amusants » :

- Le magasin de bédés de Stuart
- Les étoiles de notre univers
- Les conflits armés dans l'Histoire
- Les nombres premiers sexys (Si, si, ça prend un « s » au pluriel.)
- Les voitures électriques

ENSAI École nationale de la statistique et de l'analyse de l'information

Bibliographie

- Guide de programmation de OpenOffice.org BASIC : `https://wiki.openoffice.org/wiki/FR/Documentation/BASIC_Guide`
- Le même en anglais (plus complet, moins d'erreurs) : `https://wiki.openoffice.org/wiki/Documentation/BASIC_Guide`
- Mémento sur la manipulation des feuilles de calcul : `https://wiki.openoffice.org/wiki/Spreadsheet_common`

ENSAI École nationale de la statistique et de l'analyse de l'information