

Introduction to VBA for Excel-Tutorial 1

What is a program? A set of **instructions** issued to a computer to perform very **specific tasks** in a particular **sequence of operation**. Take note of the three key words in the above definition:

- 1- **Instructions** are written in a language that the computer can understand or what is so called programming language syntax.
- 2- **Specific tasks** are the purpose of composing a computer program. Therefore, define the objective before you start getting frustrated; it takes time to write a computer program that runs correctly. Once you define your goal, you can find the way to do it.
- 3- **Sequence of operations** is the logic or the algorithm. It is the link between the user's inputs and the program outputs.

Word of advice: read the above definition very carefully and always remember “organization is the key to success.”

James Hiestand said in his book on Numerical Methods with VBA Programming: “*Writing a program is like talking to a young child or an animal.*”

Let me present you with this question: **what is the first step you take in writing an English paper?**

If you answered **PLAN**, you are going to be a good programmer, if not, now you know the answer. In programming, we use a Flowchart to communicate the plan. Before we dig into flowcharts; we need to learn about program components.

Components of a program:

Others will define parts of a computer program as Input/output, Arithmetic, Variable type, Decisions, Loops, Subprograms and Function, and Arrays. There is nothing wrong of course about the definition, it is very comprehensive and conclusive. I define it a little different, or I should say a little simpler. The most important parts of a program, in my opinion, are the inputs, the logic and the output.

Inputs: information passed by the user to the program via a computer input device (i.e. keyboard, mouse, etc). The inputs resemble all the known quantities in the engineering problem at hand.

Logic: how the computer is going to calculate, analyze, and/or process the input to produce the results of the engineering problem.

Outputs: the results of the processed input data by the logic.

Simple enough! Identify the inputs, understand how to process it (logically), and report the results (outputs).

Flowchart:

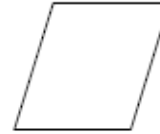
A flowchart is a graphical outline of the program emphasizing the logic or sequence of operation. If you spend time developing a complete flowchart, writing the program becomes an easier task.

The flowchart is independent of the programming language, it is meant to highlight the logic not the syntax. Keep the flowchart separate from the language syntax. A list of basic flowchart symbols is presented below; these symbols are connected together by arrows.

An oval is used for starting and ending a program



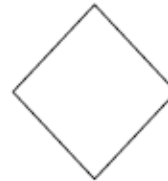
A rhombus is used for input and output.



A rectangle is used for calculation.



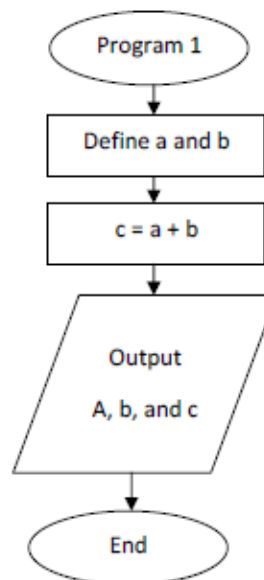
A diamond is used for decisions.



A hexagon is used to delineate loops.



Example: compose a flowchart to add two numbers.



The basics of VBA

A summary of VBA components is presented first. This is an important section to read carefully.

Code: you perform actions in VBA by executing VBA code. The code is stored in a VBA module.

Module: modules are stored in the Excel workbook. A VBA module consists of procedures.

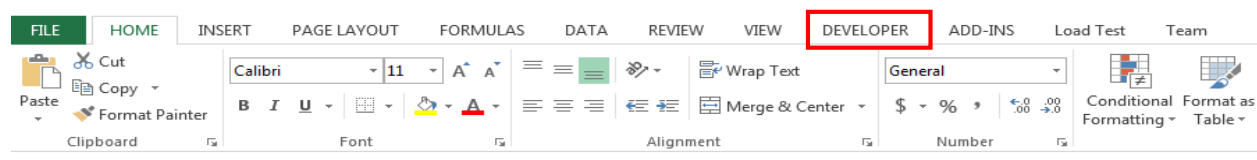
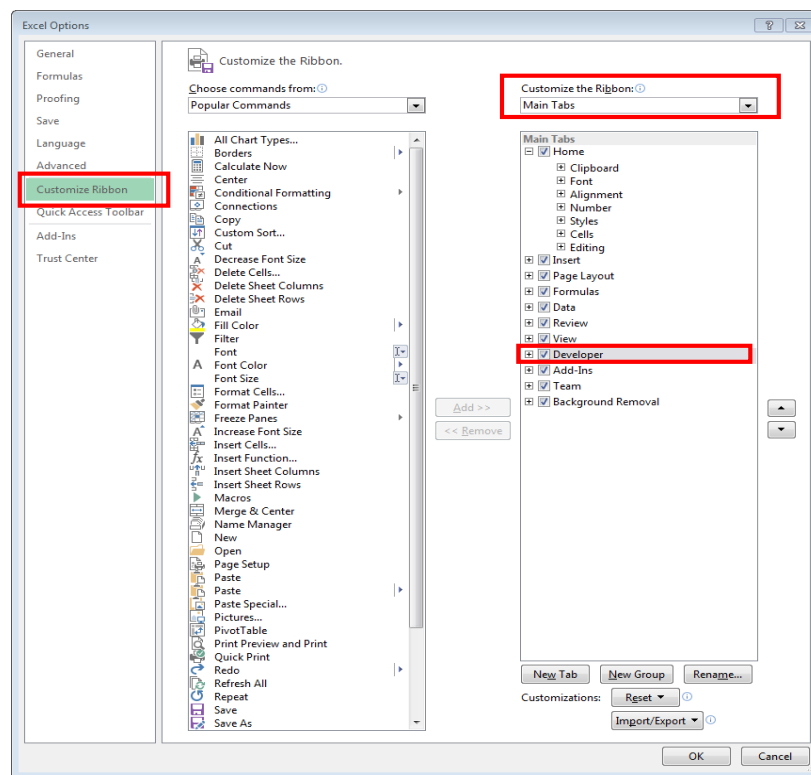
Procedure: A procedure is basically a unit of computer code that performs some actions.

- **Sub:** a Sub procedure consists of a series of statements, which will be executed to process the inputs and generate outputs.
- **Function:** a function procedure returns a single value or possibly an array.

Objects: VBA manipulates objects contained in its host application. Objects include a workbook, worksheet, range on worksheet, a chart, and/or drawn rectangle.

Developer Tab

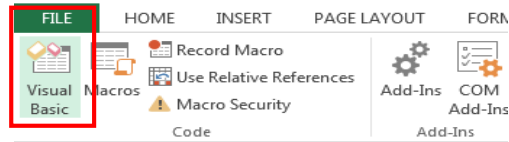
Depending on the installation and machine configuration, Excel may not display the Developer tab by default. To turn-on this tab, click File/Options/Customize Ribbon and then under Customize the Ribbon and Under Main Tabs, select the Developer check box.



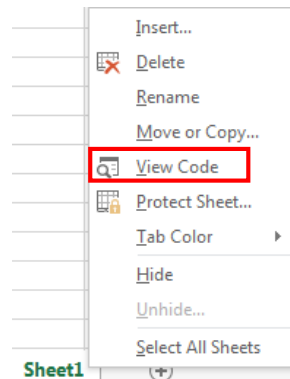
To access the Visual Basic Editor (VBE):

- Press Alt+F11

- Choose the Developer tab and select Visual Basic from the Code submenu.

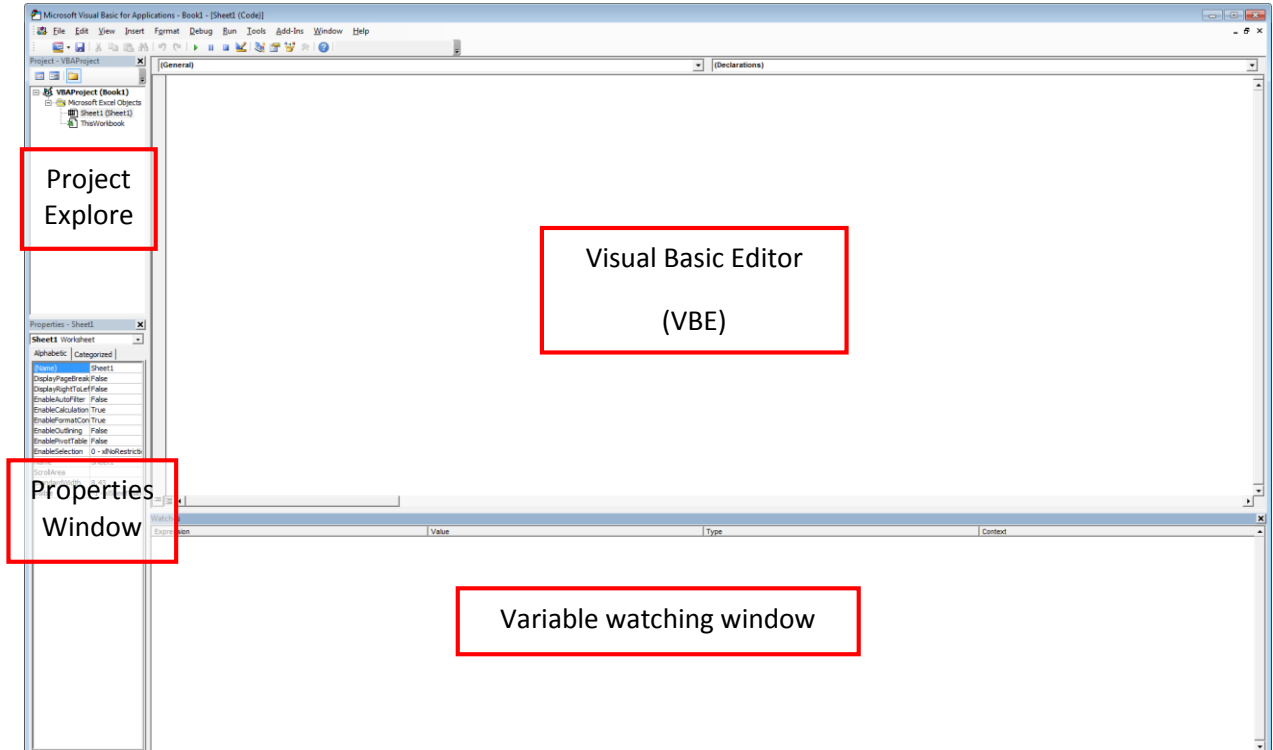


- Right Click on the Sheet1 tab (on the bottom of Excel application) and select View Code. Of course, you can perform the same operation on any of the Sheet tabs (i.e. Sheet2, etc.).

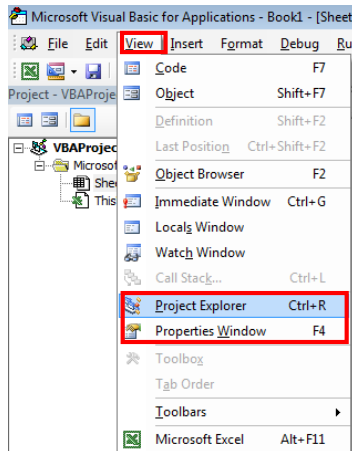


Visual Basic Environment Window

Once you perform one of the previous operations, Excel will open the following window.

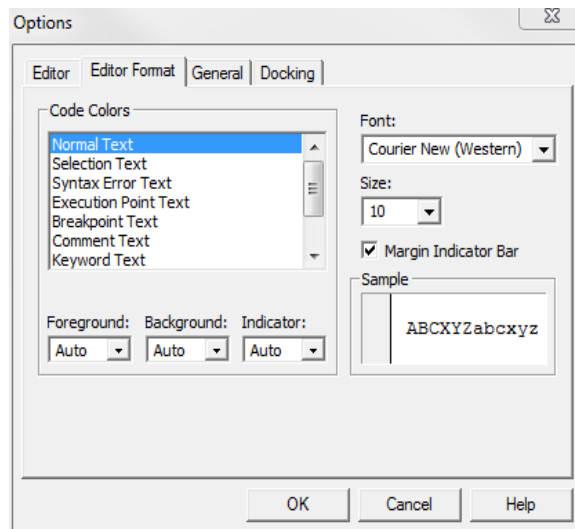
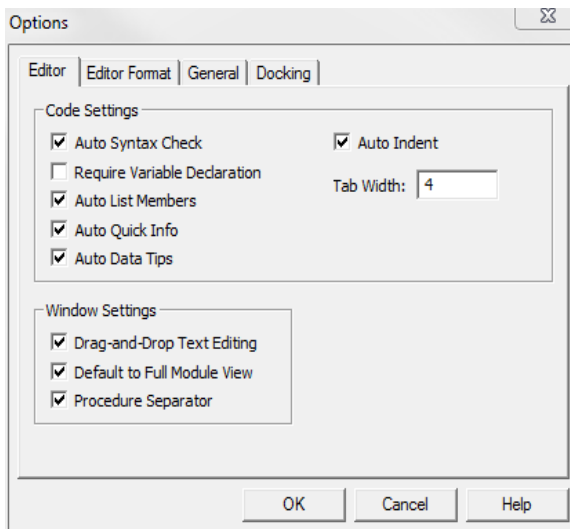


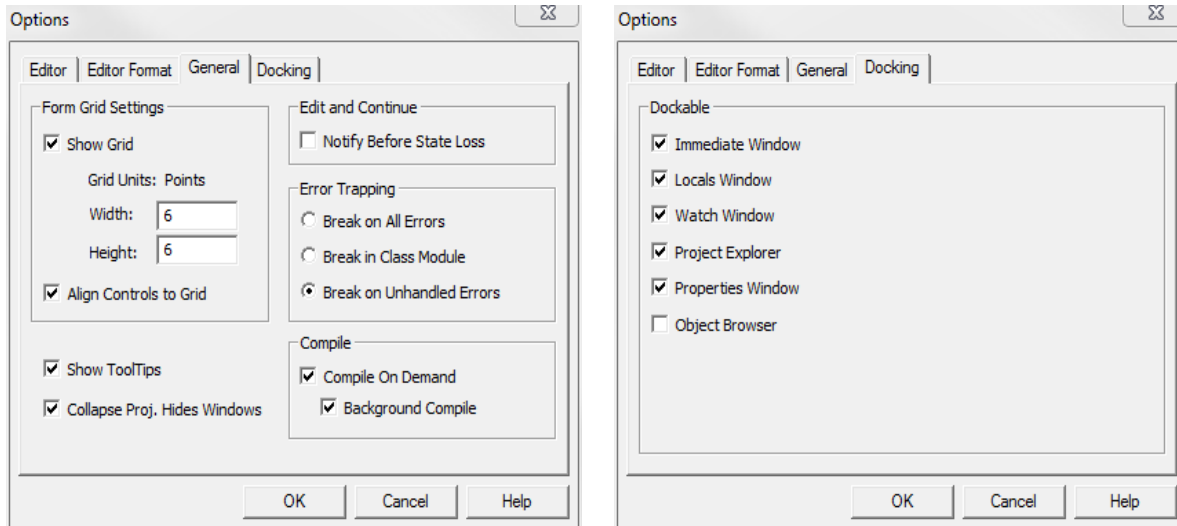
If either the Project Explorer or Properties Window is not shown, you can reopen by selecting View/Project Explorer (or press Ctrl+R) or View/Properties Window (or press F4).



Customizing the VBE environment:

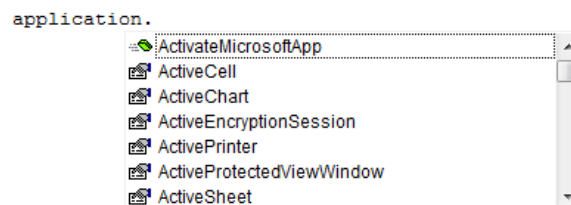
Before we start writing programs in the VBA editor, we need to learn how to customize the editor environment. To access the customization window, select Tools/Options/Editor tab. Please match selections on your editor with the selections shown below. Remember, every machine (i.e. computer) you are going to use must be changed to the same settings. Most of the settings are default anyway, but it is worth checking.





Focusing on the Editor tab options:

- 1- **Auto Syntax Check Option:** I strongly recommend leaving this option **ON**. If the Auto Syntax Check option is checked ON, the VBE pops up a dialog box when it discovers a syntax error while you are entering VBA code. It helps you with debugging while composing your code.
- 2- **Require Variable Declaration Option:** If this option is set, VBE insert “Option Explicit” at the beginning of each new VBA module. Thus, you must explicitly define each variable that is used within your program.
- 3- **Auto List Members Option:** If this option is set, VBE provides some help while you are entering VBA code by displaying a list of member items for the object. You can consider this as auto-fill; the editor suggests methods and properties.



- 4- **Auto Quick Info Option:** If this option is set, VBE provides information about the arguments available for functions, properties, and methods while you are inputting the command.

```
cells (
  _Default([RowIndex], [ColumnIndex])
```

- 5- **Auto Data Tips Option:** Keep this option **ON**; it is one of the best debugging tools. When you pause a procedure, you can hover with your mouse on a variable and its value will be displayed. More on this later as we will spend a considerable amount of time on debugging.

- 6- **Auto Indent Option:** This is an organization feature, when this option is set, the VBE automatically indent each new line of the code by the same amount as the previous line.

Tidbit: Use Tab key, not the space bar, to indent your code. Use Shift+Tab to unindent a line of code.

- 7- **Drag-and-Drop Text Editing:** When this option is set, it allows you to copy and move text by dragging and dropping. I would prefer using **Ctrl+X** to *Cut* and **Ctrl+V** to *Paste* to have better control on the placement of the code.
- 8- **Default to Full Module View:** When this option is set, procedures in a code window appear as a single scrollable window.
- 9- **Procedure Separator:** When this option is set, VBE displays separator bars between any two subsequent procedures or functions with the same Module.

```
Option Explicit
Sub test1 ()

End Sub

Sub test2 ()

End Sub

Sub test3 ()

End Sub
```