

# The macro recorder

Not only are you able to write Visual Basic code using the Excel VBE (Visual Basic Editor), but you can get Excel to write it for you using the macro recorder! We learn how to record a macro, and discuss its advantages and shortcomings.

## Covers

- To record and run a macro | 68
- To edit a macro | 70
- Project Explorer | 72
- Macros available to other workbooks | 74
- To run a macro from a toolbar button | 77

# Chapter Five

# To record and run a macro

We will record and run the simplest of macros, a macro which will place a number into a cell when we press a certain key combination.

## To record the macro

Our macro will place the number 2 into cell A1 when we press the Ctrl+q key combination.

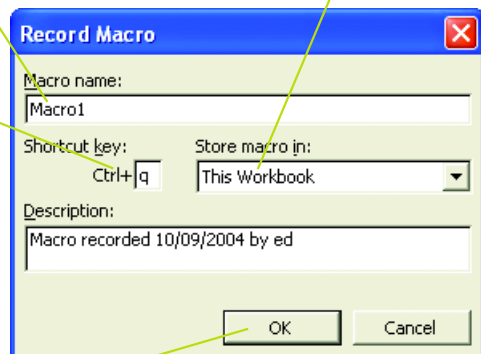
Before we start recording, select any cell apart from cell A1. From the Excel menu, choose Tools, Macro, Record New Macro...



2 Accept the default name for the macro. In future you may wish to give it a more meaningful name.

3 Choose This Workbook. Be careful not to select Personal Macro Workbook at this stage.

4 Choose the key combination that we would like to use to run the macro (Ctrl+q).



5 Click OK.



The key combination used to run the macro is case sensitive.

## ...cont'd



The macro recorder is recording your every move – mistakes included!

The Stop Recording toolbar appears

	A	B	C	D	E
1					
2					
3					
4					
5					
6					

6 Click into cell A1, type 2 and then press Enter.

	A	B	C	D	E
1	2				
2					
3					
4					
5					
6					



If you do close the Stop Recording box in error, you will need to choose View, Toolbars, Stop Recording from the Excel menu to switch it back on while the macro is recording!

7 Click the Stop button (the solid blue square). Do not click the Close button (x) to close this box!

Congratulations, you have just recorded a macro.

### To run the macro

If the 2 is still in cell A1, delete it.

	A	B	C	D	E
1	2				
2					
3					
4					
5					
6					



If you arrange the Excel and VBE windows on your screen so that you can see both whilst recording a macro, you can see the macro (VBA) code written before your very eyes!

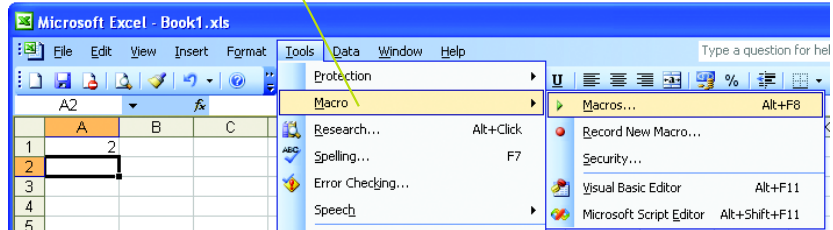
From Excel, press Ctrl+q. The 2 should appear in cell A1.

That's all it takes to run an Excel macro!

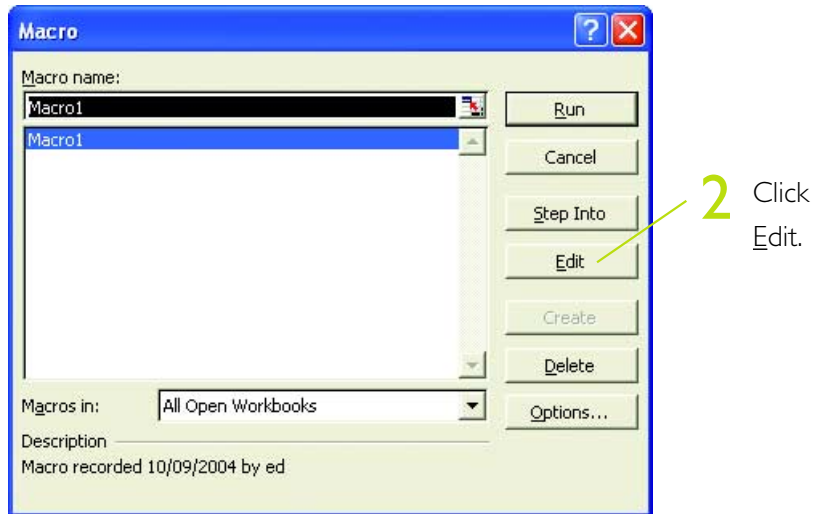
# To edit a macro

We will now view and modify the VBA code produced by the macro recorder.

From the Excel menu, choose Tools, Macro, Macros...



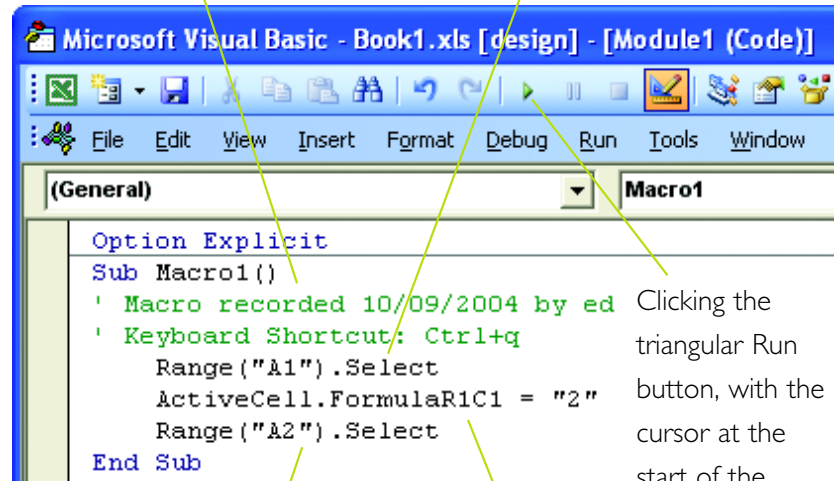
The Macro dialog box appears:



A code module now appears (see next page). Note the name in this case – Module1. Unlike the code modules that we have seen in the past, this is a general purpose code module (generally known as a standard code module) for the whole workbook, whereas the modules that we have seen so far were specific to and only available for that particular worksheet.

Comments have been automatically placed in the code

The Macro Recorder frequently uses `Select` and `ActiveCell` (another property which returns a `Range` object). We will show how this redundancy can be removed below



Clicking the triangular Run button, with the cursor at the start of the code, is an alternative way of running the macro

If the cursor moved down one cell when we pressed the Enter key (the default behavior) when we were recording the macro, this `Select` method statement will also be included

`FormulaR1C1` is used rather than `Value` by the Macro Recorder

### Removing redundancy

Remove `Select` and `ActiveCell` from the macro code and make one line as shown below.

```
Range("A1").FormulaR1C1 = "2"
```

If the macro is run again, the result will be exactly the same, i.e. you will see 2 placed into cell A1 (if it is not already there).