

Introduction.....	1
Default Formats, Menus, and Keystrokes Created Using Templates.....	2
Creating a New Template.....	2
Setting the Default Font.....	2
Menu and Key Assignments for Macros and Commands.....	3
Creating and Customizing Macros.....	3
Assigning Macros to Menus.....	5
Assigning Macros to Keys.....	5
Display Preferences.....	7
Word Processing Environment.....	7
Default Directory.....	7
Support for Graphic Formats.....	8
Macros.....	9
File Open File Extension.....	9
Proper Scrolling With Speller.....	10
Macro to change the Orientation on LaserJet Printer.....	11
Sample While EditSearchFound Looping Macro.....	11
Macro to Access Edit Summary Information From a Macro.....	12
Special Character Font Test.....	12
Macro to Save Selection to a New File.....	13
Inserting Filename with Full Path Information.....	14
Opening Last Document On Startup of Word for Windows.....	14

Introduction

Microsoft Word for Windows was created with many different types of users in mind. The program includes features that allow it to be modified to suit many personal word processing needs. This application note shows you how to customize:

- Default formats (including the default font)
- Menu and key assignments for macros and commands
- Display preferences (tabs, field codes, hidden text, etc.)
- The word processing environment (Autosave, typing replaces selection, etc.)

All modifications you make to these features are saved either in a template (such as NORMAL.DOT or LETTER.DOT) or in your WINWORD.INI file. Before making any changes to your word processing environment, you should make backup copies of the WINWORD.INI and NORMAL.DOT files. You can always go back to the original settings by restoring the old WINWORD.INI and NORMAL.DOT files. To create backups, with the directory containing Word for Windows active, type the following commands at the DOS prompt and press enter after each line:

```
copy winword.ini winword.bak
copy normal.dot normal.bak
```

If it becomes necessary to restore either of the original files, with the directory containing Word

for Windows active, type the following commands at the DOS prompt and press enter after each line:

To Modify	Use These Commands
WINWORD.INI	del winword.ini copy winword.bak winword.ini
NORMAL.DOT	del normal.dot copy normal.bak normal.dot

Default Formats, Menus, and Keystrokes Created Using Templates

A document template provides a pattern for shaping a final document. By setting up and using document templates, much time and effort can be saved. The following elements can be contained in a template:

- Boilerplate text -- text that is the same in every letter, memo, report, etc. This text is inserted automatically when you create a new document.
- Styles -- standard formatting for all of your paragraphs, including the fonts and type sizes you need to use.
- Glossary items -- standard text and graphics that you insert by typing the name of the glossary entry.
- Macros -- programs that allow you to change menus and key assignments to fit the type of document you are working on. Macros also cause Word to perform editing and formatting tasks for you automatically.

When working with default formats, menus, and keystrokes, you can choose to have the changes effected on a global basis (that is, for all documents) or effected only on documents based on a specific template. Global changes are saved in NORMAL.DOT. You can make template-specific changes to one of the sample templates provided with Word for Windows, or you can create a new template of your own.

Creating a New Template

Information about creating new templates can be found in the "Document Template" section on page 69 of the *Microsoft Word for Windows User's Reference*.

Setting the Default Font

The global default font is defined in the Normal style of the NORMAL.DOT template. The NORMAL.DOT template is used whenever another template has not been specified. To change the style of the default font, do the following:

***Note:** Before beginning this procedure, close any open documents or templates you have open in Word for Windows.*

1. From the File menu, choose New.
2. Select Document in the New box and select Normal in the Use Template box. Choose OK.
3. From the Format menu, choose Define Styles.
4. Select Normal in the Define Style Name box.
5. Choose Options. Select (turn on) Add to Template.
6. Choose Character. Select the default font characteristics you want, including font name, point size, color, etc.
7. Choose OK twice.

8. To save the changes, choose *Save All* from the *File* menu.
9. Word will ask, "Save changes to Document1?" Choose *No*.
10. Word will ask, "Save global glossary and command changes?" Choose *Yes*.

*This method of changing the default font in NORMAL.DOT can be used to change the fonts in any template. The template attached to the active document when *Format Define Styles* is selected will be modified. For more information about modifying styles, see pages 342-349 in the *Microsoft Word for Windows User's Reference*.*

Menu and Key Assignments for Macros and Commands

If no template has been specified, the NORMAL.DOT template, which contains the default settings, is used. The settings, styles, and glossaries in NORMAL.DOT are always available unless they are overridden by the settings, styles, and glossaries in an active template.

Creating and Customizing Macros

Word for Windows allows you to create new macros and to customize existing macros. The easiest way to write a macro is to record it using the macro recorder. For information about creating a macro, see "Recording a Macro" on page 186 of the *Microsoft Word for Windows User's Reference*. Use the keyboard (not the mouse) to select the steps you want to be recorded in your macro. (Using the keyboard records your actions more precisely than using mouse movements does. Mouse movements may use points on the screen that are not located in consistent locations.)

Most menu and keyboard commands in Word for Windows are actually Word for Windows macros. If you don't like the way a particular command works, you can modify the appropriate macro to suit your needs. The following examples modify the File Save and File Save As commands so that the Create Backup option is selected (turned on) by default. For more information on modifying existing macros, see "Macro Edit" on pages 187-188 of the *Microsoft Word for Windows User's Reference*.

1. From the Macro menu, choose Edit.
2. Select (turn on) Show All.
3. Select the FileSave (or the FileSaveAs) macro from the Edit Macro Name list box.
4. Choose OK.
5. Make the desired modifications to the macro (as shown in the appropriate table below).
6. When you are finished, choose Close from the File menu.
7. Word will ask, "Keep changes to Global: FileSave (or FileSaveAs)?" Choose Yes to save the changes.
8. When you exit Word, you will be asked, "Save global glossary and command changes?" Choose Yes.

Original FileSave macro:

Original FileSaveAs macro:

```
Sub MAIN
Super FileSave
End Sub
Sub MAIN
Super FileSaveAs
End Sub
```

FileSave macro modified to select the Create Backup option as the default:

FileSaveAs macro modified to select the Create Backup option as the default:

```
Sub MAIN
On Error Goto Finish
Dim FileDlg As FileSaveAs
GetCurValues FileDlg
FileDlg.CreateBackup = 1
If FileDlg.Name = "" Then
```

```
Else
```

```
End If
```

```
Finish:
```

```
End Sub
```

```
Sub MAIN
```

```
On Error Goto Finish
```

```
Dlg:
```

```
FileDlg.CreateBackup = 1
```

```
Dialog FileDlg
```

```
Super FileSaveAs FileDlg
```

```
Finish:
```

```
End Sub
```

More information on creating and modifying macros can be found in the *Microsoft Word for Windows Technical Reference*.

Assigning Macros to Menus

You may find there are certain task commands you can access only from the keyboard that you would prefer to access from the menus using the mouse. With Word for Windows you can add any macro command to any of the available menus by using the Assign To Menu command from the Macro menu.

For example, suppose you frequently work with large tables and often need to select an entire table. You could use a mouse to select the whole table by clicking on the top or bottom of the table and dragging to the other end of the table. Using the keyboard, you could place the insertion point anywhere in the table and press alt+5 (on the numeric keypad with num lock off) to select the entire table. Or, you could add the Select Table command to the Edit menu. To add a command to a menu, follow the steps outlined in the "Adding Commands or Macros to Menus" section on page 195 of the *Microsoft Word for Windows User's Reference*.

Note: You must have the Full Menus option selected in the View menu when you add a menu option. If the Short Menus option is selected, you cannot access the Macro menu or choose the Assign To Menu command. Added menu options will appear only when View Full Menus is selected.

You can also use the Assign To Menu command to remove seldom-used menu options from menus to keep the menus uncluttered. This procedure is outlined in the "Removing Commands or Macros from Menus" section on page 196 of the *Microsoft Word for Windows User's Reference*.

You can also reset all the menu options back to their original settings. To reset your menus back to their original settings:

1. From the Macro menu, choose Assign To Menu.
2. Choose Reset All.
3. Choose OK.

Note: Resetting your menus will reset ALL menu options, not just the most recently changed option.

Assigning Macros to Keys

In addition to assigning macros to menus, you can assign macros to key sequences, making Word for Windows quicker and easier to use. For example, perhaps you frequently switch in and out of draft mode. The current key combination (alt, v, d) is quick, but you could change the key combination to ctrl+a for better proximity and maximum speed. The following steps illustrate the procedure for assigning macros to keys, using the draft view example:

1. From the Macro menu, choose Assign To Key.
2. From the Assign Macro Name list box, select ViewDraft. If a macro already has an assigned key, the key is listed in the Current Keys box.
3. From the Context box, select Global or Template, as appropriate. If you are editing a document that

uses the Normal template, only Global will be available.

- Choose Template if you want the key sequence to be available only to documents using the current template.
 - Choose Global if you want the key sequence to be available to all documents.
4. Press ctrl+a. This sequence will now be listed in the Key box.
 - If ctrl+a is currently unassigned, "[currently unassigned]" will appear in the Key box.
 - If ctrl+a is currently assigned to another macro, that macro name will be listed in the Key box. If you accept a key sequence that is currently assigned to another macro, the key sequence will be reassigned to the ViewDraft macro.
 5. When you want to accept the key sequence listed in the Key box, choose Assign.
 6. Choose OK.
 7. When you exit Word, you will be asked, "Save global glossary and command changes?" Choose Yes.

You can also unassign most key sequences, whether they are defaults or not. For the procedure, see "Removing Key Assignments" on page 172 of the *Microsoft Word for Windows User's Reference*.

Display Preferences

Word for Windows allows you to set up your display with only the elements you want to see. For example, you may want to see paragraph marks and tabs but not spaces; or you may need to distinguish regular spaces from nonbreaking spaces. The options available are described in the "View Preferences" section on pages 266-267 of the *Microsoft Word for Windows User's Reference*. To select any combination of these options, do the following:

1. From the View menu, choose Preferences.
2. Select (turn on) the special characters and features that you want to see in the document window. For example, to show table gridlines, press alt+g (or click the check box next to Table Gridlines). An "x" will appear in the Table Gridlines check box. To turn gridlines off, press alt+g again.
3. Choose OK.

Word Processing Environment

Word for Windows allows you to customize your word processing environment. For example, you can automatically save your document at set intervals of time; change the units of measure to inches, centimeters, points, or picas; or change settings such as Background Pagination, Typing Replaces Selection, Prompt For Summary Info, or Your Name. These options are described in the "Utilities Customize" section on pages 53-54 of the *Microsoft Word for Windows User's Reference*. To select any combination of these options, do the following:

1. From the Utilities menu, choose Customize.
2. From the Autosave Frequency box, select the interval you want; select one of the options in the Unit of Measure box; and select the options you want to turn on (Background Pagination, Prompt For Summary Info, Typing Replaces Selection).
3. Choose OK.

Default Directory

Word for Windows defaults to the directory from which Word for Windows was executed. For example, if you start Word for Windows from C:\WINDOWS, the default directory is C:\WINDOWS. To change the default to a different directory, do the following (this procedure works only if you are using Windows version 3.00):

1. In the Program Manager, select (click once) the Word for Windows icon.
2. From the File menu, choose Properties.
3. In the Command Line box, type the full path to the directory you want to be the default, and to

that path, append "WINWORD.EXE". For example, if you want C:\DATADOC to be your default directory, type "c:\datadoc\winword.exe" (without the quotation marks) and choose OK.

4. When the error message "The specified path is invalid" appears, choose OK.
5. From the File menu, choose Properties.
6. Choose the Change Icon button.
7. Type in the real path to the Word for Windows executable file (WINWORD.EXE) and choose OK. Choose OK again.
8. When the error message "The specified path is invalid" appears, choose OK.

After you perform this procedure, when Word for Windows is loaded and you choose Open from the File menu, the C:\DATADOC directory will be accessed. Also, all files will be saved to this directory unless you specify a different directory.

Another possible means to change the default directory, would be to use a Word for Windows AUTOEXEC macro to change the current directory on startup of the Word for Windows program. To do this choose Edit from the Macro menu and type the word "autoexec" (without quotes) in the macro name box and click on the OK button. The resulting macro editing screen will appear as follows:

```
Sub MAIN
```

```
End Sub
```

Between the Sub MAIN and End Sub statement use the CHDIR command to change the directory, as in the following example:

```
Sub MAIN
```

```
CHDIR "C:\WINWORD\DATADOC"
```

```
End Sub
```

Any path on your hard disk can be indicated between the quote characters ("pathname"). The path indicated between the quotes will be the default path when the program is started.

Choose Close from the File menu and choose YES when prompted to "Keep changes to Global: Autoexec?".

Support for Graphic Formats

The following are the supported graphics file formats in Word for Windows 1.10a:

(ADI) AutoDesk's AutoCAD Device-Independent Binary Plotter Format
(BMP) Microsoft Windows Paintbrush Bitmap Graphics File Format
(CGM) Computer Graphics Metafile
(EPS) Encapsulated PostScript
(HPGL) Hewlett-Packard Graphics Language
(IMA) Zenographics Mirage Graphics File Format
(PCX) ZSoft (Microsoft Windows Paintbrush) Graphics File Format
(PIC) Lotus PIC Graphics Format
(PIC) Micrografx Draw! Graphics Format
(PIC) Video Show Graphics Format
(TIFF) Tagged Image File Format (Specification 4.0 and 5.0)
(WMF) Windows Metafile Format

New to Word for Windows 1.10a

- Support for color TIFF
- a new EPS graphics filter
- support for color bitmaps (DIBS) via either the Windows Clipboard or the Insert Picture command
- Color PCX

Macros

File Open File Extension

You can change the FileOpen macro in Microsoft Word for Windows to show all files in the Files listbox by editing the macro to use a search mask of *.* instead of *.DOC. You can specify any search mask desired.

To edit the macro, do the following:

1. From the Macro menu, choose Edit.
2. Check the ShowAll box.
3. Type FileOpen in the Edit Macro Name box.
4. A macro window will be displayed showing the current macro, which is shown below.

```
Sub MAIN
Dim dlg As FileOpen
GetCurValues dlg
Dialog dlg
Super FileOpen dlg
End Sub
```

5. Alter the macro to include a new fourth line

```
dlg.Name = "*.*)"
```

as shown below.

```
Sub MAIN
Dim dlg As FileOpen
GetCurValues dlg
dlg.Name = "*.*)"
Dialog dlg
Super FileOpen dlg
End Sub
```

6. Close the macro window saving changes to Global: FileOpen.

7. When you close Word for Windows and you are asked, "Save global and glossary changes?", answer yes.

Now each time you choose the FileOpen command, the File listbox will show all files in the current directory.

Since you have edited a built-in macro, it is now considered a user-defined macro. When you choose Cancel in the dialog box, Word for Windows will display the message "Word Basic Error = 102 Command Failed." This is proper behavior for user-defined macros.

If you want to avoid having this message displayed, insert the line "On Error Goto <label>" as the second line of the FileOpen macro. Also place the <label> above the End Sub line of the macro. This will prevent the error message from being displayed.

With the change to show all files and the error handling to trap the Word BASIC 102 errors generated by selecting the Cancel button, your FileOpen macro should look as follows:

```
Sub MAIN
On Error Goto Done
Dim dlg As FileOpen
GetCurValues dlg
dlg.Name = "*.*)"
Dialog dlg
Super FileOpen dlg
Done:
End Sub
```

Proper Scrolling With Speller

A document may not scroll properly to show misspelled words in context when you check spelling in

Galley or Page view. It will scroll properly, however, if you check spelling in Draft view.

The built-in UtilSpelling macro can be modified to save the current view, change to Draft view, check spelling, and change back to the original view. This will allow the page to always scroll while checking spelling.

Use the following steps to modify the macro:

1. From the Macro menu, choose Edit.
2. Type "UtilSpelling" (without the quotation marks) in the Edit Macro Name box and press ENTER.
3. Modify the macro as follows:

```
Sub MAIN
fViewDraft = Abs(ViewDraft())
fViewPage = Abs(ViewPage())
fViewOutline = Abs(ViewOutline())
On Error Goto done
ViewDraft 1
Dim dlg As UtilSpelling
GetCurValues dlg
Dialog dlg
done:
ViewPage fViewPage
ViewDraft fViewDraft
ViewOutline fViewOutline
End Sub
```

4. From the File menu, choose Save All.
5. When asked to save Global Glossary and Command changes, select Yes.

Macro to change the Orientation on LaserJet Printer

The following macro allows you to change the orientation of the printer setup from portrait to landscape using the HP/PCL driver version 3.20:

Macro to change orientation to landscape on LaserJet printer.

```
Sub Main
SendKeys "%l{enter}"
FilePrinterSetup .Printer = "PCL / HP LaserJet on LPT1:", .SETUP
FilePrinterSetup .Printer = "PCL / HP LaserJet on LPT1:"
End Sub
```

Changing the "l" to an "r" changes the orientation from landscape to portrait. You can modify this

macro to select the orientation for other printers by changing "PCL / HP LaserJet on LPT1:" to the printer you want to use. To customize the key that is selected, check the printer dialog box for the underscored character that is used with the ALT key (indicated as "%" with SendKeys) to make that selection.

Sample While EditSearchFound Looping Macro

The following macro can be used to search for the occurrence of text within a document, and when found change the formatting. The macro below will prompt for the search text and change the formatting on the search text to Bold. After scanning the document, a message box will post the number of times the search text was found.

```
Sub MAIN
  StartOfDocument
  counter = 0
  searchtext$ = InputBox$("Enter text to search for?")
  EditSearch .Search = searchtext$, .WholeWord = 1, .MatchCase = 0, \
  .Direction = 2, .Format = 0
  FormatCharacter .Bold = 1      'Line to apply formatting
  If Not EditSearchFound() Then
    MsgBox "Search Text Not Found"
    Goto bye :
  End If
  While EditSearchFound()
    counter = counter + 1
    EditSearch .Search = searchtext$, .WholeWord = 1, .MatchCase = 0, \
    .Direction = 2, .Format = 0
    FormatCharacter .Bold = 1      'Line to apply formatting
  Wend
  A$ = Str$(counter)
  MsgBox "Search text was found" + A$ + Chr$(160) + "time(s)"
  bye:
End Sub
```

More Information:

The counter variable is used to count the number of times the search text was found (counter = counter + 1). If the search text is not found a message box will appear posting a message that the search text was not found. If the text is found, the formatting for the text is changed to bold and the counter is incremented by 1. Any formatting command or no formatting command can be used in place of the line that reads "FormatCharacter .Bold = 1."

Macro to Access Edit Summary Information From a Macro

It is possible to access the information in the Edit Summary Information dialog from a macro and post

the information in a message box or insert the information into the active a document. The following sample macro will display the number of words, characters and pages in the active document.

```
Sub MAIN
EditSummaryInfo .Update
Dim dlg As Dialog EditSummaryInfo
GetCurValues dlg
A$ = dlg.NumWords
B$ = dlg.NumChars
C$ = dlg.NumPages
MsgBox "Words =" + A$ + Chr$(160) + "Characters = " + B$ + Chr$(160) \
+ " Pages =" + C$
End Sub
```

Additionally, it is possible to use a similar macro to insert the contents of fields within the Edit Summary Dialog at the cursor position in the document. The following macro will insert the Title, Subject, and Author text as it appears in the Edit Summary Information dialog.

```
Sub MAIN
EditSummaryInfo .Update
Dim dlg As Dialog EditSummaryInfo
GetCurValues dlg
A$ = dlg.Title
B$ = dlg.Subject
C$ = dlg.Author
Insert "Title:" + Chr$(160) + A$ + Chr$(13)
Insert "Subject:" + Chr$(160) + B$ + Chr$(13)
Insert "Author:" + Chr$(160) + C$ + Chr$(13)
End Sub
```

Special Character Font Test

Different font sets have different extended characters (PostScript fonts for example). Although they look the same as the ANSI set on the screen, they will print differently.

The following macro will generate a five-column document of all of the extended characters for a given font. One can then print the document after formatting it for the chosen font. The printout will show the code and symbol you get with this font.

1. From the Macro menu in Word for Windows, choose Edit.
2. Type the following as the macro name:

Chartest

The following commands can be entered below Sub Main:

```

FormatSection 5
i = 33
While i < 256
val$ = Str$(i)
Insert Chr$(i)
Insert Chr$(32)
Insert Val$
Insert Chr$(11)
i = i + 1
Wend

```

This macro generates characters 33 through 256 of the ANSI set.

3. Close the Chartest macro and save changes when prompted.
4. To run the macro open a new document and choose Run from the Macro menu. Select Chartest from the list of macro and click on the OK button.
5. Highlight the whole document with CTRL+5 on the numeric keypad and choose any font available under Format Character. Print the document to evaluate the available extended characters and the associated numbers.
6. To later print one of the characters in a document, press ALT and enter the associated number preceded by a zero, ALT +0125 (use the numeric keypad with NUM LOCK on). The character will need to be formatted for the same font as in the chartest printout.

Macro to Save Selection to a New File

The following macro will copy whatever text is highlighted in a Word for Windows document to a new file and then prompt with the File Save dialog.

```

Sub MAIN
A$ = Selection$()
If SelType() = 1 Then
MsgBox "Nothing is selected"
Goto finish
End If
Super EditCopy
FileNew .NewTemplate = 0, .Template = "normal"
Super EditPaste
On Error Goto finish
Super FileSave
finish:
End Sub

```

If the selection is less than one character (nothing is selected), a message box will display the message "Nothing is selected". If the selection one character or more, the selected text will be copied to a new

file based on the "normal" template.

Inserting Filename with Full Path Information

The {filename} field will insert the name of the document without path information, i.e.

Field Code

The name of the file is {filename}

Field Result

The name of the file is README.DOC

The following macro can be used to insert the name of the document and the path information for the file.

```
Sub MAIN
EditSummaryInfo .Update
Dim dlg As Dialog EditSummaryInfo
GetCurValues dlg
A$ = dlg.Directory
B$ = dlg.FileName
Insert A$ + "\" + B$
End Sub
```

For example:

The name location of the file is C:\WINWORD\README.DOC.

More Information:

In order to determine the document name and directory, the macro accesses the EditSummaryInfo dialog and extracts the filename and directory information. The directory and the filename are added together with a backslash between them with the following macro line:

```
Insert A$ + "\" + B$
```

The path and filename (C:\WINWORD\README.DOC) are inserted at the position of the cursor.

Opening Last Document On Startup of Word for Windows

To load the last file edited in Word for Windows, choose File and select the file listed after the number 1 at the bottom of the menu. You can load up to the last four files in this manner. Each of the last four files is listed at the bottom of the menu after the appropriate number.

To automatically load the last document edited when you start Word for Windows, use one of the following methods:

FIRST METHOD:

Choose Edit from the Macro menu and type AUTOEXEC in the Edit Macro Name box, and select Global in the context box. Press ENTER or click OK. On the blank line after Sub Main, type "File1" (without quotes). From the File menu, choose Save All. When the dialog box asks to keep changes to Global:AUTOEXEC, press "Y" or click Yes. Do the same for the dialog box asking to save global glossary and command changes. From the File menu, choose Exit.

To prevent the AUTOEXEC macro from running when starting, use the following command:

```
winword /m
```

ALTERNATIVE METHOD:

Use the command line "winword /mfile1" to start WinWord with the last document edited. The parameter /m tells Winword to run a macro at startup and file1 is the name of the macro to open the last document edited (there is not space after the "/m" switch).

file1 can be replaced with:

to open up the 2nd 3rd or 4th documents edited last in Word for Windows.

Note:

Using run-time Windows, type the desired command at the DOS prompt. If using Windows 3.0, use the File Run command to enter the desired command. If the full version is Windows 3.00, the desired command may be entered in the command line of the Word program icon by choosing Properties from the File Menu in Program Manager.