

Editing macro code

Now you are sure the macro can be run by both the **Run** button in the Macro dialog, and by the **Ctrl + Shift + T** shortcut keys you specified, but you probably will not need it to return focus to cell A2 after each run. The code can be edited to remove the instruction to return focus, and also to style the text it writes:

- 1 On Excel's Developer tab, click the **Visual Basic** button in the Code group to launch the Visual Basic Editor
- 2 In Project Explorer, double-click the project's **Module1** item to see the macro VBA code
- 3 Next, delete this instruction line that returns focus **Range("A2").Select**
- 4 Now, add these instructions anywhere within the subroutine to style the text in bold red
ActiveCell.Font.Bold = True
ActiveCell.Font.Color = vbRed

- 5 Click the  **Save** button

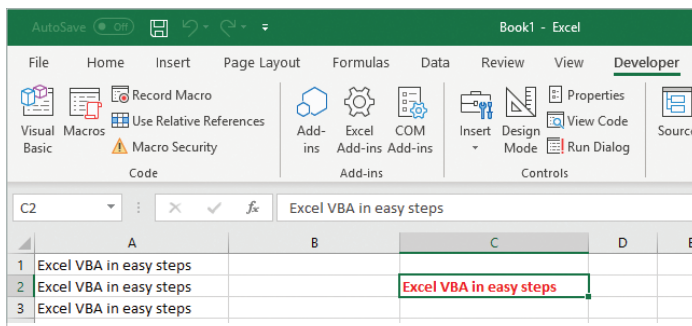
```

(General) BookTitle
Sub BookTitle ()
    BookTitle Macro
    Keyboard Shortcut: Ctrl+Shift+T

    ActiveCell.Font.Bold = True
    ActiveCell.Font.Color = vbRed

    ActiveCell.FormulaR1C1 = "Excel VBA in easy steps"
End Sub
    
```

- 6 Return to Excel and select any cell – press the **Ctrl + Shift + T** shortcut keys to run this edited macro



As you type instructions a pop-up box will often appear when you type a period/full stop, to offer a list of suggestions from which you can choose an item.



The eight Visual Basic color constants are **vbRed**, **vbGreen**, **vbBlue**, **vbYellow**, **vbMagenta**, **vbCyan**, **vbBlack**, and **vbWhite** – see pages 52-53 for more on constants.



Although the lines of VBA code are executed from top to bottom, their order is unimportant in this macro – the cell's styling can be set before or after its content is added.