Microsoft Word 2013™
Macros (Level 3)

Contents
Introduction 1
Creating a Macro 2
Running a Macro 4
Editing a Macro 4
A Second Example 5
Word Units 6
A More Complex Example 6
Setting Changing Shortcut Keys 8
Copying Macros to Another Document or Template 8
The DEVELOPER Tab 9
Macro Security Levels 10
Deleting a Macro 11

Introduction

A macro is a series of commands and instructions grouped together which, when run, carry out specific tasks automatically. They are especially useful for repetitive tasks or ones which involve several separate steps. You can use macros to:

- carry out routine editing and formatting
- combine multiple commands
- automate a complex series of tasks

In fact the majority of Word commands themselves are macros - there are over 1500 macros built into the system.

There are two ways to create your own macro in Word. The easier method is to use the Macro Recorder to record the actions you want carried out. Having recorded them once, they can then be repeated over and over again simply by using the macro. The other method is to write the macro in Visual Basic. This is much more difficult and to do it you need to know how to program. You can also record a macro and then use the Visual Basic Editor to alter part of it.
Microsoft Word 2013 Macros

Macros can be assigned to a Quick Access Toolbar button or shortcut key. You can also press <Alt F8> (or use the [Macros] button on the VIEW tab) to display a list of the available macros. You can then select the required macro from the list and run it.

By default, macros are stored in the special Normal.dotm template so that they are available for use with all documents but they can also be stored in a specific template or in a document. Because viruses are often spread through macros, they now have to be stored in special macro-enabled files and templates.

Creating a Macro

In order to write a macro from scratch, using Visual Basic, you need to know how that programming language works. In this document it is assumed that you do not have that expertise and so the Macro Recorder will be used instead. This is in fact the way most macros are created, even by users who do know Visual Basic. The code produced by the Recorder can then be edited to produce exactly what’s required. You will be trying this out later in the notes.

Let’s begin with a relatively simple macro which automatically types some text (e.g. your name or department name) in bold letters and centres it. If you want the macro stored in a particular template then you must either open or create a document based on that template or open the template file itself. If you want the macro available in all documents or just in a particular one, then you open or create the document in the usual way (as described below):

1. Load up Word, as usual, or press <Ctrl n> to create a new document
2. Move to the VIEW tab, click on the down arrow attached to [Macros] on the far right then choose Record Macro...

The Record Macro dialogue box will appear:

3. Enter a name for the macro - type BoldName

Note that the name must consist of a continuous alphanumerical sequence - you cannot use Bold-Name or Bold Name.

By default the macro will be stored in Normal.dotm, a template used to store macros for all your documents. You can, however, store the macro in the current document by selecting this option from the Store macro in: box:
Microsoft Word 2013 Macros

4. Click on the list arrow on the right of the Store macro in: box and choose Document1 (document)
5. If you want, type a description for the macro in the Description: box
6. To assign your macro to the keyboard or button, click on the appropriate icon under Assign macro to - here, click on [Keyboard] and the Customize Keyboard dialog box will appear:

7. In the Press new shortcut key: box enter the keystrokes required, e.g. <Alt b> (case is significant)
8. Change Save changes in: to the current document (Document1)
9. Check this isn’t already assigned under Currently assigned to: then press <Enter> for [Assign]
10. Finally, press <Esc> to [Close] the Customize Keyboard window

Everything you now do is being recorded, so take great care to do it properly. Note the mouse pointer in Word has a recording icon attached to it.

11. Carry out the sequence of actions you want to record:
   a. Press <Ctrl b> or click on [Bold] to turn on bold text
   b. Press <Ctrl e> or click on the [Center] button to centre the text
   c. Now type in your text (here, type in your name or that of your department)
   d. Press <Ctrl b> or click on [Bold] again to turn off bold text
   e. Press <Enter> then press <Ctrl l> or click on [Align Left] to return to normal typing
12. Click on the [Macros] list arrow and choose Stop Recording to finish recording your macro

IMPORTANT: when recording a macro, mouse movements made within the document window are not saved. For example, you cannot use the mouse to move the insertion point, to select text or items, or right click to copy, for example, but you can use it to click on menus and other options.

Tips

• Before you begin recording a macro, plan the steps you want the macro to perform - it’s a good idea to write them down on paper in the order required
• If you make an error while recording a macro this, together with the corrections, will also be recorded. However, you can always edit the macro later to remove any unwanted steps
• Note the Pause Recording option - you can use this to temporarily stop recording, test out the next sequence of steps then Resume Recorder again
• Some commands display dialog boxes and you may need to answer these inside the macro. This can sometimes be avoided - for example, if you save a document immediately before closing it then you are not asked whether you want to save any changes
Microsoft Word 2013 Macros

- If the macro includes the command Advanced Find or Replace, click on [More] button on the Find tab, and make sure All is set in the Search box. This stops Word asking whether you want to continue searching if you reach the end of the document.
- If you want to use the macro in other documents make sure it doesn’t depend on the current content.
- If you use a particular macro often, assign it to a toolbar button or shortcut key. That way, you can run the macro directly without having to open the Macros dialog box.

Running a Macro

You can now try out your newly created macro as follows:

1. Press <Ctrl a> to select your current text and <Delete> it.
2. Now press <Alt b> - your text should appear, centred and in bold letters.

Macros which have not been assigned to a keystroke (or toolbar button) must be run using the [Macro] button on the Ribbon. Even macros which have been assigned can be run using this method, as you will see:

3. Click on the [Macros] button (or press <Alt F8>)
4. Make sure All active templates and documents is selected in the Macros in: box
5. In the Macro name box, select the macro you want to run - BoldName
6. Press <Enter> or click on [Run] - your text should appear a second time.

Tip: To see the built-in macros, press <Alt F8> then select Word commands from the Macros in: drop-down list.

Editing a Macro

You can make changes to a macro you have recorded using the Visual Basic Editor. For example, you may wish to remove unnecessary steps, or add instructions that could not be recorded in Word. Ideally, you need to know how to write programs in Visual Basic to do this. However, follow the instructions below and try to make sense of what’s happening:

1. Click on the [Macros] button (or press <Alt F8>)
2. Make sure the Macros in: box is set to All active templates and documents
3. In the Macro name: box select the macro you want to edit (here, BoldName) then click on [Edit]

The Editor window now appears. Don’t worry too much about how this works; just read through the macro instructions, which hopefully will make some sense.
Microsoft Word 2013 Macros

4. Find the line which types out the text and edit it in some way - (e.g. add your middle name)
5. In the line above which ends in ParagraphCenter, change Center to Right
6. Click on the [Close] button (or press <Alt q>) to close the editor and return to your document (the changes are saved automatically)

Tip: You can switch between your document and the Visual Basic Editor by pressing <Alt F11>. Note that you will need to close the Editor window when you have finished editing.

7. End by testing out your macro - press <Alt b> - you should find your new text is displayed, this time aligned on the right

Though the macro appears to work perfectly, it could in fact be improved. You may have noticed the lines which read Bold = wdToggle. By clicking on the [Bold] button, bold wasn’t turned on, it was toggled. If you switch bold on before you run the macro, then your text doesn’t appear in bold and at the end of the macro you are not returned to plain letters. Try it:

8. Type in some text (to check it appears as normal then press <Enter> for a new line
9. Press <Ctrl b> to turn on [Bold] then use <Alt b> to run the macro and note what happens
10. Type in some more text (it appears in bold!) then press <Enter> for a new line

To overcome this problem you would need to establish normal text at the very start of your macro. You could have done this when recording the macro (display the Font dialog box by clicking on the group arrow then select a Font Style of Regular). Try this, if you like, or edit the first line of the macro as follows:

11. Press <Alt F11> to switch into the Editor
13. Close the Editor (<Alt q>) then run your macro again - it should work whether or not bold is already on

A Second Example

This second example of a macro reverses the order of two words. It could be used, for example, to reverse someone’s name from Surname Firstname to the more conventional Firstname Surname order.

1. Type in your name (limit it to two words – your name and family name)
2. Move to the VIEW tab, click on the down arrow attached to [Macros] then choose Record Macro...
3. Enter a name for the macro - e.g. NameSwitch
4. Store the macro in the current document by selecting this from Store macro in:
5. This time, assign it to a [Button] – the Word Options dialog box appears
6. Select Project.NewMacros.NameSwitch then [Add >>] it to the Quick Access Toolbar
7. Next, click on [Modify...] and choose a symbol for the toolbar button then press <Enter> for [OK]
8. Finally, click on [OK] – your button should be added to the Quick Access Toolbar

Next, record the macro:

9. Carry out the following sequence of actions:
   a. Press <Ctrl Shift left_arrow> to select the second part of your name
   b. Now press <Ctrl x> to [Cut] it to the Clipboard
   c. Press <Ctrl left_arrow> to move the insertion point before the first part of your name
   d. Now press <Ctrl v> to [Paste] it from the Clipboard
   e. Finally, press <Ctrl right arrow> to move the insertion point to the end of the line
Microsoft Word 2013 Macros

10. On the VIEW tab, click on the [Macros] list arrow and choose Stop Recording to end your macro.

11. Test out your macro by clicking on the new Quick Access Toolbar button. The macro should reverse the order of any two words – try it elsewhere in your text, if you like.

Word Units

If you look at the code recorded in the NameSwitch macro, you will find reference to wdWord (another Word-specific parameter is wdToggle). This is how a macro picks up a single word in the text. Other units include wdCharacter, wdSentence, wdParagraph, wdLine, wdTable, wdCell, wdColumn, and wdRow.

Selecting these units using the keyboard (remember, you can’t use the mouse) isn’t well known, so here’s a list of how to do so:

- wdCharacter – hold down <Shift> and press <left_arrow> (or <right_arrow>)
- wdWord – hold down both <Shift> and <Ctrl> and press <left_arrow> (or <right_arrow>)
- wdSentence – hold down both <Shift> and <Ctrl> and press <up_arrow> (or <down_arrow>)
- wdLine – hold down <Shift> and press <Home> (or <End>) – if you press just <Home> or <End> first, you move the insertion point to the start/end of the line; then select using <Shift> as well
- wdCell – press <Home> to move to the start of the text in the cell then <Shift End>
- wdColumn – press <Alt Page Up> to move to the top cell then <Shift Alt PageDown>
- wdRow – press <Alt Home> to move to the leftmost cell then <Shift Alt End>
- wdTable – hold down <Alt> and press <5> on the numeric keypad with <Num Lock> turned off (or use combinations of <Alt PageUp/Home> and <Shift Alt PageDown/End> as above)

Note: In the above, holding down <Shift> selects – if you don’t, the key presses move the insertion point.

A More Complex Example

This next example involves a little more programming. If you want to apply a macro to some text where the number of letters varies, you may need to build this factor into your macro. This next macro shows you how (it makes alternate letters in a word upper/lower case). Begin by recording some of the steps:

1. Click immediately after your name to reposition the insertion point
2. On the VIEW tab, click on the down arrow attached to [Macros] then choose Record Macro...
3. Enter a name for the macro – e.g. UpperLower
4. Store the macro in the current document by selecting this from Store macro in:
5. Assign it to a [Button]
6. Select Project.NewMacros.UpperLower then [Add >>] it to the Quick Access Toolbar
7. Next, click on [Modify...] and choose a symbol for the toolbar button then press <Enter> for [OK]
8. Finally, click on [OK] – a second button should be added to the Quick Access Toolbar

IMPORTANT: the macro is now recording so take care not to do anything you don’t want in the macro.

9. Carry out the following sequence of actions:
   a. Press <Ctrl Shift left_arrow> to select the word
   b. Press <left_arrow> to release the selection and place the typing position before the word
   c. Hold down <Shift> and press the <right_arrow> key once to select the first character
   d. On the HOME tab, click on [Change Case] and choose UPPERCASE
Microsoft Word 2013 Macros

e. Press <right_arrow> to release the selection
f. Repeat step 9c to select the next character
g. Repeat step 9d, setting [Change Case] to lowercase (then press <right_arrow>)
h. Press <Ctrl right_arrow> to move to the end of the word

10. Move to the VIEW tab, click on the [Macros] list arrow and choose Stop Recording to end your macro

Obviously the macro so far only works on the first two letters of a word. Test it out, if you like, by typing in another word then run the macro. To get it to work for the whole word requires some extra programming:

11. Click on the [Macros] button (or press <Alt F8>)
12. Select the macro UpperLower and click on [Edit] – all the macros are displayed

Tip: You could also have pressed <Alt F11> at step 11 to switch directly to the Visual Basic Editor.

You now have to add two extra features to the macro. Firstly, the computer must work out how many characters there are in the selected word then, secondly, it needs to repeat the code setting up the case for each character in turn. The first problem is solved by using the built-in function, Len(), while for the second you need to set up a loop. Two variables are used – x holds the length of the word (plus the following space) while n acts as a counter for the loop. If necessary, delete any superfluous lines of code you may have accidentally recorded.

13. Edit the macro so that it reads as below (the edits are shown in bold):

   Sub UpperLower()
     ' UpperLower Macro
     Selection.MoveLeft Unit:=wdWord, Count:=1, Extend:=wdExtend
     x = Len(Selection) + 1
     Selection.MoveLeft Unit:=wdCharacter, Count:=1
     For n = 1 To x / 2
       Selection.MoveRight Unit:=wdCharacter, Count:=1, Extend:=wdExtend
       Selection.Range.Case = wdUpperCase
       Selection.MoveRight Unit:=wdCharacter, Count:=1
       Selection.Range.Case = wdLowerCase
       Selection.MoveRight Unit:=wdCharacter, Count:=1
     Next n
     Selection.MoveRight Unit:=wdWord, Count:=1
   End Sub

14. When you have finished editing the macro, [Close] the Editor
15. At the end of your text, type in another word then click on your [UpperLower] button to run the macro

If the macro doesn’t work properly, double check your code matches that above in the Visual Basic Editor.
Setting Changing Shortcut Keys

In the last two examples, you assigned the macro to a toolbar button instead of a shortcut key. Sadly, you don’t have the opportunity to do both when recording the macro. You can set a key later, however:

1. Click on the [Customize Quick Access Toolbar] down arrow to the right of the Quick Access Toolbar in the top left of your Word window, and choose More Commands…
2. Next, choose Customize Ribbon from the list on the left.
3. Under the heading Keyboard shortcuts: at the foot of the Word Options dialog box, click on [Customize…] – the Customize Keyboard dialog box appears.
4. Make sure Save changes in: is set to the current document (i.e. Document1).
5. Under Categories: scroll down and select Macros (any EN macros in the list are those used by EndNote).
6. In the Macros: box on the right select the macro to be redefined (here, UpperLower).
7. Click in the Press new shortcut key: box, press in a new key combination, e.g. <Alt u> then [Assign] it.
8. [Close] both the Customize Keyboard and Word Options dialog (click on [OK]) boxes.
9. Type in a new word then press <Alt u> to check your shortcut key is working.

Copying Macros to Another Document or Template

You can copy a macro for use in another document or template using the Organizer. This also lets you rename a group of macros, known as a Project. Individual macros can be renamed in the Visual Basic Editor.

In this next exercise, you are going to copy your macros from your original document into a new one.

2. Change Save as type: to a Word Macro-Enabled Document (*.docm), enter a file name (e.g. TestMacro) then press <Enter> for [Save].

Note that files containing macros have to be saved as a special file type. If you were to try and save your document as an ordinary Word document, you would be given the choice of either saving as a macro-free document (the macros would automatically be discarded) or to return to the Save As dialog box to change the type to a Word Macro-Enabled Document.

4. Press <Alt b> - you’ll find that the BoldName macro no longer works (neither do the toolbar buttons).
5. Press <Alt F8> to display the macros - there shouldn’t be any of yours.
Microsoft Word 2013 Macros

6. Click on [Organizer...] and check you are on the Macro Project Items tab (you can also move Styles)

In the Organizer window, the left half of the window represents the active document, Document2. The right half of the window is currently set to the default template, Normal.dotm, but you can change this if you want to move the macros somewhere else:

7. Close the Normal.dotm file by clicking on [Close File...] on the right
8. Now click on the same button which has been renamed [Open File...] – it will be showing all the templates in a Templates folder on the right-hand side of the Open window
9. Click on the Documents folder on the left-hand side of the Open window
10. Change the setting to the right of the File name: box to Word Macro-Enabled Documents (*.docm)
11. Choose your original document (TestMacro.docm) and [Open] it

12. Select NewMacros then click on [< - Copy] to add them to your Document2 file

The current name for the set of macros isn’t very helpful and it’s a good idea to change this:

13. Select it in the left-hand box, click on [Rename...] and change the name to TestMacros - click on [OK]
14. Click on [Close] to shut down the Organizer window
15. Press <Alt F8> to display the available macros (all three should now be showing)
16. Select BoldName and [Run] it

Note: You cannot use a shortcut key to run the macro as that information isn’t passed across in the Organizer. Try setting up the shortcut keys, as described in the previous section.

The DEVELOPER Tab

If you use macros a lot, you may wish to add the DEVELOPER tab to your Ribbon. This has its own buttons for [Record Macro], [Pause Recording] etc. It also has controls which can be used on Forms. To display the tab:

1. Move to the FILE tab and choose Options
2. Next, click on Customize Ribbon
3. In the pane on the right, turn on the DEVELOPER tab by clicking in the check box next to Developer
4. Press <Enter> for [OK] to display the tab (and close the Word Options dialog box)
Microsoft Word 2013 Macros

Macro Security Levels

Because macros have been used to spread viruses between computers, Microsoft has introduced Security settings into Office. On the PCs in the IT labs, this is set at the lowest level as the University already has comprehensive anti-virus software. On your own PC, however, this might not be the case. The security check is carried out whenever a file is opened.

To see the security settings:

1. Click on the [Macro Security] button on the new DEVELOPER tab

   **Note:** when this tab isn’t displayed, click on the FILE tab, then Options (on the left) followed by Trust Center (again on the left) and finally the [Trust Center Settings...] button (on the right).

2. Choose the level of security required - here set it to Disable all macros with notification then press <Enter> for [OK]

3. Switch to your original macros document (TestMacro.docm) and [Close] it

4. Now reopen the document (move to the FILE tab, click on Open and select it from the list of files under Recent Documents)

   If an extra yellow bar appears below the Ribbon:

5. Click on [Enable Content] to enable the macros

   **Note:** If you set security to Enable all macros then no macro checking is performed (files are opened automatically). If the anti-virus software on your computer is kept up-to-date (as is the case on the PCs in the IT labs) then there is no problem setting this level; if it isn’t then you are asking for trouble if you choose this option.

6. You are now asked whether you want to make this a Trusted Document – if you answer [Yes] then the content will be enabled when you next open this file; if [No] then another Security Warning appears

   **Note** also that Windows allows you to nominate Trusted Sources for macros. You can set up a special folder on your own PC in which to store macro files and then declare this to be a Trusted Source. Opening files from the folder will then bypass the macro security.
Microsoft Word 2013 Macros

Deleting a Macro

In the above exercises, both macros were only stored in the documents themselves, so you can get rid of them simply by deleting the documents. If you wanted to remove the macros from a file (or the Normal.dotm template) you can do so either as a single project via the Organizer (simply select it and click on [Delete]) or via the Macros dialog box:

1. Click on the [Macros] button on the VIEW or DEVELOPER tab (or press <Alt F8>)

2. In the Macros in: box, check that it is set to the current document (e.g. TestMacro.docm)
3. In the Macro name: box, click on the name of the macro you want to delete
4. Click on [Delete] and confirm the deletion by pressing <Enter> for [Yes]
5. Press <Esc> or click on [Close] to close the Macros dialog box
6. End by closing down Word – there’s no need to save the files
7. <Delete> the file TestMacro.docm from Documents – you don’t need this any more

Note: See Microsoft Excel 2010 Visual Basic for further training on writing macros. Though this uses Excel, the code is equally applicable to Word.