

# Excel

## VBA programming

### Visual Basic for Applications

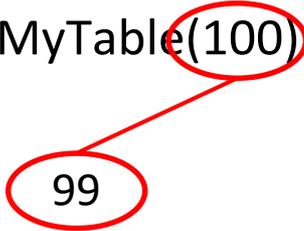
Hervé Hocquard

<http://www.labri.fr/perso/hocquard>

# ARRAYS

- Declaration
  - `Dim MyTable(1 To 100) As Integer` or `Dim MyTable(100) As Integer...`
  - Index starts at 0 by default:
    - Option Base 1
- Multidimensional arrays
  - `Dim MyTable(1 To 10,1 To 10) As Integer`
- Assignment
  - `MyTable(2)=17` 'one dimension
  - `MyTable(3,4) = 125` 'two dimensions

99



- Dynamic arrays
- Creation
  - `Dim MyTable() As Integer`
- Resizing
  - `ReDim MyTable(NumberElements)`
- Resize keeping the data already present
  - `ReDim Preserve MyTable(NumberElements)`

- **Structure to display the content:**

```
Dim month As Variant
Dim m As Variant
month = Array("January", "March", "August", "December")
For Each m In month
    MsgBox m
Next m
```

- **Or...**

```
Dim month As Variant
Dim i As Integer
month = Array("January", "March", "August", "December")
For i = 0 To 3
    MsgBox month(i)
Next i
```

- Functions on arrays:
  - **Lbound** : smallest index of the array
  - **(Lbound,i)** : smallest dimension index **i** of the array
  - **Ubound** : larger index
  - **(Ubound,i)** : largest dimension index **i** of the array
  - **Array(...)** : returns an array (must be assigned to a Variant)
  - **Erase** : clears the array from memory

- Examples of array functions

```
Dim month As Variant, i As Integer  
month = Array("January", "March", "August", "December")
```

```
For i = LBound(month) To UBound(month)
```

```
    MsgBox month(i)
```

```
Next i
```

```
Erase month
```

Give the lower bound

Give the upper bound

# STRINGS

- Concatenation: & ("Hervé" & " Hocquard ")
- Periodic construction:
  - **String (20,"x")**: repeats the character 'x' 20 times
  - **Space(10)**: generates a sequence of 10 spaces
- Burst: **Split (chain,separator)**
  - s = Split ("c: \ windows \ system32 \ driver.dll", "\")
  - s must be of type **Variant**

```
Sub eclatetout()
```

```
Dim Tableau() As String
```

```
Dim i As Integer
```

```
'découpe la chaine en fonction des espaces " "
```

```
'le résultat de la fonction Split est stocké dans un tableau
```

```
Tableau = Split("c:\windows\system32\driver.dll", "\")
```

```
'boucle sur le tableau pour visualiser le résultat
```

```
For i = 0 To UBound(Tableau)
```

```
    'Le résultat s'affiche dans la fenêtre d'exécution de l'éditeur de macros
```

```
    Debug.Print Tableau(i)
```

```
Next i
```

```
End Sub
```

- Length: `Len(chain)`
- Positioning: `InStr(string,character)`
  - `pos = InStr("The weather is nice","h")` returns **9**
- Lowercase, uppercase:
  - `LCase("Hello")` returns "hello "
  - `UCase("Hello")` return "HELLO"
- Character selection: `Mid, Left, Right`
  - `Left("Hocquard",3)` returns "Hoc"
  - `Right ("Hocquard",3)` returns "ard"
  - `Mid("Hocquard",5,2)` returns "ua"

- Function **Format(...)** returns a character string formatted according to the parameters
- Same operation as in Excel
- `Format (12121.13,"## '###. 00")` returns `12'121.13`
- `Format ("Hello","<")` ⇔ `UCase("Hello")`
- `Format (Date,"yy/mmmm/dd ")`
  - The `Date` function returns the current date. This must be formatted before display in a dialog box, otherwise it will be displayed in the form specified in the regional options (`dd/mm/yy`)
- Other functions here:  
<https://silkyroad.developpez.com/VBA/ManipulerChainesCaracteres/>

Structured type

Simple or structured fields

# USER-DEFINED DATA TYPES

- VBA lets you create custom, or user-defined, data types. A user-defined data type can ease your work with some types of data.
- Unlike arrays, this structured type allows you to group data of different types.
- Example: a book is identified by a code, a title, one or more authors, a publisher and possibly the date of publication.
- Book is a User-Defined Data Type variable; each of these five data is a field that can be simple or structured.

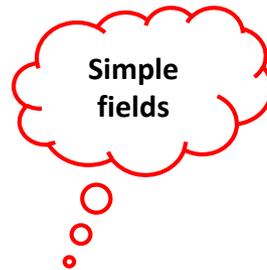
- The User-Defined Data Types are declared in VB with the word **Type**.

- Syntax:

```
Type RecordName  
  Field1 As type1  
  Field2 As type2  
  ...  
End Type
```

- Example:

```
Type book  
code As Integer  
title As String * 40  
author As String * 50  
editor As String * 50  
date As Madate  
End Type
```



```
Type Madate  
day As Integer  
month As Integer  
year As Integer  
End Type
```



- Example:

```
Type book
code As Integer
title As String * 40
author As String * 50
editor As String * 50
date As Madate
End Type
```

```
Type Madate
day As Integer
month As Integer
year As Integer
End Type
```

- To access a control:

```
Dim livre As book
livre.author = "Durand"
livre.date.year = 1980
```

'we see here that we could replace livre by an array in the type book... Dim livre(1 To 10000) as book...

livre(9).author = "Durand" if it is the ninth book on the list ...

...

A student is defined by his name, first name, date of birth and his note:

```
Private Type Student
name As String * 40
firstname As String * 40
dateofbirth As Date
note As Double
End Type
```

A class can contain at most 30 students:

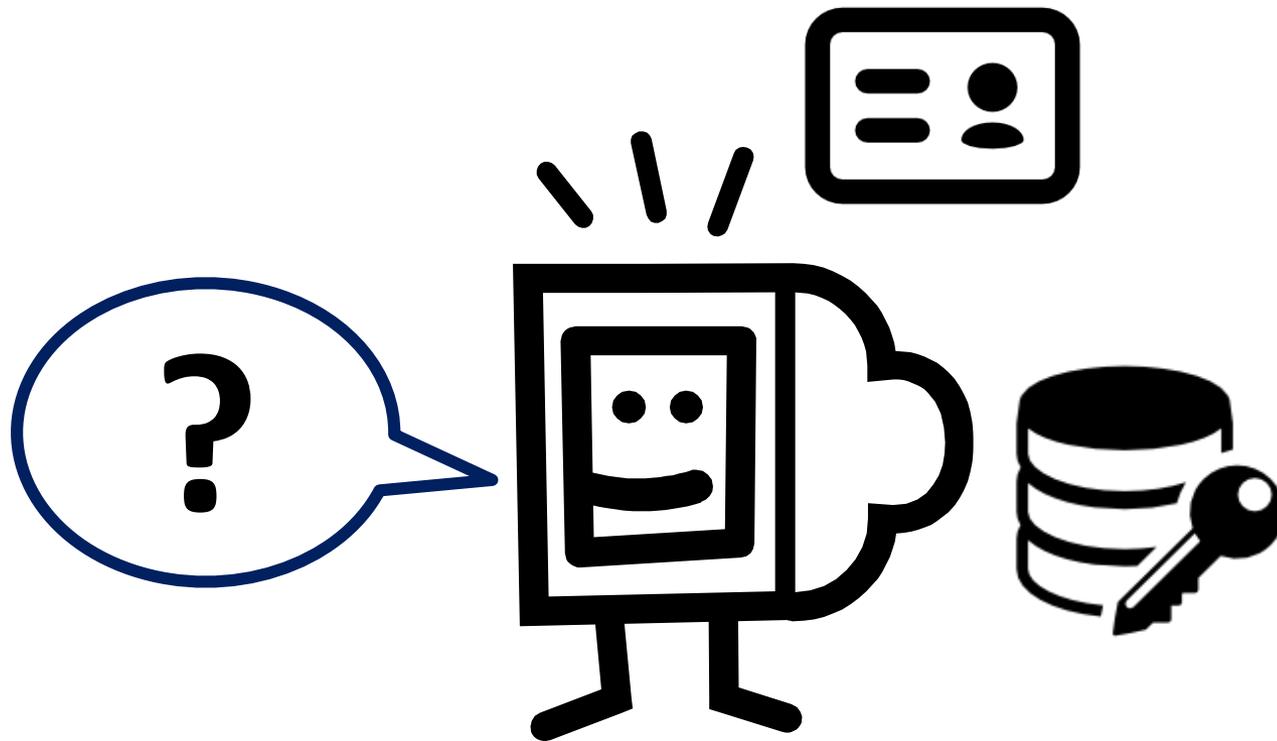
```
Const NbMax = 30 'for the maximum number of students
```

```
Private Type Class
list(NbMax) As Student 'the list is an array of students
nbr As Integer 'the real number of students
End Type
```

We then declare the class of students:

```
Dim c As Class
```

- The previous example will be completed in the next course on graphical interfaces ...
- How to define a matrix?
- Create a program that displays the number of rows and columns of a matrix entered on Sheet1 of the Workbook.



To be continued with Dialog Boxes and UserForms...

# Thank you

Herve Hocquard( [hocquard@labri.fr](mailto:hocquard@labri.fr) )

<http://www.labri.fr/perso/hocquard/Teaching.html>

