

# Excel macros made easy



Jane Barrett, IT Training & Engagement Team  
Information System Services

Version 1.1



# Scope

## Learning outcomes

- Understand the concept of what a macro is and what it does.
- Demonstrate how to record a macro.
- Understand how macros can be assigned within workbooks.

# Table of Contents

<b>Scope</b> .....	<b>2</b>
<b>Table of Contents</b> .....	<b>2</b>
<b>Introduction</b> .....	<b>3</b>
<b>Macros or Visual Basic for Applications (VBA)?</b> .....	<b>3</b>
What are macros and VBA? .....	3
Writing your first macro .....	3
Write a simple macro .....	3
Practice writing some macros .....	5
<b>Choosing where to store macros</b> .....	<b>5</b>
The Workbook .....	5
Personal Macro Workbook .....	5
New Workbook .....	5
<b>Choosing macro shortcut keys</b> .....	<b>6</b>
<b>Recording a macro with absolute references</b> .....	<b>6</b>
Write a macro with an absolute reference .....	6
<b>Understanding macro security</b> .....	<b>7</b>
Three ways to trust a macro enabled workbook .....	7
<b>Saving macros</b> .....	<b>8</b>
Saving your macros .....	8
<b>Implementing macro security</b> .....	<b>8</b>
Setting the Trust Center's macro settings .....	9
Set a trusted folder for your macros .....	9
<b>Recording macros with relative references</b> .....	<b>10</b>
Record a macro with relative references .....	11
<b>Using shapes to run macros</b> .....	<b>12</b>
Assign a macro to a button.....	12
Add a shape to your worksheet.....	13
Connecting the macro to the shape .....	14
Connecting a hyperlink to a shape .....	14
<b>VBA (the code behind the macros)</b> .....	<b>15</b>
Record a macro and watch the VBA appear .....	16
<b>Contacts</b> .....	<b>17</b>
For queries or help .....	17
<b>Useful resources</b> .....	<b>17</b>
Training videos .....	17
Booking other courses.....	17
<b>Your feedback</b> .....	<b>17</b>
Help us improve your training.....	17

# Introduction

Have you ever spent too much time on a repetitive task in an Excel worksheet trying to reorganise or format data?

If this sounds familiar, then you need to know about **macros**. This course will show you how they can be used to turn repetitive tasks into single clicks. You will learn how to record, edit and assign your macros to workbooks, or worksheets, and start to explore what VBA, (Visual Basic for Applications) has to offer.

## Macros or Visual Basic for Applications (VBA)?

### What are macros and VBA?

Generally there is a lot of confusion when the words **macro** and **VBA** are mentioned, often in the same sentence. A macro simply records mouse clicks as you go about a task and then converts the actions into a VBA programming code. The advantage of this is that you can record a repetitive task once using a macro and then play all the steps back again using a single mouse click.

To use macros you just need to understand the task you wish to record. If you want to look at the code and alter it, you can, **but you don't have to**, the macro does all the hard work for you. Macros can give you a good insight into how VBA works, so if you would like to learn to write it at a later date, it will give you a good starting point.

VBA is not just used in Excel, but is the building block for all the main office applications.

### Writing your first macro

The macro recorder is able to record and playback every mouse click or keystroke you make. The process works in three distinct phases.

1. Assign a name and shortcut key for your new macro, then start the macro recorder to record.
2. Perform the series of mouse clicks and keystrokes you wish to record (if it's a long sequence it is a good idea to plan this phase before you start to avoid errors).
3. Stop the recorder - don't forget to do this otherwise you will end up with a very long macro!

Once you have completed these steps you can then play your macro using the shortcut key or review the VBA code written by the macro.

## Task 1

### Write a simple macro

1. Open a new Excel workbook.
2. Select **View** → **Macros** → **Record macro**.

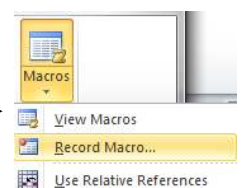


Figure 1

- The macro dialogue box appears, type **MyName** into the macro name box. (macro names must begin with a letter, no spaces are allowed).
- Type a lower case **n** into the Ctrl+ box.
- Select **This Workbook** in the store macro drop down list (the other options will be explained later).
- Write a short description of what your macro does in the Description box.
- Click the **OK** button then **STOP** and do nothing else!

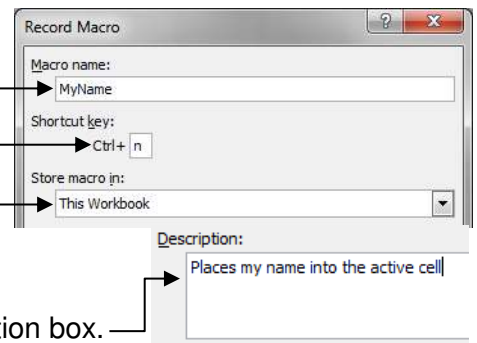


Figure 2

You are now recording your macro. There's no time element to a macro recording so you can be relaxed. However, the macro will now record every key press and mouse click, so it is important that you do not press any keys or click anywhere with the mouse.

### Record a macro that will put your name into the active cell

- Type your **name** into the currently active cell, remember that every click and key stroke is recorded so be careful not to move to any other cell. If you did your name would always appear in the cell you moved to when the macro was played.
- Press **Ctrl + Enter**, this saves the value to the cell without moving to the next line. This keystroke combination is really useful when recording macros.

### Stop the macro recording

You can do this in two ways:

- click, View → Macros → Stop Recording or
- click the **Stop button** on the bottom left of the status bar

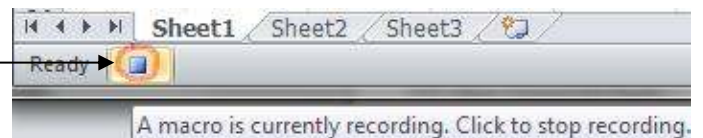


Figure 3

It is quite common for people to forget they are recording a macro and remember to stop it.

### Test the macro

- Click in any blank cell.
- Click **View → Macros → View Macros**. The macro dialogue appears listing only the **MyName** macro, as you have only recorded one macro so far.
- Click the **Run** button. Your name should appear in the currently active cell.
- Click in another blank cell and press **Ctrl + n**, once again your name appears in the active cell.

## End task

## Task 2

### Practice writing some macros

Using the same work book, create macros to:

1. format text to font **Arial Black**, size **16** and **centered**
2. format a number to show **5** decimal places
3. write your name in cell **C3**
4. change the sheet colour to **yellow** and all text to **red**
5. save the workbook as **Macros 1** - as the workbook contains macros you must select the **Save as type** to **Macro enabled workbook** otherwise you will lose your macros.

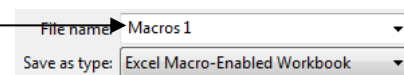


Figure 4

## Task End

## Choosing where to store macros

When you record a macro, Excel offers you three possible places to store it.

Each one determines which workbooks your macro will be available to and who can access it. Therefore it is really important that you are aware of the differences in the choices on offer.

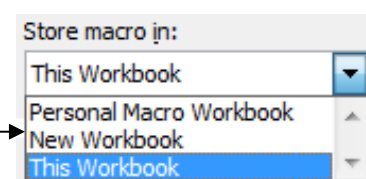


Figure 5

## The Workbook

If you select **This Workbook** as an option to store your macro then the macro you create will be available to other workbooks as well, but only if the workbook containing the macro is open.

## Personal Macro Workbook

The **Personal Macro Workbook** is a special workbook called **Personal.xlsb**. This is stored in a folder called XLSTART buried deep in the file system where nobody is likely to find it and change it. Whenever you open Excel, the Personal Macro Workbook opens in the background, but it is hidden so you never see it. As it is always open every workbook has access to its macros at all times.

The main purpose of the Personal Macro Workbook is to act as a container to store macros that you want to be available to every workbook. However, if you store macros in the Personal Macro Workbook they will only work on your machine. If you email a piece of work that depends on one of these macros, it will not work on the recipient's machine. In this type of scenario you will want to store the macro in **This Workbook**.

## New Workbook

The **New Workbook** option is rarely used as it opens a new Excel file and the macro is only available in this new file, hence you would always have to make sure that this file was open if you wanted to access the macro.

[Click here to download full PDF material](#)