# CHAPTER 16 MACROS

INFOCUS

***Macros*** enable you to automate routine operations and tasks in Microsoft programs. A macro is a small program that records the actions you take to complete a specific task, for example, adjusting the page layout to a specific format. You then run the macro to apply all of the recorded actions in just a single click! Needless to say the time-saving potential with well-developed macros is huge.

Further, if you need to shield users from complex tasks, then perhaps a macro is the answer. For example, if you need to pull together five or six documents to create a monthly report, a macro would enable you to hand this process over to even a novice user.

## In this session you will:

- ✓ gain an understanding of ***Word*** macros
- ✓ learn how to set a macro security level
- ✓ learn how to save a document as macro-enabled
- ✓ learn how to record a macro
- ✓ learn how to run a macro
- ✓ learn how to assign a macro to the ***Quick Access Toolbar***
- ✓ learn how to assign a keyboard shortcut to a macro
- ✓ learn how to edit a macro
- ✓ learn how to delete a macro
- ✓ learn how to create a ***MacroButton*** field
- ✓ learn how to copy a macro
- ✓ gain an understanding of how to develop macros.

# UNDERSTANDING MACROS IN WORD

In earlier versions, macros allowed you to record the keystrokes used to perform an operation. These days, macros have evolved into a full programming language, allowing you to fully automate virtually every facet of document production. Macros have huge time-saving potential, as a series of tasks can be performed in just a single click.

## What is a Macro?

A ***macro*** is simply a programmed set of instructions that tell Word what it should do. Macros are written or recorded in a ***procedure***, otherwise known as a ***routine***.

## How are Macros Created?

Word offers two main ways of creating macros. Macros can be ***recorded*** using the built-in macro recorder, which records what you do and then converts this into a macro program. This is a great way of creating macros for performing routine, complex or boring and repetitious tasks. Once recorded, these tasks can be performed quickly over and over again by ***running*** the macro.

Macros can also be developed from scratch. In other words, you can ***type*** the programming steps yourself rather than recording them using the built-in macro recorder.

## What Types of Macros Are There?

There are two types of macro – ***global*** and ***local***.

A ***global*** macro is available to all documents. For example, you may have a macro that adds your company name and file details to the footer. Since you want all documents to have this, the macro to add the footer should be available to all documents and would therefore need to be a global macro.

A ***local*** macro is only available to a particular document or template. For example, you may have a monthly report document that needs to have information imported into it from other sources. You can set up a macro that will import for you so that you don't have to perform the steps each time.

## Where Are Macros Located?

Macros are located in special macro sheets attached to either a document or template. Local macros, because they pertain to one document, can be found in the macro sheet for that document. Global macros are stored in the macro sheet for the ***Normal*** template. This then makes them available to all documents based on that template.

Some advanced users store a collection of routines (macros) in a template and then attach that template to documents where those macros are required.

## How Do You Access Macros?

Macros can be created, viewed and edited using the various macro commands on the ***Code*** group of the ***Developer*** tab. If you cannot see the Developer tab, you will need to add it to the ***Ribbon***. This is done by ***customising the ribbon*** in ***Word Options*** to display the Developer tab.

Macros can only be viewed using the ***Visual Basic Editor*** which is accessed by clicking on ***Macros*** on the ***Developer*** tab or by pressing Alt + F11.

## What is VBA?

***VBA*** (***Visual Basic for Applications***) is the programming language used to create macros. Earlier versions of Word used a more primitive version of this language known as ***WordBasic***. VBA is a common programming language found in virtually all Microsoft Office applications. Once you have learned it for one product, you can easily adapt what you've learned to the other products.

## Do I Need to be a Programmer to create Macros?

Absolutely not! While macros may appear cryptic and difficult to understand at first, tools such as the macro recorder make creating macros easy and effortless.

# SETTING MACRO SECURITY

Recording or writing macros allows you to hack into Word and manipulate the application to suit your needs. One of the consequences of this is that macros become a potential source of viruses. To reduce the risk of viruses, Microsoft has a ***Trust Centre*** that allows you to enable or disable macros based on whether or not they are stored in a trusted location or have a digital signature.

## Try This Yourself:

**Open File** *Before starting this exercise you MUST open the file W881 Macros_1.docx...*

1 If the ***Developer*** tab is not visible, click on the **File** tab, click on **Options**, then click on ***Customise Ribbon*** and click on ***Developer*** for ***Customise the ribbon***, then click on **[OK]**

2 Click on the ***Developer*** tab, then click on ***Macro Security*** ⚠ to display the ***Trust Centre*** dialog box

*By default, all macros are currently disabled unless they are in a trusted location...*

3 Click on ***Trusted Locations*** to see the list of trusted folders

4 Click on **[Add new location]** to open the ***Microsoft Office Trusted Location*** dialog box

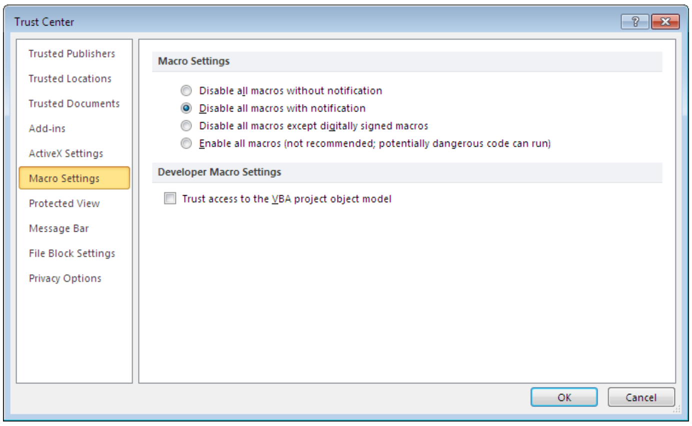
5 Click on **[Browse]** then navigate to and select the course files folder

6 Click on **[OK]**

*The course files folder will be added to the list of Trusted Locations...*

7 Click on **[OK]** to close the dialog box, then save and close the document

2



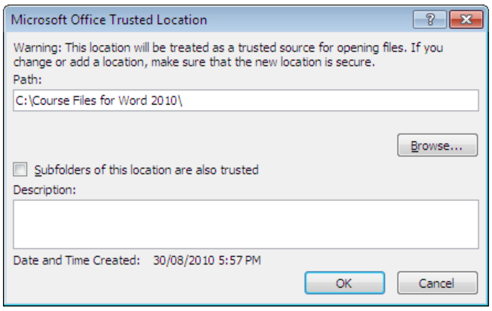
Macro Settings

○ Disable all macros without notification
● Disable all macros with notification
○ Disable all macros except digitally signed macros
○ Enable all macros (not recommended; potentially dangerous code can run)

Developer Macro Settings

□ Trust access to the VBA project object model

5



Warning: This location will be treated as a trusted source for opening files. If you change or add a location, make sure that the new location is secure.

Path:

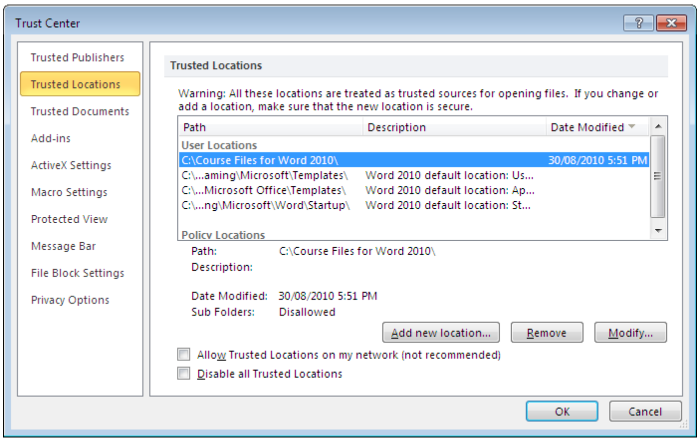
C:\Course Files for Word 2010\

□ Subfolders of this location are also trusted

Description:

Date and Time Created: 30/08/2010 5:57 PM

6



Trusted Locations

Warning: All these locations are treated as trusted sources for opening files. If you change or add a location, make sure that the new location is secure.

| Path | Description | Date Modified |
|---|---|---|
| User Locations | | |
| C:\Course Files for Word 2010\ | | 30/08/2010 5:51 PM |
| C:\...aming\Microsoft\Templates\ | Word 2010 default location: Us... | |
| C:\...Microsoft Office\Templates\ | Word 2010 default location: Ap... | |
| C:\...ng\Microsoft\Word\Startup\ | Word 2010 default location: St... | |
| Policy Locations | | |

Path: C:\Course Files for Word 2010\

Description:

Date Modified: 30/08/2010 5:51 PM

Sub Folders: Disallowed

□ Allow Trusted Locations on my network (not recommended)

□ Disable all Trusted Locations

## For Your Reference…

To ***set*** the ***level of macro security***:

1. Click on the ***Developer*** tab, then click on ***Macro Security*** ⚠
2. Click on the required level of security
3. Click on **[OK]**

## Handy to Know…

- Ideally you should only run macros that are digitally signed and from a trusted publisher. In reality, most macros aren't digitally signed and you need to place them in a trusted location to make them work. Bottom line: don't use macros that you aren't completely sure about.

# Saving A Document As Macro-Enabled

***Microsoft Word 2010*** enables you to save in several different file formats that control whether or not macros are saved with the file. The default document format of ***.docx*** does not allow macros to be saved with the document. To ensure that macro code can be saved, you must change the document type to ***.docm*** which is known as a ***Word Macro-enabled Document***.

## Try This Yourself:

*Same File*

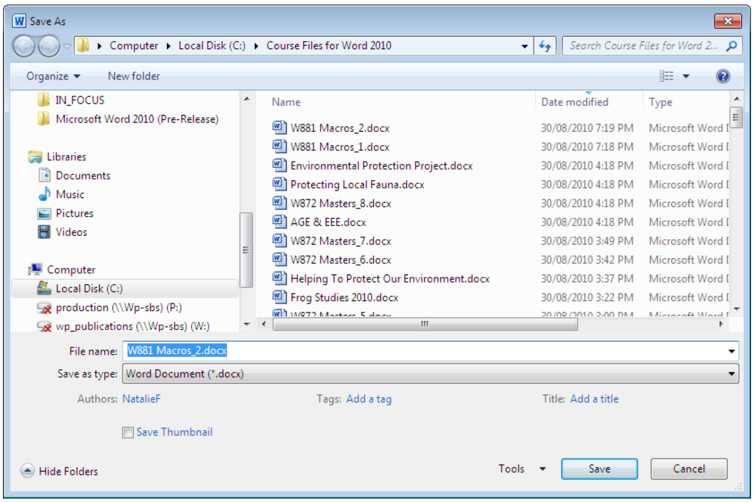
*Continue using the previous file with this exercise, or open the file W881 Macros_2.docx...*

1 Click on the ***File*** tab, then click on **Save As** to open the ***Save As*** dialog box

2 Click on the drop arrow ▾ for ***Save as type*** and click on ***Word Macro-Enabled Document (*.docm)***

3 Click on **[Save]**

*The document will now be saved with the file extension .docm. You can now record and run macros in this document*


Save As
Computer ▸ Local Disk (C:) ▸ Course Files for Word 2010
Search Course Files for Word 2...
Organize ▾
New folder
IN_FOCUS
Microsoft Word 2010 (Pre-Release)
Libraries
Documents
Music
Pictures
Videos
Computer
Local Disk (C:)
production (\\Wp-sbs) (P:)
wp_publications (\\Wp-sbs) (W:)
Hide Folders
Tools ▾
Save
Cancel

| Name | Date modified | Type |
|---|---|---|
| W881 Macros_2.docx | 30/08/2010 7:19 PM | Microsoft Word [ |
| W881 Macros_1.docx | 30/08/2010 7:18 PM | Microsoft Word [ |
| Environmental Protection Project.docx | 30/08/2010 4:18 PM | Microsoft Word [ |
| Protecting Local Fauna.docx | 30/08/2010 4:18 PM | Microsoft Word [ |
| W872 Masters_8.docx | 30/08/2010 4:18 PM | Microsoft Word [ |
| AGE & EEE.docx | 30/08/2010 4:18 PM | Microsoft Word [ |
| W872 Masters_7.docx | 30/08/2010 3:49 PM | Microsoft Word [ |
| W872 Masters_6.docx | 30/08/2010 3:42 PM | Microsoft Word [ |
| Helping To Protect Our Environment.docx | 30/08/2010 3:37 PM | Microsoft Word [ |
| Frog Studies 2010.docx | 30/08/2010 3:22 PM | Microsoft Word [ |

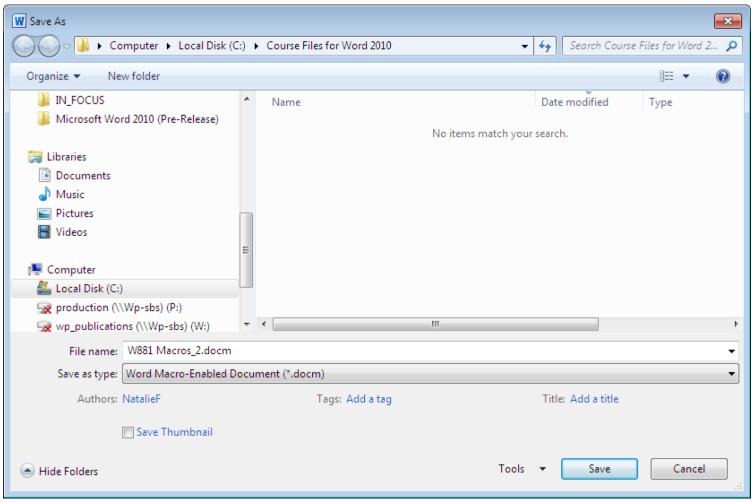
File name: W881 Macros_2.docx

Save as type: Word Document (*.docx)

Authors: NatalieF Tags: Add a tag Title: Add a title

□ Save Thumbnail

1


Save As
Computer ▸ Local Disk (C:) ▸ Course Files for Word 2010
Search Course Files for Word 2...
Organize ▾
New folder
IN_FOCUS
Microsoft Word 2010 (Pre-Release)
Libraries
Documents
Music
Pictures
Videos
Computer
Local Disk (C:)
production (\\Wp-sbs) (P:)
wp_publications (\\Wp-sbs) (W:)
Name
Date modified
Type
Hide Folders
Tools ▾
Save
Cancel

No items match your search.

File name: W881 Macros_2.docm

Save as type: Word Macro-Enabled Document (*.docm)

Authors: NatalieF Tags: Add a tag Title: Add a title

□ Save Thumbnail

2

## For Your Reference…

To ***save*** a ***document*** as ***macro-enabled***:

1. Click on the ***File*** tab, then click on **Save As**
2. Navigate to the save location, then click on the drop arrow ▾ for ***Save as type*** and click on ***Word Macro-Enabled Document (*.docm)***

## Handy to Know…

- Macros cannot be saved in documents with the file extensions ***.docx*** or ***.dotx***, while they can be stored in documents with the file extensions***.docm*** or ***.dotm*** (template document). You can create a macro in a document that is not macro-enabled, but you won't be able to save it.

# Recording A Macro

***Recording*** a macro is a bit like recording an answering message for your phone, and like a phone, you use a recorder to create it. The macro recorder takes care of writing the macro's commands. The first step is to specify a name for the macro and determine where it will be located. If you store it in the ***Normal*** template, it will become a ***global*** macro, available to all documents.

## Try This Yourself:

***Same File***

*Continue using the previous file with this exercise, or open the file W881 Macros_3.docm...*

1 Click on the ***Developer*** tab, then click on ***Record Macro*** to open the ***Record Macro*** dialog box

2 Type **NameDate** in ***Macro name*** (no spaces)

3 Click on the drop arrow for ***Store macro in*** and click on the name of the current document

4 Click in ***Description*** and type **Inserts the author's name and today's date.**

5 Click on **[OK]** to start recording

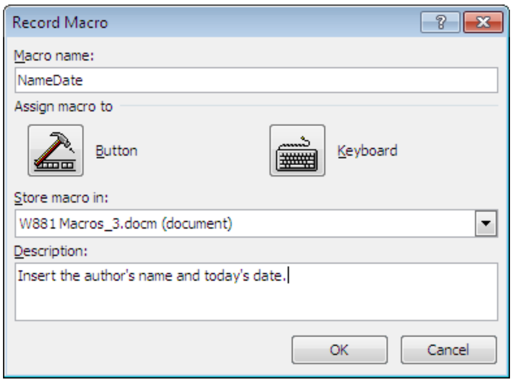
*The Record Macro tool will change to a Stop Recording tool...*

6 Carefully perform each of the steps listed in the macro sequence

7 Click on ***Stop Recording*** to stop creating the macro

*You have recorded the steps as you have completed them. The next step will be to test-drive the macro*

4



Record Macro

Macro name:

NameDate

Assign macro to

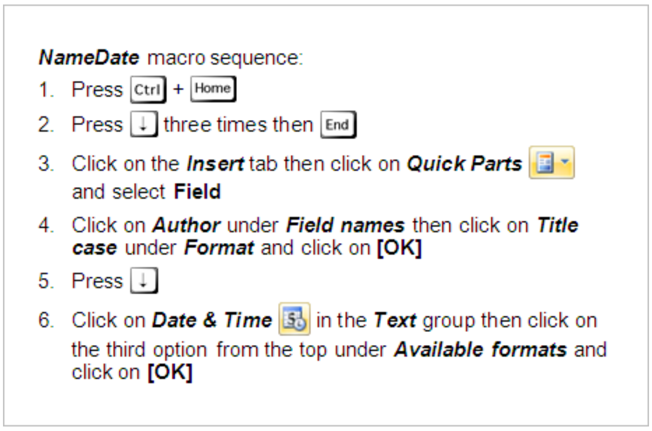
Store macro in:

W881 Macros_3.docm (document)

Description:

Insert the author's name and today's date.

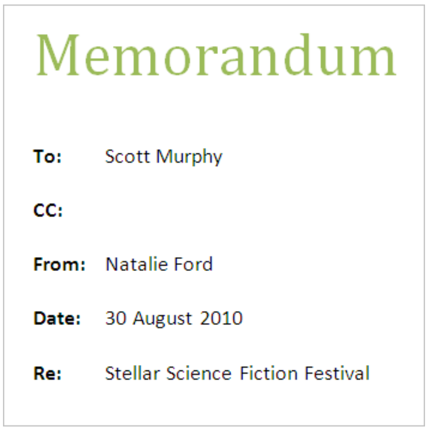
6



***NameDate*** macro sequence:

1. Press Ctrl + Home
2. Press ↓ three times then End
3. Click on the ***Insert*** tab then click on ***Quick Parts*** and select **Field**
4. Click on ***Author*** under ***Field names*** then click on ***Title case*** under ***Format*** and click on **[OK]**
5. Press ↓
6. Click on ***Date & Time*** in the ***Text*** group then click on the third option from the top under ***Available formats*** and click on **[OK]**

7



Memorandum

**To:** Scott Murphy

**CC:**

**From:** Natalie Ford

**Date:** 30 August 2010

**Re:** Stellar Science Fiction Festival

## For Your Reference…

To ***record*** a ***macro***:

1. Click on ***Record Macro*** on the ***Developer*** tab
2. Type the macro name and select a location
3. Click on **[OK]** then perform the steps
4. Click on ***Stop Recording***

## Handy to Know…

- The macro recorder does not allow you to use the mouse on the document itself. You must use keystrokes.
- Click on ***Pause Recording*** to stop recording temporarily.

# Running A Macro

Once a macro has been recorded it should be ***run*** at least once to ensure that it performs as expected. All macros available to the current document will be listed in the ***Macros*** dialog box.

This dialog box allows you to perform a variety of tasks on macros. You can ***edit*** them, ***delete*** them, and for the purpose of this exercise, ***run*** them.

## Try This Yourself:

***Same File***

*Continue using the previous file with this exercise, or open the file W881 Macros_4.docm...*

1 Delete the author and date text that were inserted when you created the macro

2 Click on ***Macros*** on the ***Developer*** tab to open the ***Macros*** dialog box

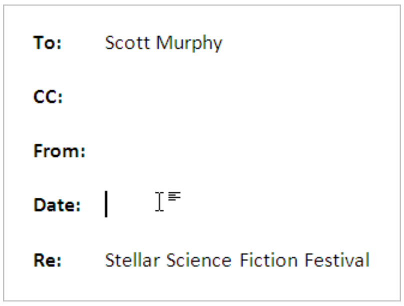
3 Click on ***NameDate*** if it is not already selected and click on **[Run]**

*The macro will run and insert the author and today's date in the correct positions.*

*As the cursor was positioned when recording the macro, it doesn't matter where the insertion point is when you run it. Depending upon what your macro does, you may need to position the insertion point first when recording a macro…*

4 Delete the author and date text that were inserted when you ran the macro, then save the document

1



**To:** Scott Murphy

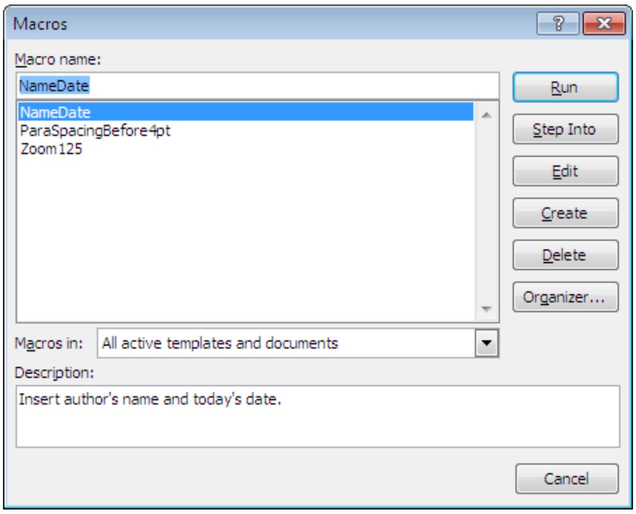
**CC:**

**From:**

**Date:**

**Re:** Stellar Science Fiction Festival

2



Macro name: NameDate

NameDate
ParaSpacingBefore4pt
Zoom125

Macros in: All active templates and documents

Description: Insert author's name and today's date.

## For Your Reference…

To ***run*** a ***macro***:

1. Position the insertion point if necessary
2. Click on ***Macros*** on the ***Developer*** tab
3. Click on the ***Macro name***
4. Click on **[Run]**.

## Handy to Know…

- The ***Macros*** dialog box can be displayed by pressing Alt + F8.

# ASSIGNING A MACRO TO THE TOOLBAR

Running a macro from the ***Macros*** dialog box is not the most efficient way to do it. The time saved running a macro is reduced by the need to open the ***Macros*** dialog box. One way around this is to assign the macro to a button on the ***Quick Access Toolbar***, which is always visible.This way you can run the macro by simply clicking on the button. You can also change the icon and button name.

## Try This Yourself:

***Same File***

*Continue using the previous file with this exercise, or open the file W881 Macros_5.docm...*

1 Click on the drop arrow for ***Customise Quick Access Toolbar*** on the ***Quick Access Toolbar*** and select **More Commands** to open the ***Word Options*** dialog box

2 Click on the drop arrow for ***Choose commands from*** and select **Macros**

3 Click on ***Project.NewMacros. NameDate*** then click on **[Add]**

*The macro will be added to the Quick Access toolbar list...*

4 Click on **[Modify]** to display the ***Modify Button*** dialog box

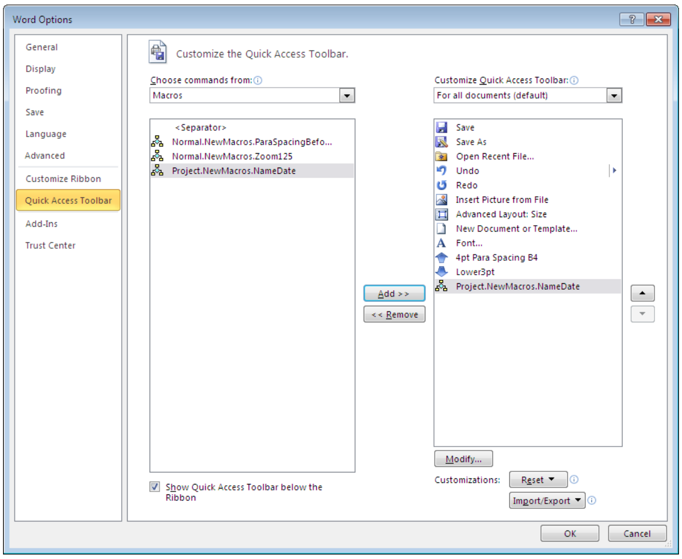
5 Click on an icon of your choice then edit the ***Display name*** so that it reads **Name and Date**, then click on **[OK]**

6 Click on **[OK]**

*The new button will appear in the Quick Access Toolbar. You can point to it to display the name of the macro...*

7 Click on ***Name and Date*** to run the macro

8 Delete the name and date text that was inserted when you ran the macro, then save the document



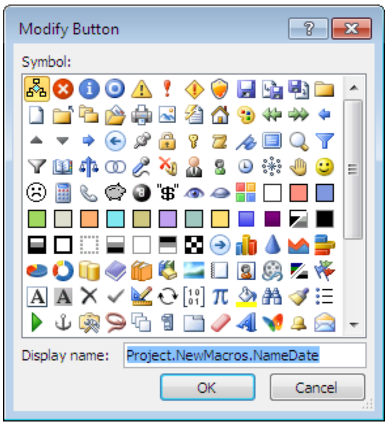
Choose commands from: Macros

Customize Quick Access Toolbar: For all documents (default)

☑ Show Quick Access Toolbar below the Ribbon

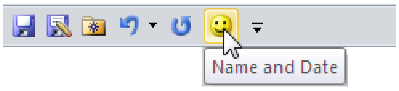
3

4



Display name: Project.NewMacros.NameDate

6



## For Your Reference…

To ***assign*** a ***macro*** to a ***toolbar***:

1. Click on the drop arrow for ***Customise Quick Access Toolbar*** and select **More Commands**
2. Click on the drop arrow for ***Choose commands from*** and select **Macros**

## For Your Reference (cont.)…

3. Click on the ***Macro name*** and click on **[Add]**
4. Click on **[Modify]** to change the name and/or icon
5. Click on **[OK]** and **[OK]**

# Assigning A Keyboard Shortcut To A Macro

Running a macro from the ***Macros*** dialog box or clicking on a macro tool on the ***Quick Access Toolbar*** may still take unnecessary time if you tend to work with your hands on the keyboard rather than on the mouse. You can assign a ***keyboard shortcut*** to a macro so that you can run the macro simply by pressing a combination of keys. This makes the process really quick.

## Try This Yourself:

*Same File*

*Continue using the previous file with this exercise, or open the file W881 Macros_6.docm...*

1 Right-click on the ***Quick Access Toolbar*** and select **Customise the Ribbon** to open the ***Word Options*** dialog box

2 Click on **[Customise]** for ***Keyboard shortcuts*** to open the ***Customise Keyboard*** dialog box

3 Scroll down the list of ***Categories*** and click on ***Macros*** then click on the drop arrow ▾ for ***Save changes in*** and click on ***W881 Macros_6.docm***

*The macros stored in this document will be listed under Macros...*

4 Click on ***NameDate*** to select it then click in ***Press new shortcut key***

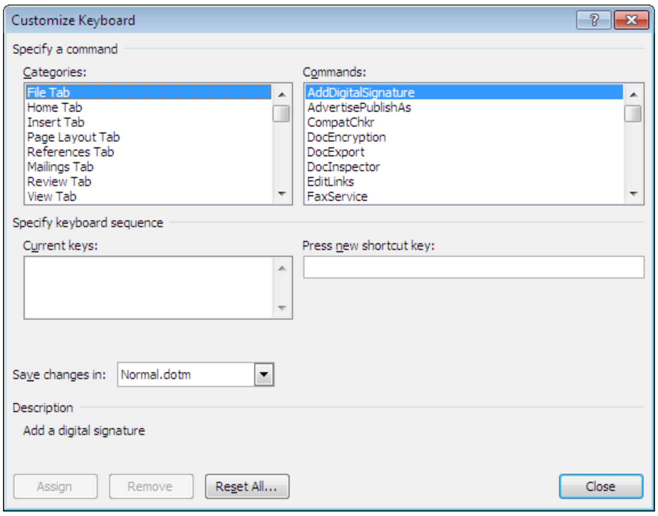
5 Press Alt + N then click on **[Assign]** to create the keyboard shortcut

6 Click on **[Close]** then **[OK]**

7 Press Alt + N to run the macro

8 Delete the name and date text that was inserted when you ran the macro, then save the document

2



Customize Keyboard

Specify a command

Categories:

File Tab
Home Tab
Insert Tab
Page Layout Tab
References Tab
Mailings Tab
Review Tab
View Tab

Commands:

AddDigitalSignature
AdvertisePublishAs
CompatChkr
DocEncryption
DocExport
DocInspector
EditLinks
FaxService

Specify keyboard sequence

Current keys:

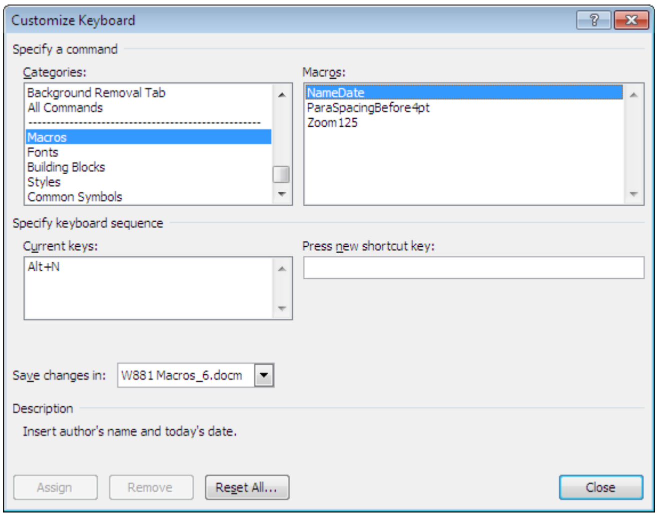
Press new shortcut key:

Save changes in: Normal.dotm

Description

Add a digital signature

5



Customize Keyboard

Specify a command

Categories:

Background Removal Tab
All Commands
--------------------------------
Macros
Fonts
Building Blocks
Styles
Common Symbols

Macros:

NameDate
ParaSpacingBefore4pt
Zoom125

Specify keyboard sequence

Current keys:

Alt+N

Press new shortcut key:

Save changes in: W881 Macros_6.docm

Description

Insert author's name and today's date.

## For Your Reference…

To ***assign*** a ***keyboard shortcut*** to a ***macro***:

1. Right-click on the ***Quick Access Toolbar*** and select ***Customise Quick Access Toolbar***
2. Click on **[Customise]** then scroll down the list of ***Categories*** and click on ***Macros***

## For Your Reference (cont.)…

3. Select the document containing the macro then click on the macro name
4. Click in ***Press new shortcut key***
5. Press the shortcut key combination
6. Click on **[Assign]** then **[Close]** then **[OK]**

# EDITING A MACRO

When you record a macro you actually create a series of commands in a programming language called ***Visual Basic for Applications*** (***VBA***). Each time you run the macro, the code is executed. VBA can be edited using the ***Visual Basic Editor*** that you access from the ***Developer*** tab. The advantage of using the editor is that you can easily change, delete or copy macro code.

## Try This Yourself:

***Same File***

*Continue using the previous file with this exercise, or open the file W881 Macros_7.docm...*

1 Click on the ***Developer*** tab, then click on ***Visual Basic*** to open the ***Microsoft Visual Basic*** editor

2 If the code isn't visible, double-click on ***Modules*** under ***Project (W881 Macros_7)*** then double-click on ***NewMacros***

3 Spend a few moments reading through each line of code – see if you can work out what each line does

4 Change the date code to **"dddd, d MMMM yyyy"**

5 Click on ***Save*** then click on ***File*** and select **Close and Return to Microsoft Word**

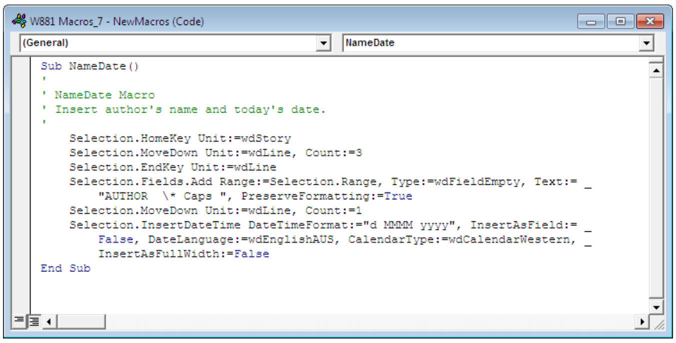
*Now to test it...*

6 Press Alt + N to run the macro

*Notice the new date format…*

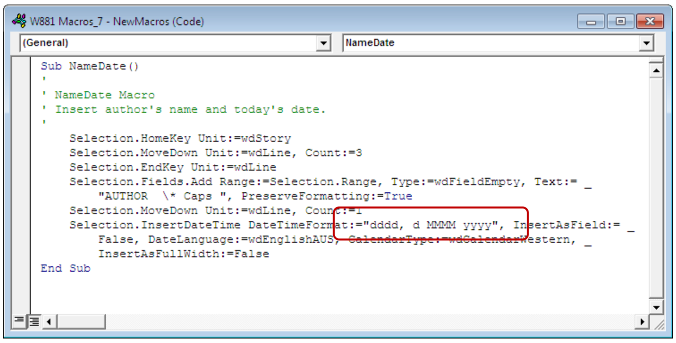
7 Delete the name and date text that was inserted when you ran the macro, then save the document

2



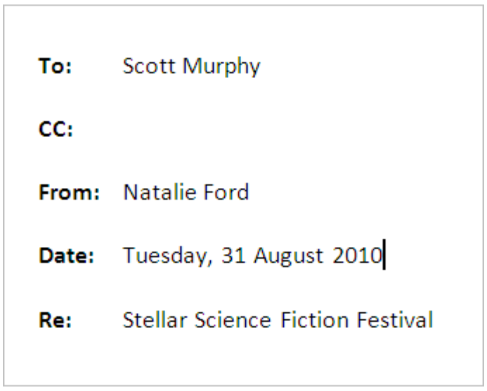
```
Sub NameDate()
'
' NameDate Macro
' Insert author's name and today's date.
'
    Selection.HomeKey Unit:=wdStory
    Selection.MoveDown Unit:=wdLine, Count:=3
    Selection.EndKey Unit:=wdLine
    Selection.Fields.Add Range:=Selection.Range, Type:=wdFieldEmpty, Text:= _
        "AUTHOR  \* Caps ", PreserveFormatting:=True
    Selection.MoveDown Unit:=wdLine, Count:=1
    Selection.InsertDateTime DateTimeFormat:="d MMMM yyyy", InsertAsField:= _
        False, DateLanguage:=wdEnglishAUS, CalendarType:=wdCalendarWestern, _
        InsertAsFullWidth:=False
End Sub
```

4



```
Sub NameDate()
'
' NameDate Macro
' Insert author's name and today's date.
'
    Selection.HomeKey Unit:=wdStory
    Selection.MoveDown Unit:=wdLine, Count:=3
    Selection.EndKey Unit:=wdLine
    Selection.Fields.Add Range:=Selection.Range, Type:=wdFieldEmpty, Text:= _
        "AUTHOR  \* Caps ", PreserveFormatting:=True
    Selection.MoveDown Unit:=wdLine, Count:=1
    Selection.InsertDateTime DateTimeFormat:="dddd, d MMMM yyyy", InsertAsField:= _
        False, DateLanguage:=wdEnglishAUS, CalendarType:=wdCalendarWestern, _
        InsertAsFullWidth:=False
End Sub
```

6



**To:** Scott Murphy

**CC:**

**From:** Natalie Ford

**Date:** Tuesday, 31 August 2010

**Re:** Stellar Science Fiction Festival

## For Your Reference…

To ***edit*** a ***macro***:

1. Click on the ***Developer*** tab then click on ***Visual Basic***
2. Double-click on ***Modules*** then ***NewMacros***
3. Make the changes
4. Click on ***Save*** then ***close***

## Handy to Know…

- You can also display the ***Visual Basic Editor*** by pressing Alt + F11.
- You can record macros by typing the code directly into ***Visual Basic Editor***. The more complex your macros, the more likely it is that you'll need to know how to write VBA code.

# DELETING A MACRO

Macros can tend to accumulate so it's good idea to remove any macros you don't need. This helps to maintain the integrity of the document, and also means that you don't have any redundant macro code hanging around. Macros should be deleted from the ***Macros*** dialog box. If you remove a macro button from a toolbar, all that happens is that the button is removed.

## Try This Yourself:

***Same File*** *Continue using the previous file with this exercise, or open the file W881 Macros_8.docm...*

1 Click on the ***Developer*** tab, then click on ***Macros*** to open the ***Macros*** dialog box

2 Click on the ***NameDate*** macro then click on **[Delete]**

*A confirmation dialog box will be displayed...*

3 Click on **[Yes]**

*The macro will be removed from the list...*

4 Click on **[Close]**

*Notice that the button for the macro still remains on the Quick Access Toolbar...*

5 Click on ***NameDate*** to see what happens

*A message box tells you that the macro either can't be found or has been disabled...*

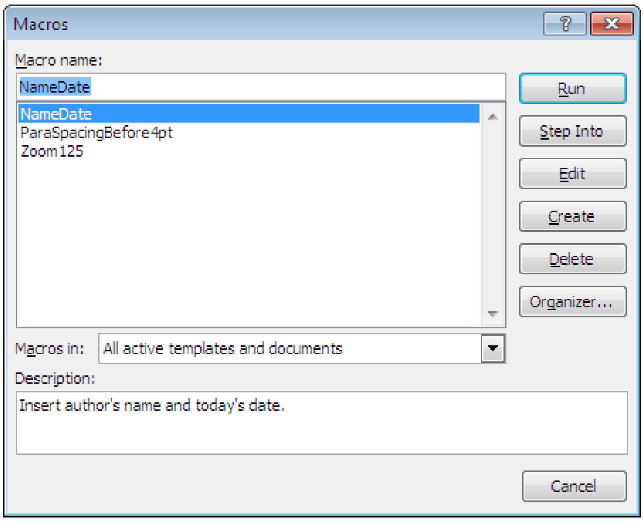
6 Click on **[OK]**

*Let's remove the button...*

7 Right-click on ***NameDate*** and select **Remove from Quick Access Toolbar**

8 Save and close the document

1



Macros

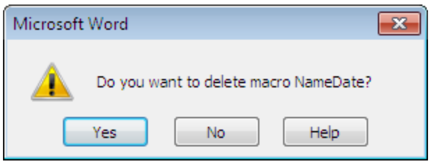
Macro name: NameDate

NameDate
ParaSpacingBefore4pt
Zoom125

Macros in: All active templates and documents

Description:
Insert author's name and today's date.

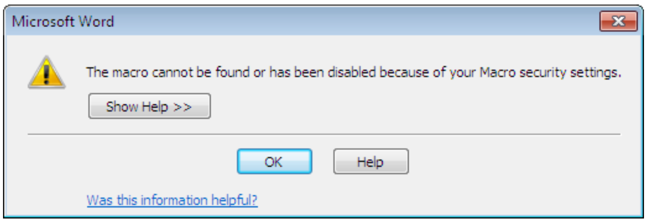
2



Microsoft Word

Do you want to delete macro NameDate?

5



Microsoft Word

The macro cannot be found or has been disabled because of your Macro security settings.

### For Your Reference…

To ***delete*** a ***macro***:

1. Click on ***Macros*** on the ***Developer*** tab
2. Click on the macro
3. Click on **[Delete]**
4. Click on **[Yes]** then click on **[Close]**

### Handy to Know…

- Effectively, to remove all traces of a macro requires two steps: one to remove the actual macro code via the ***Macros*** dialog box and one to remove the button from the ***Quick Access Toolbar*** (if one was added for this macro).

# CREATING A MACROBUTTON FIELD

The ***MacroButton*** field is great for running macros directly from text embedded in the document. This is ideal to help novice users enter text or other information in exactly the right place.

The ***MacroButton*** field requires the name of the macro to run and the prompt to display in the document. The macro runs when the user double-clicks on the prompt.

## Try This Yourself:

***Open File*** *Before starting this exercise you MUST open the file W881 Macros_9.docm...*

1 Click in the empty cell to the right of **From:** after the tab

*We will insert the prompt here...*

2 Click on the ***Insert*** tab, then click on ***Quick Parts*** and select **Field**

3 Click on the drop arrow for ***Categories*** and select **Document Automation**

4 Click on ***MacroButton*** then scroll down the list of ***Macro names*** and click on ***NameDate***

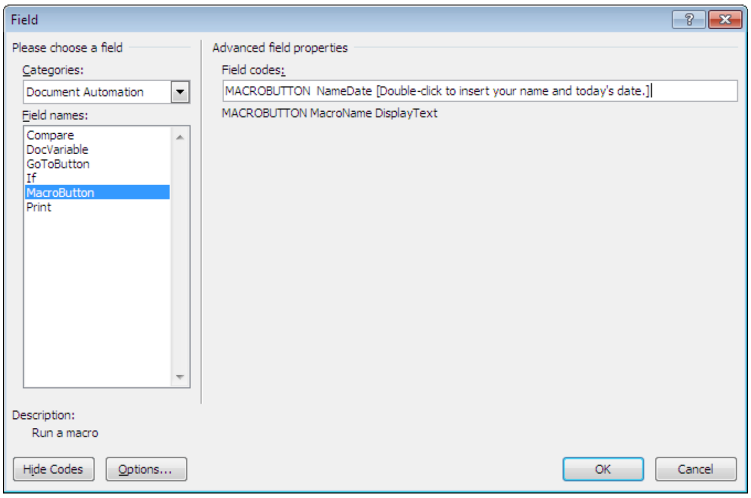
5 Click on **[Field Codes]** then click after ***NameDate*** and type **[Double-click to insert your name and today's date.]**

6 Click on **[OK]** to create the field

7 Double-click on the prompt to run the macro

*The prompt will be replaced with the author's name and the current date…*

8 Save the document

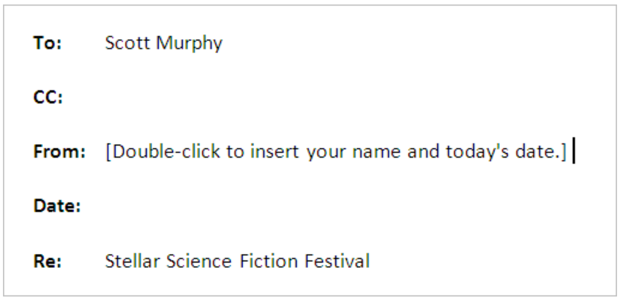


Field codes: MACROBUTTON NameDate [Double-click to insert your name and today's date.]

MACROBUTTON MacroName DisplayText

5

6



**To:** Scott Murphy

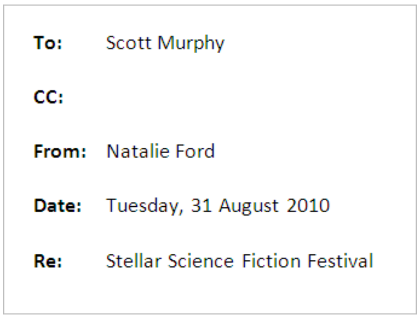
**CC:**

**From:** [Double-click to insert your name and today's date.]

**Date:**

**Re:** Stellar Science Fiction Festival

7



**To:** Scott Murphy

**CC:**

**From:** Natalie Ford

**Date:** Tuesday, 31 August 2010

**Re:** Stellar Science Fiction Festival

### For Your Reference…

To ***insert*** a ***MacroButton field***:

1. Click on ***Quick Parts*** and select **Field**
2. Select ***Document Automation*** in ***Categories*** then click on ***MacroButton***
3. Select a macro name, then click on **[Field Codes]**, type the prompt and click on **[OK]**

### Handy to Know…

- The ***MacroButton*** field doesn't actually display a button. It displays text that, when double-clicked, will run the assigned macro.
- Ideally, a macro such as this would be saved to a template, which opens as a blank document for users. This reduces the risk of users deleting macro fields.

# COPYING A MACRO

If you have recorded a macro and saved it in a particular document or template, you might like to ***copy*** the macro to another document or template so that it can be used elsewhere. Microsoft Word allows you to copy macro project items between files – as well as styles – using the ***Organiser***.

## Try This Yourself:

**Same File**

*Continue using the previous file with this exercise, or open the file W881 Macros_10.docm...*

1 Click on the ***Developer*** tab, click on ***Macros*** to open the ***Macros*** dialog box, then click on **[Organiser]**

2 Click on **[Close File]** under ***Normal.dotm*** then click on **[Open File]**

3 Click on **[All Word Templates]** and select **All Files**

4 Navigate to the course files folder and open ***W881 Macros_11.docm***

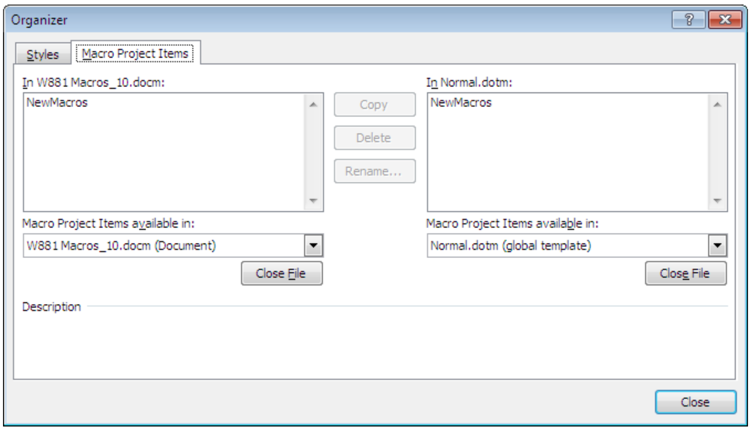
5 Click on ***NewMacros*** under ***W781 Macros_10.docm***

6 Click on **[Rename]** and type ***MemoMacros***

7 Click on **[OK]** then click on **[Copy ->]** to copy the macro project item

8 Click on **[Close]** then click on **[Yes]** to save the changes

9 Open ***W881 Macros_11.docm*** and click after ***From:*** then run the new macro from the ***Macros*** dialog box, then save and close both documents



In W881 Macros_10.docm:
NewMacros

In Normal.dotm:
NewMacros

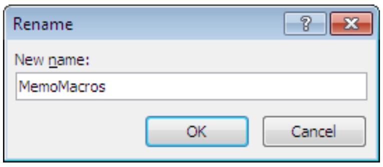
Macro Project Items available in:
W881 Macros_10.docm (Document)

Macro Project Items available in:
Normal.dotm (global template)

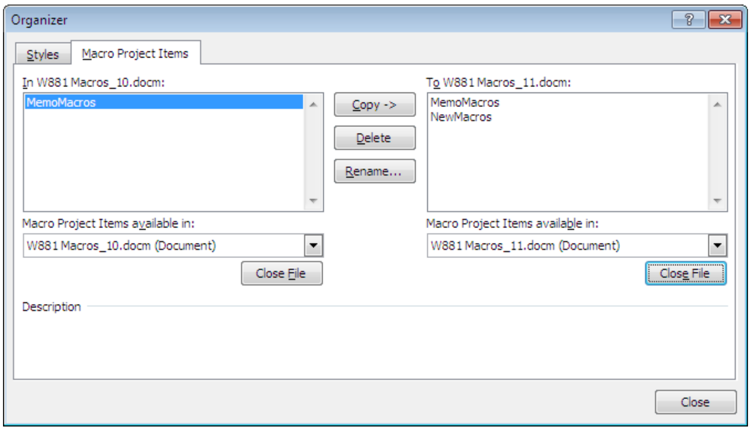
Description

1

6



New name:
MemoMacros



In W881 Macros_10.docm:
MemoMacros

To W881 Macros_11.docm:
MemoMacros
NewMacros

Macro Project Items available in:
W881 Macros_10.docm (Document)

Macro Project Items available in:
W881 Macros_11.docm (Document)

Description

7

## For Your Reference…

To ***copy macros between files***:

1. Click on ***Macros***, then on **[Organiser]**
2. Open the file with the macro in the left pane and the target file in the right pane
3. Rename the item if necessary then click on **[Copy]** then click on **[Close]**

## Handy to Know…

- You can't copy a macro project item if the target file has one with the same name already. Rename the item, then copy it.
- You can also access the ***Organiser*** by clicking on ***Document Template*** on the ***Developer*** tab.

# TIPS FOR DEVELOPING MACROS

If you spend a lot of time completing the same word-processing tasks over and over, macros can be an ideal solution to speed up document production. They are also a great way for users with limited knowledge to perform complex operations without assistance. It is important, therefore, that macros are well written. There are a number of ways to ensure that macros achieve this.

## 1 Test Your Macros Thoroughly

Macros are really just programs where you pit your wits against the operation of the computer. The macro will perform EXACTLY the way you specify – they have no initiative whatsoever. Macros should be tested several times to ensure that the same (correct) results are produced each time. Also, it is a good idea to test a macro on several computers as settings on the development computer can sometimes differ to those on other computers, which may affect the operation and performance of the macro.

## 2 Conform to Company Guidelines and Standards

Macros that don’t adhere to or even deliberately flaunt company standards and guidelines are doomed to failure. Before embarking on a new template ensure that you are familiar with the company’s standards for macro development.

## 3 Document Macros

It is most likely that a week from now you will have forgotten what the macro does and how it works. Even worse, the next person to do your job will have little idea of the existence, role, and operation of your macro unless you document what you have done. When you develop complex macros you should include comment lines in the macro code which explain what is happening, what you have done, and also indicate the current version and revision dates. All macro languages can be commented on without impacting on the performance of the macro.

## 4 Print a Copy of the Code

In Microsoft Word you can print a copy of the macro on paper. This is a handy way of checking the code and also serves as a useful archiving document. Macros can be printed from within the VBA editor using the **Print** command on the **File** menu.

## 5 Train Other Users

Train other users to use the macro correctly. If training is not possible produce a set of instructions or explanatory notes on the correct use of the macro.

# NOTES: