

EIGHT BUG REDUCTION TIPS – VBA

You cannot completely eliminate bugs in your programs, but there are a few tips that will help you keep them to a minimum

Use an Option Explicit at the beginning of your module – Doing so will require that you define the data type for every variable that you use. It's a bit more work, but you'll avoid the common error of misspelling a variable name.. And there's a nice side benefit: Your routines will often run faster.

Format your code with indentation – Using indentation to delineate code segments is quite helpful. If you have several nested For...Next loops, for example, consistent indentation will make it much easier to keep track of them all.

Be careful with On Error Resume Next – This statement causes Excel to ignore any errors and continue. In some cases, using this statement will cause Excel to ignore errors that shouldn't be ignored. You may have bugs and not even realize it.

Use lots of comments – Nothing is more frustrating than revisiting code that you wrote six months ago – and not having a clue as to how it works. Adding a few comments to describe your logic can save you lots of time down the road.

Keep your subroutines and functions simple – Writing your code in smaller modules, each of which has a single, well-defined purpose, makes it much easier to debug them.

Use the macro recorder to help you identify properties and methods – If I can't remember the name or syntax of a property or method, it's often quicker to simply record a macro and look at the recorded code.

Consider a different approach – If you're having trouble getting a particular routine to work correctly, you might want to scrap the idea and try something completely different. In most cases, Excel offers several alternative methods of accomplishing the same thing.

Understand Excel's debugger – Although it can be a bit daunting at first, you'll find that Excel's debugger is an excellent tool. Invest some time and get to know it.