

Understanding the VBA/Excel environment

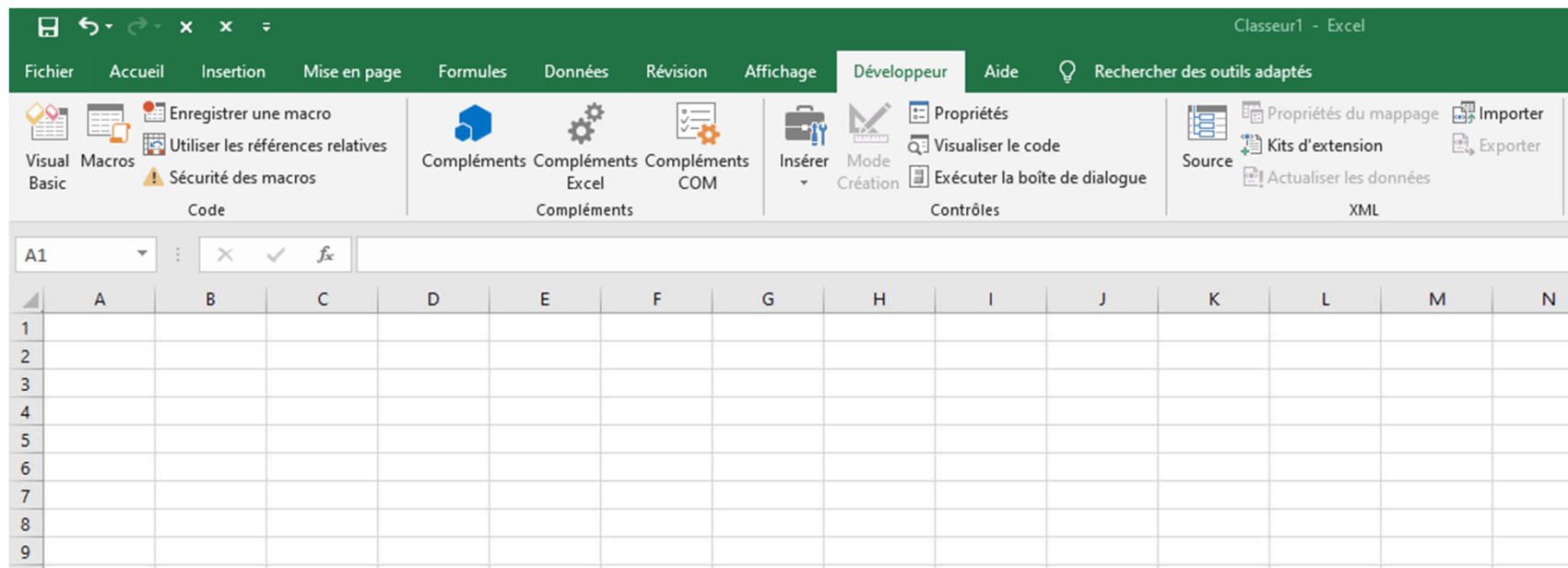
Installing VBA
& Creating your first programme



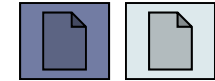
Adding VBA Interface in Excel



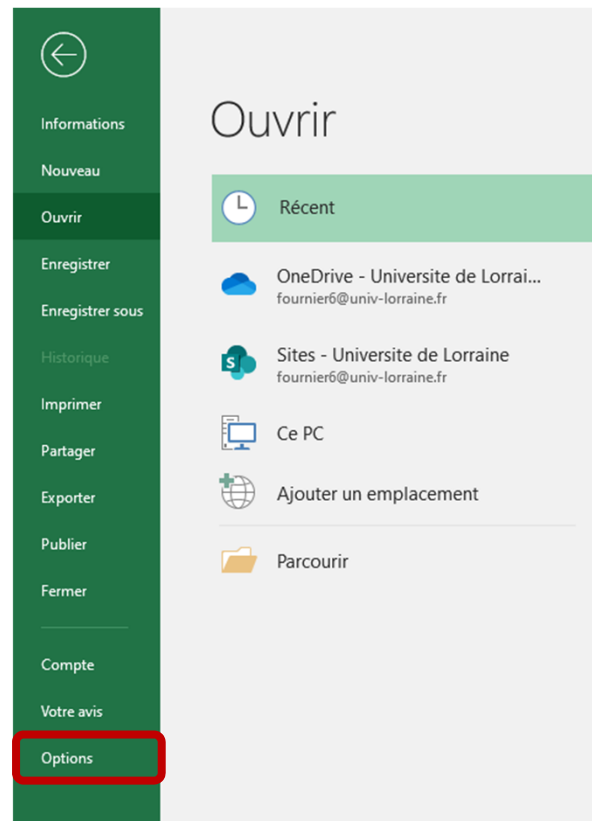
- ▶ VBA is **not** installed in the default configuration
- ▶ You have to add it to Excel to display the « *Développeur* » menu



Adding VBA Interface in Excel

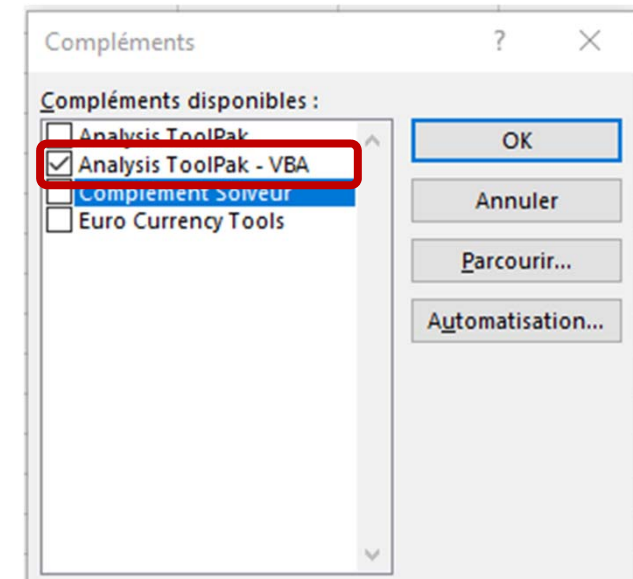
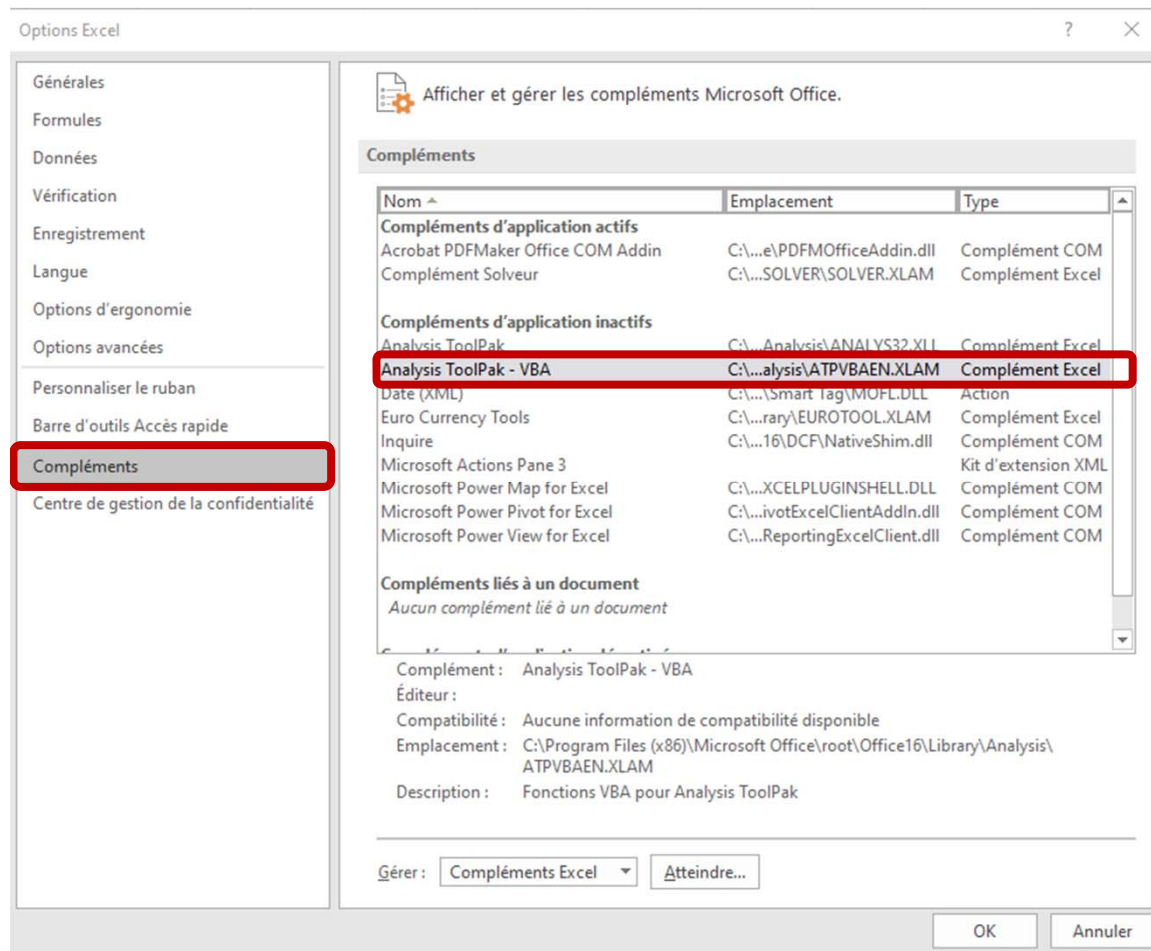


First: Display the Excel « Options »



Modify Excel Options

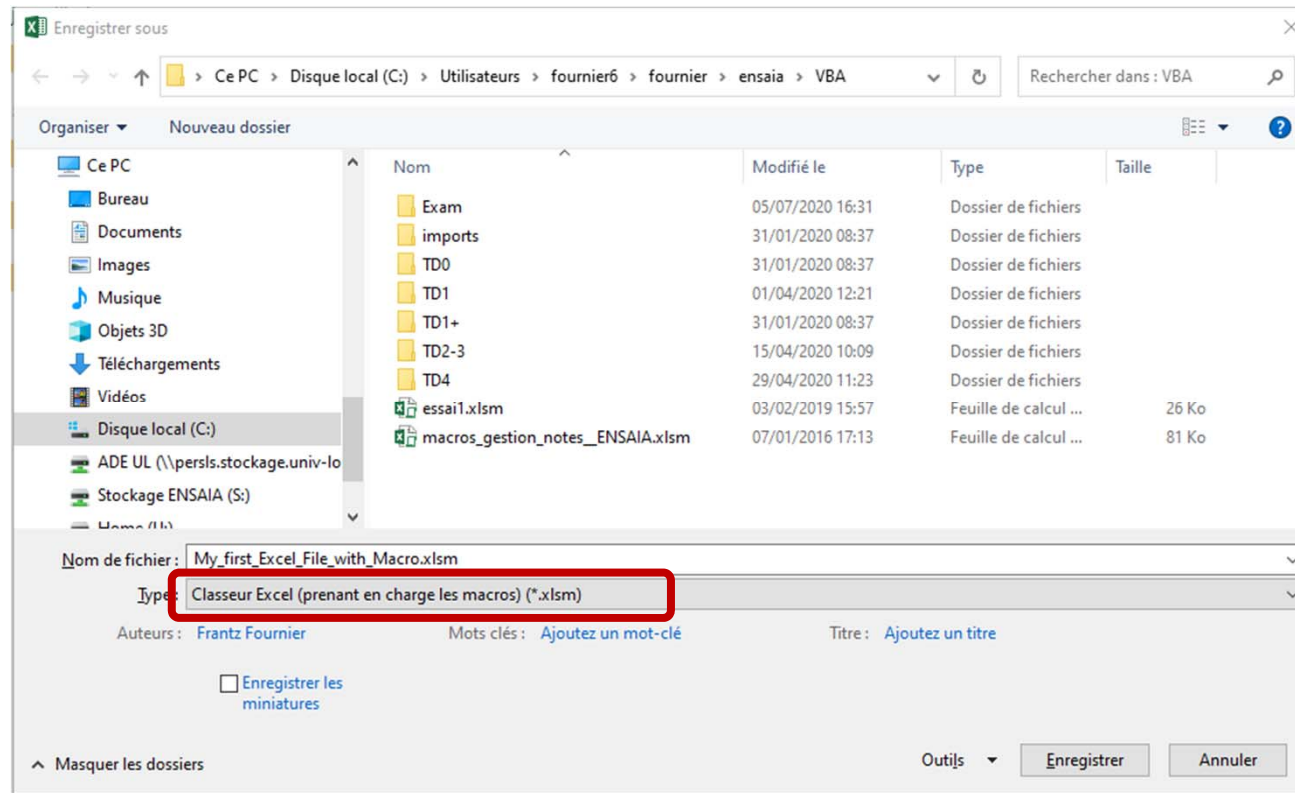
Then: Select the « Compléments » option
And add the VBA Toolpack



Create a « VBA compatible » Excel file

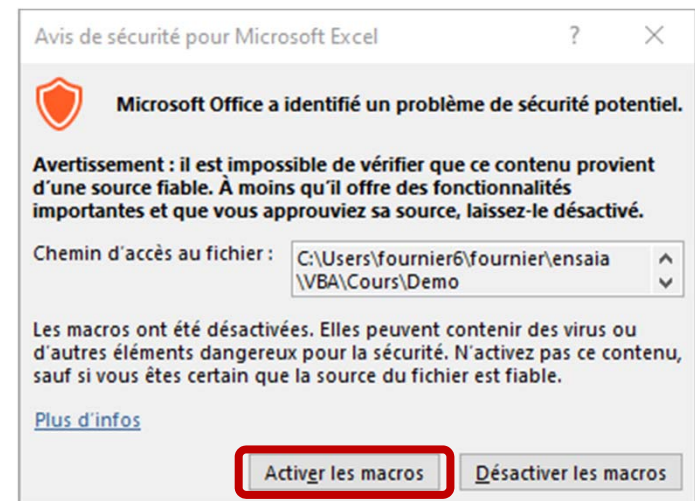
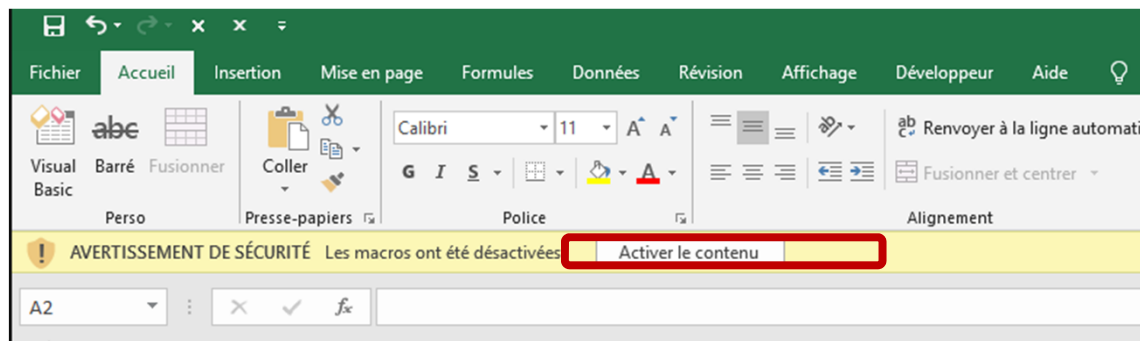
In order to use VBA in an Excel file, you must save it as an **Excel file with Macro**, i.e. with « .xlsm » extension.

Use « save as... » then select the right file type and give it a suitable file name.



Use an Excel file containing « macro »

When opening an .xlsm file, you must activate the use of macros.

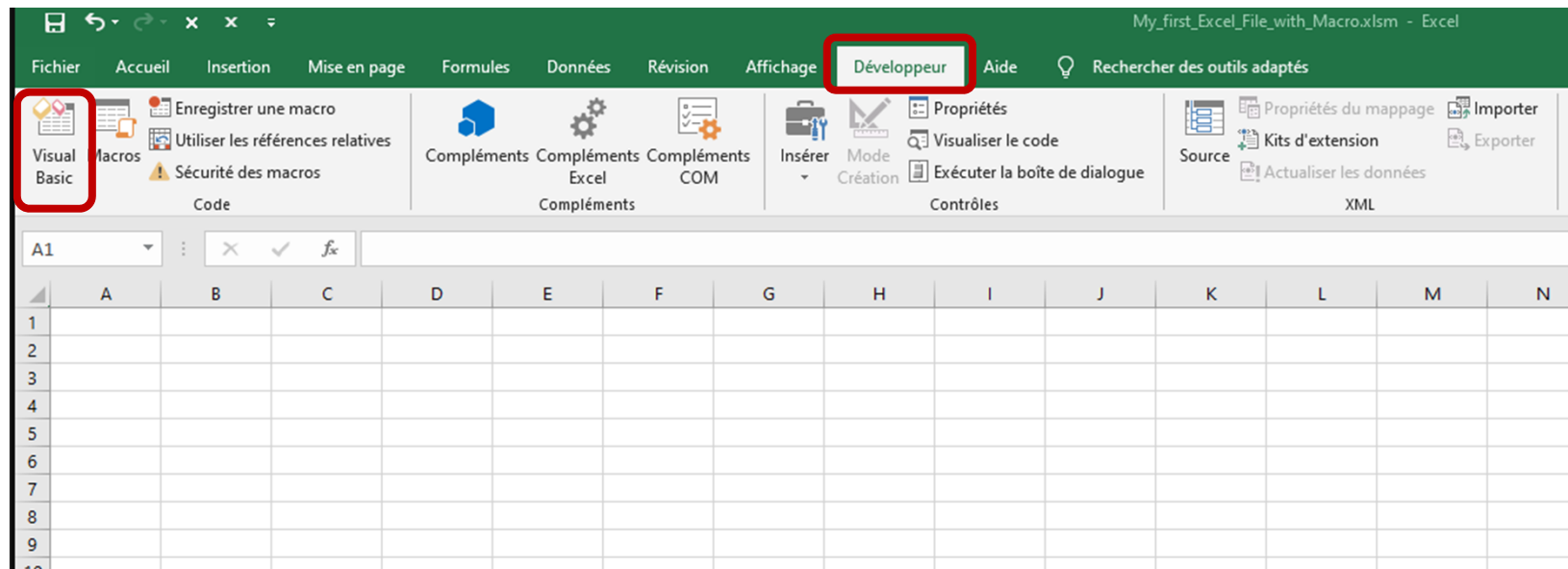


Be careful: VBA's Macro does many automated actions. They can either help you to proceed your data but it can also disturb or delete your data. You **must** trust the file before activating macros.



Launching VBA Editor-Debugger

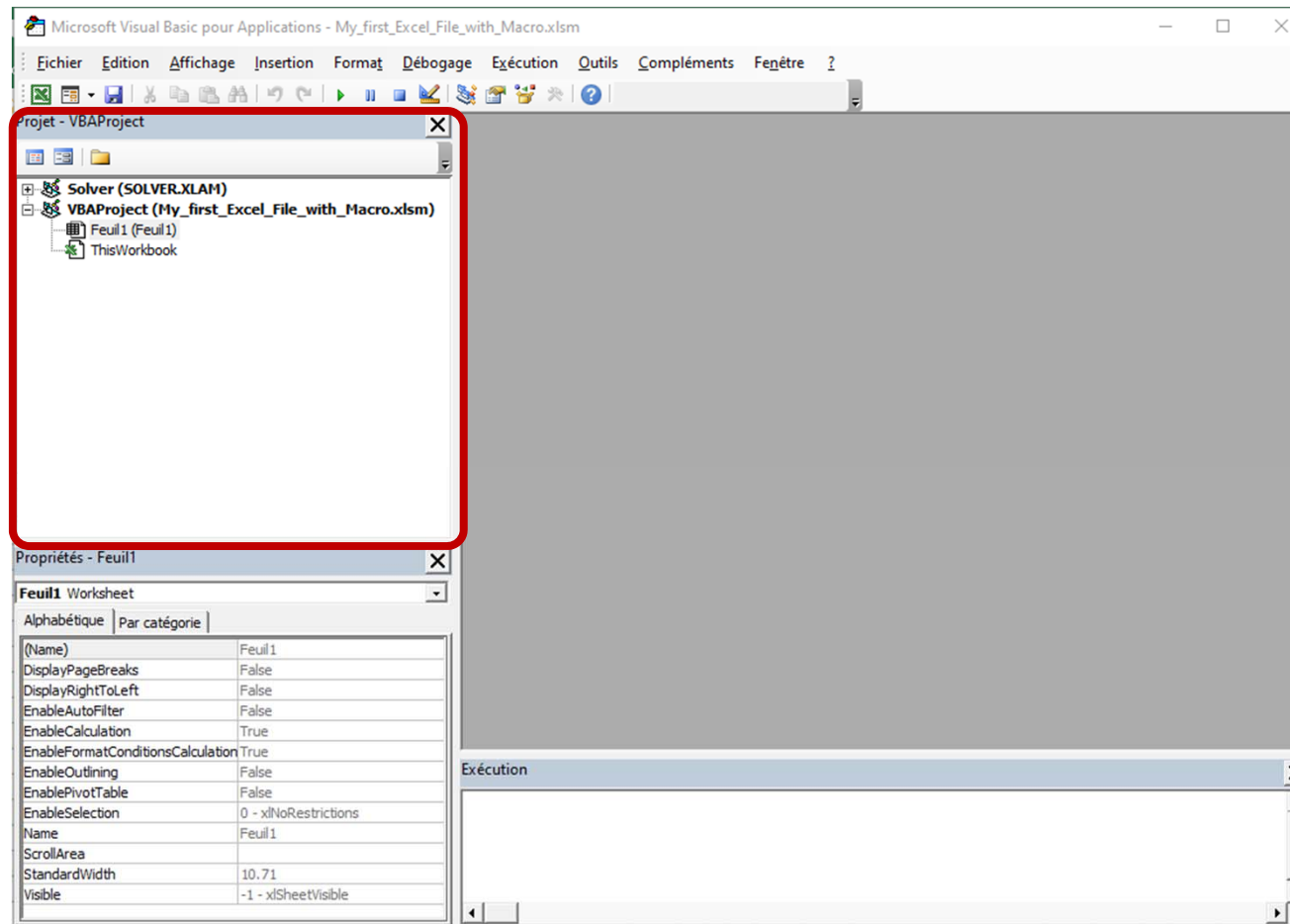
From the « Developpeur » menu, select « Visual Basic »



Keyword shortcut: **Alt+F11**



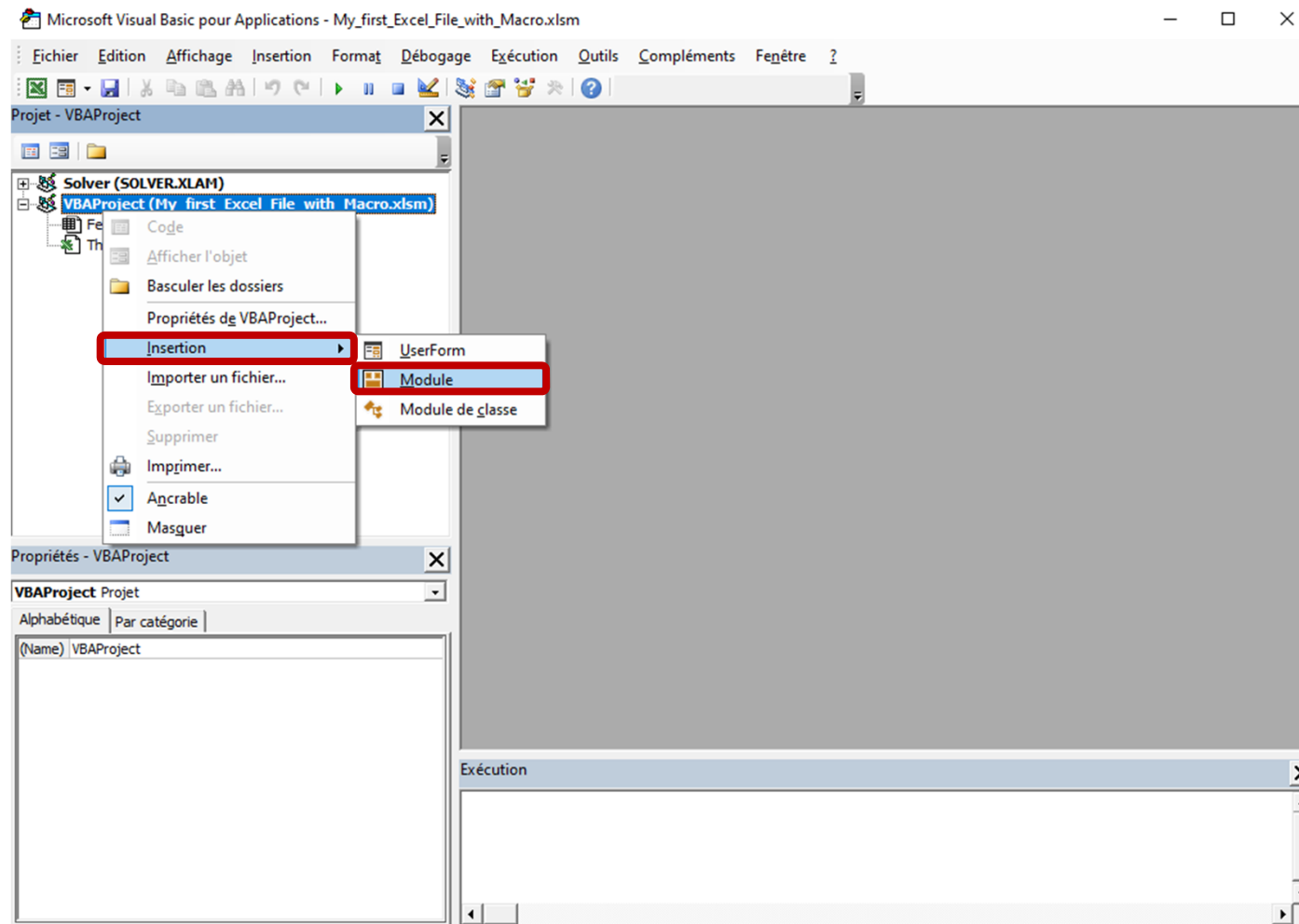
Presentation of the VBA Editor-Debugger



Your « project » contains by default a *Sheet* and a *ThisWorkBook* (for « private » programme only)



VBA Editor-Debugger



Right click on the *VBAProject* name to insert a *Module* to create your first programmes. Programmes will then be « public » and therefore usable from all part of your Excel file.



Module organisation

Write your programmes in the Editor windows.

It is automatically divided into 2 parts :

- *Declaration* section to declare shared variables and constant
- *Programming* section to programme functions and subs

The screenshot shows the VBA Editor window with the 'Déclarations' tab selected. The code is divided into two sections: 'Shared declaring section' and 'Programming sections'.

Shared declaring section:

- `Option Explicit` (labeled: Explicite variables declaration)
- `Option Base 1` (labeled: Matrix index default numbering)
- `Public Const R As Double = 8.314` (labeled: Global constante value R)
- `Public Diam as Double` (labeled: Global variable Diam)

Programming sections:

- `Function RectangleArea(Width As Double, Height As Double) As Double`
`RectangleArea = Width * Height`
`End Function` (labeled: Code of a new VBA Function)
- `Sub Hello()`
`MsgBox "Bonjour"`
`End Sub` (labeled: Code of a new VBA Sub)

Note: Any text following the ' symbol is a comment and is not read by VBA
The symbol " _ " is used for line continuation

