

WordPerfect® for Windows® Macro Tutor

CHAPTER PREVIEW

See <http://gmccomb.com/tutor/> for more information

Table of Contents

Chapter 1: What Macros Are All About

- Macros Are Shortcuts
- Where the Word “Macro” Comes From
- Macros Can Be Recorded or Programmed
- Knowing When to Use Macros
- Should You Use a Macro, or Some Other Feature?
- Other Features That Are Helpful
- Teaming Up the Automation Features of WordPerfect

Chapter 2: Recording and Playing Macros 1

- Getting Ready for Recording
- Recording a Macro
- Playing a Macro
- If You Make a Mistake While Recording
- Giving Names to Macros
- Avoiding Common Macro Problems

Chapter 3: Different Ways to Play Macros

- Playing the Last Macro
- Adding a Macro to the Toolbar
- Assigning a Macro to a Menu
- Assigning a Macro to the Keyboard

Chapter 4: Editing and Printing Your Macros

- Recording a Sample Macro
- Opening the *Italsent.wcm* Macro
- Understanding the Macro Property Bar
- A Brief Discussion of the Innards of a Macro
- Watch Out For Mistakes!
- Adding a Comment to the Macro
- Saving Your Changes
- Printing a Macro
- Viewing Macros Without Opening Them
- Viewing and Printing the “Shipping Macros”

Chapter 5: Important Macro Concepts

- Commands: Macros Are Made of These
- Understanding Syntax
- Entering Commands
- Using Parameter Names
- Using the Macro Tools Toolbar

The Concept of Compiling
Using Macros in Templates
Getting Help on Macros
What About Visual Basic for Applications Macros?

Chapter 6: Creating Macros That Pause for User Input 9

Creating a Fax Cover Sheet with Pauses
Creating a Letter Opener Macro and Adding Pauses
Using a Different Key/Action to Unpause the Macro
Telling the User What to Do When a Macro Is Paused

Chapter 7: How to Use Variables

What Is a Variable
Creating and Using Variables
Using Variables To Collect Lots of Data
Saving Space with the BARLAB2.WCM Macro
Building a More Elaborate Variable Collection Macro
Stuff You Can Store in a Variable
Assigning Variables Within a Macro

Chapter 8: Getting Information from the User

GetString: The All-purpose Data Picker-upper
Using GetNumber to Store Number Values
Using GetUnits to Store Measurement Values
Displaying a Menu of Choices

Chapter 9: Asking WordPerfect What It's Doing: All About System Variables

What Is a System Variable
List of Commonly Used System Variables
System Variable Sampler Macro
A Practical Macro for Inserting the Filename

Chapter 10: Inserting Information Into a Document

Inserting Text: An Overview of the Type Command
Positioning the Insertion Point
Selecting Text in the Document
Removing Text Already in the Document
Working Inside Tables
Sample Text for the Test Macros in This Chapter

Chapter 11: Displaying Messages to the User

In Review: Using MacroStatusPrompt
Using the Prompt Command
Using the MessageBox Command
Message Example: Saving All Documents
Message Example: Showing the Time

Chapter 12: Controlling the Flow of Macros

- Creating “Zones” In Your Macros
- Referencing a Label
- Using Labels for “Condition Trapping”
- Call Example: A Macro to Select Printing Options

Chapter 13: Macros That Make Decisions: Using the If Command

- The Anatomy of the If Command
- Commands for True Or False, But Not Both
- Getting to Know Operators
- Real World If: The TWOKEY.WCM Macro
- Nesting If's
- Testing More than One Condition

Chapter 14: Other Ways to Make Decisions: Using Switch

- The Makeup of a Switch Structure
- Providing for a Default Result
- Testing for Numeric Matches
- Using Multiple Commands for a CaseOf Match
- Using Switch with Menu
- Getting Fancy with the Continue Command
- Going or Jumping to Other Parts of the Macro
- A Practical Example of the Switch Statement

Chapter 15: Over and Over Again: Making Macros Repeat Commands

- Why and When to Use Loops
- Using ForNext
- Using ForEach

Chapter 16: More of the Same: Other Ways to Repeat

- Using the While Command
- Using A Loop with a Counter
- Using Repeat/Until
- Using Repeat/Until with Searches
- Using Repeat/Until to Assure Proper User Input

Chapter 17: Working with Text Strings

- A String is Just a Collection of Characters
- Inserting a String into a Document
- Counting the Length of a String
- Retrieving Just a Portion of a String
- Determining the Location of a Given Character
- A Practical Example of StrPos and SubStr
- Converting String Case

Chapter 18: Creating Custom Dialog Boxes

- Accessing the Dialog Editor
- Creating a New Dialog Box
- Adding a Control
- Setting Control Properties
- Resizing and Positioning the Dialog Box
- Testing the Dialog Box
- Saving the Dialog
- Exiting the Dialog Editor
- Enhancing the Basic Dialog Box
- Connecting the Dialog with the Rest of the Macro

Chapter 19: “How Do I...?”

- Play a Macro When Starting WordPerfect
- Display the Files In a Specific Directory
- Check If a File Exists
- Delete a File On Disk
- Copy a File on Disk
- Close All Files At Once
- Save All Files At Once
- Save a File With a Random Name
- Specify a Directory, Name, and Format for Saved Files
- Save a File With a New Name and Delete the Original
- Append the Network User Initials To All Saved Files
- Check if a Document Is “Blank”
- Close or Clear a Document Without Saving
- Save a Document Then Close
- Insert the Document Name in the Document
- Check If The Insertion Point Is at the End of the Document
- Check If The Insertion Point Is at the Beginning of the Document
- See If One Or More Characters Is in a String
- Compare Two Strings Without Regard to Case
- Run Programs from Within WordPerfect
- Create a Temporary “Scratchpad” to Write Text
- Ask the User a Question
- Limit the Length of Text for the GetString Command
- Limit Entry To Numbers Only
- Accept Upper or Lower Case Responses to a Prompt
- Display a Prompt in the Status Bar
- Make the Macro Wait
- Ring the Bell
- Recover From Failed Searches
- Recover from Errors
- Recover from the Cancel Key
- Ignore Not Found, Errors, and Cancels
- Share Variables Between Macros
- Transferring Document Text to a Macro Variable

Change the Case of the Next Character

Appendix A: Using the Applications Archive

Index..... 199

You need to be aware of a few other aspects of WordPerfect's operation as it relates to macros, but we'll save these for later in the chapter. For now, let's get to the meat of the chapter and discuss recording a macro.

Recording a Macro

For a sample macro, we'll record one that toggles the case of the current character — that is, if the current character is lower case, the macro makes it upper case, and vice versa. This obviously requires that there be text already in the document. So, begin by opening a new, blank document by choosing File, New. Type *wordperfect*, with the capitalization shown. For example purposes, position the insertion point immediately before the *w*. You may now record the macro.

The first step in recording a macro is to turn on the macro recorder:

1. Choose Tools, Macros, Record. The Record Macro dialog box appears as shown in Figure 2.1.

You will notice this box may contain a listing of files already on your computer. These are the macros that came with WordPerfect when you installed it.

2. In the *File name* box, type *Chartog*, for “character toggle” (you can type the name in upper or lower case), and choose Record.

Note that the Macro Record indicator appears in the Application Bar at the bottom of the WordPerfect window. Notice also that when the mouse pointer is over the document window it turns into a slash-in-circle. This denotes that you cannot use the mouse within the document area while recording a macro.

3. Hold down the Shift key and press the right arrow key. This selects the current character, which is *w*.
4. Press Ctrl+K. This toggles the case of the character — *w* becomes *W*.
5. Press F8. This turns selection off. These are all the steps necessary for the macro.
6. Turn macro recording off by choosing Tools, Macros, Record. The Macro Record indicator disappears, and macro recording stops.

You've just created your first macro!

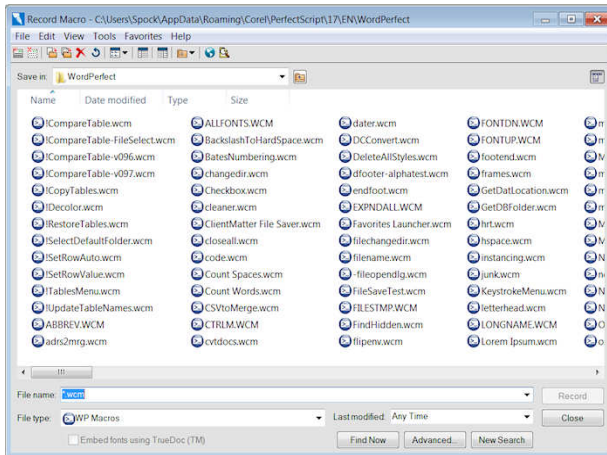


Fig 2.1. In the Record Macro dialog box give the macro a name, then choose Record.

The idea behind macros is that they save you time. So it makes sense to take advantage of the shortcuts built into WordPerfect to record a macro. Rather than choose the Tools, Macros, Record command to turn macro recording on and off, merely press Ctrl+F10 instead.

Playing a Macro

Once a macro is recorded you can play it back any number of times. As an example of playing a macro, let's play back the *Chartog* (character toggle) macro recorded in the previous section. Before playing the macro, prepare WordPerfect by placing the insertion point immediately before the *p* in *Wordperfect*.

1. Choose Tools, Macro, Play. The Play Macro dialog box appears.
2. The Play Macro dialog box looks just like the Record Macro dialog box, except it has a button marked Play instead of Record. Pick the macro you want to and Choose Play. WordPerfect plays the *Chartog* macro, and the *p* becomes a *P*.

As with recording a macro, WordPerfect offers a keyboard shortcut to play a macro. Pressing Alt+F10 is the same as choosing Tools, Macro, Play.

Feel free to experiment with the *Chartog* macro to see how it works with other characters.

If You Make a Mistake While Recording

Nobody's perfekt...er, perfect. Mistakes are common when recording a macro. Only for some strange psychological reason, making a mistake while the macro recorder is running causes normally calm people to burst into nervous hysterics. The macro recorder is on, and

so is the critical eye of Joe Friday. Any mistake you make can and will be used against you in a court of law — or so the feeling goes.

Relax. Mistakes are easy to fix. If you find you've made a mistake, you have several options:

- Stop recording and start over. Merely press Ctrl+F10 to stop the recorder. Reset the text and formatting of the document (if necessary), and turn the macro recorder back on. Give the new macro the same name as the first. WordPerfect will warn you that a macro with the same name already exists. Choose Yes, you wish to replace it.
- Fix the mistake, and continue as if nothing happened. For example, if you make a typographical error, backspace over the mistake, and type the correction.

Truth be told, depending on how soon you realize you made a mistake, WordPerfect doesn't even record most boo-boos. For example, record the following actions:

1. Type "this is a teast" (do not press the Enter key).
2. Press Backspace to delete the *ast*.
3. Type *st*, and press Enter. The text in the document should read "this is a test."

Even though you corrected yourself, the macro actually recorded "this is a test" (in Chapter 4 you will learn how to view a macro file and look at the commands that were recorded). As long as you don't press the Enter key, you may make corrections and WordPerfect will record only the final text.

The same holds true for dialog boxes. If you make a mistake in a dialog box — you choose the wrong options, for instance — just fix the problem before you choose the OK button to close the dialog box. With *very few* exceptions, WordPerfect will only record the final setting.

Giving Names to Macros

The *Chartog* macro described in the previous sections is stored in a separate file. Unless you specify otherwise, WordPerfect automatically appends a .WCM extension to the end of the file. The actual file, as stored on your computer's hard disk, is called *Chartog.wcm*.

WordPerfect is quite flexible when it comes to naming macros. Any valid Windows filename will do, though I personally advise that you stick with the .wcm extension. Doing so makes it easier to play your macros, as the .wcm extension is assumed. When playing your macros, you don't need to type the .wcm extension if your macros are so named; you *do* have to type the full filename if you use another extension.

When recording macros, try to give them meaningful names. For example, for a macro that sets the margin to two inches, you might name it *MARG2*. Make the name long enough so you'll understand its purpose weeks or months after you've recorded it, but try to avoid giving macros long (say, more than 10 or 12 character) names, as doing so simply means more typing to play the macro each time.

If you plan on using a macro quite frequently, you can give it a "shortcut" name, using the Ctrl or Ctrl+Shift keys. You play them back by pressing the Ctrl key and a letter key (or Ctrl, Shift, and a letter key). We'll call these "quick access" macros. As you can see in the following table, a number of key combinations are already taken by WordPerfect's built-in shortcuts. However, a couple of openings are available for use in giving shortcut names to your quick access macros.

Key	Pressing Ctrl+Key	Pressing Ctrl+Shift+Key
A	Select all	QuickWords <i>insert</i>
B	Bold	Insert bullet
C	Copy text	Dropcap start
D	Date text	Date code
E	Center justification	<None>
F	Find/replace	Find next
G	Go to	<None>
H	Outline body text	Print history dialog
I	Italic	<None>
J	Full justification	<None>
K	Case toggle	<None>
L	Left justification	Line break
M	<None>	Document map
N	New document	New from project
O	Open document	Outline define
P	Print	Print document
Q	Find QuickMark	Set QuickMark
R	Right justification	Document map focus
S	Save document	Save all
T	New from project	<None>
U	Underline	<None>
V	Paste	Paste simple
W	Insert symbol	<None>
X	Cut	<None>
Y	<None>	<None>
Z	Undo	Redo

As you can see, Ctrl+M and Ctrl+Y are not otherwise taken by a built-in WordPerfect shortcut, so you can use these to name a quick-access macro.

To give a macro a Ctrl+key name, name the macro *Ctrlx.wcm*, where x is a letter or number. Conversely, to give a macro a Ctrl+Shift+key name, name the macro *Ctrlstfx.wcm*. For example: *Ctrlstfm.wcm* is played by pressing the Ctrl+Shift+M keys at the same time.

You don't need to use the Play Macro dialog box to play back a quick access macro. Rather, simply press the appropriate shortcut keys — Ctrl+M, for example.

There are yet other shortcuts to playing macros, including adding a macro to the Macro pop-up menu, to a button on the Toolbar, and to most any key or key combination. Read about these techniques in Chapter 3, "Different Ways to Play Macros."

While the Ctrl+M key combination is not defined as a WordPerfect built-in shortcut, later versions of WordPerfect come with a *ctrlm.wcm* macro. This macro is intended for activating a feature commonly used when writing macros. Rather than overwrite this existing macro with your own, rename it first (maybe to *ctrlm-old.wcm*), in case you'd like to use it later on.

Avoiding Common Macro Problems

Like all things in life, macros aren't perfect. Here are some of the most common problems you will encounter as you record and play your macros, and how to avoid them.

- Use only the keyboard when working inside the document
- Avoid differences in the "context" of WordPerfect
- Remember where your macros are stored

Using the Keyboard When Working Inside the Document

One limitation to always remember: macros cannot record mouse actions within the document area. During macro recording, placing the mouse anywhere inside the document area of the WordPerfect window makes the mouse pointer turn into a slash-in-circle. WordPerfect is saying, "don't do anything with the mouse here, because I won't record it." (Note that you can still use the mouse to select menu commands during macro recording.)

The reason for this is actually quite logical. When you record a macro, WordPerfect needs to be told *exactly* what to do. It needs absolute directions of what text to remove, which paragraph to make bold. If you use the mouse to select text or to move the insertion point, WordPerfect will have no idea where it's at in the document. This could cause erratic behavior when playing back the macro, so it is not supported.

During macro recording, if you need to move the insertion point, or select text, you must do so using only the keyboard. Many users of WordPerfect are not as familiar with the keyboard commands used for moving the insertion point or selecting text as they are with using the mouse for these tasks. But in order to become proficient at recording and using macros, you need to acquaint yourself with WordPerfect's keyboard commands. A list of keyboard commands is available by searching for the phrase 'Using WordPerfect shortcut keys' in the on-line help.

Avoid Differences in the “Context” of WordPerfect

WordPerfect has many “contexts,” also called states. In non-macro use, the contexts of WordPerfect are not of critical importance. Not so when you add macros to the equation. If you use macros, you must remember to play the macro within the same context used to record the macro.

In the macro described earlier in this chapter, *Chartog*, you first need to type some text, and position the insertion point before the character you want to case toggle. This is required because this is how the macro was recorded.

As an experiment, try to play the *Chartog* macro in a blank document, with no text. What happens? Because the context is different than when the macro was recorded (there is no text to select), WordPerfect reports an error (see Figure 2.2).

The error specifies the offending command in the macro file. In this case, the command name is `SelectCharNext`, which tells WordPerfect to select the next character. As a precautionary measure, when such an error occurs, macro playback abruptly stops.

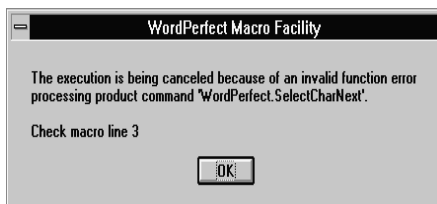


Fig. 2.2. An error message appears when you try to run the *Chartog* macro when the insertion point has not selected text.

There are other contexts you must be aware of. For example, suppose you record a macro that creates a header, types some text, then closes the header. The macro works as long as you always play it at the main document window. However, if you play the macro while already inside a header, WordPerfect will report an error. You cannot place a header inside another header.

Remembering the context of WordPerfect, so that macros play back in the way you expect, is your responsibility. Whenever possible, record macros using a common context. Whenever possible, start and stop macro recording only from the main document window. Avoid macros that switch between documents. Use styles or templates, instead of macros, to select formatting options that might change (like fonts and printers).

Know Where Your Macros Are Stored

As it comes “out of the box,” WordPerfect is set so that macros you record are stored on your computer’s hard disk drive. The exact location depends on the File Preferences settings. Into this folder was placed a number of macros that come with WordPerfect (these are called “shipping macros”).

If you're not sure of exactly which folder is used to hold your macro files, choose Tools, Settings, then choose Files. The Files Settings dialog box appears. Choose Merge/Macro. The folder used to store your macro files is shown in the two boxes.

- The *Default macro folder* is the primary directory for your macros.
- The *Supplemental macro folder* is an optional directory that contains additional macros, for example, if you are on a network.

WordPerfect also allows you to store macros in project templates instead of separate disk files. Macros stored in a template are available whenever you use that template — and conversely, are not available when you aren't using the template. The subject of project templates is quite involved, as is beyond the scope of this book. If you are unfamiliar with templates, be sure to read about them in WordPerfect's on-line help.

Chapter 6

Creating Macros That Pause for User Input

In this chapter:

- How to add a pause during macro recording
- How to add a pause to an already-recorded macro
- Other ways to use the PauseKey command
- Using the PauseSet command
- Adding a message prompt to paused macros

A common task for a macro is starting a document, and stopping mid-way through to wait for user input. You type some text, then tell the macro to continue. A fax cover sheet or a memo is a good example of a process where you might want the macro to complete most of the document for you, but still allow you to enter the *From*, *To*, and *Subject* lines.

In this chapter you will learn how to record a macro that pauses for user-input. You will see that you can have your macro pause just once, or any number of times. You'll also learn how to add a pause command to an already recorded macro that lacks the pausing instruction.

Creating a Fax Cover Sheet with Pauses

A fax cover sheet makes an excellent example of recording a macro that pauses for user-input. Though WordPerfect comes with several fax templates, recording your own lets you easily create a cover sheet just for you. Modifying one of the fax templates that come with WordPerfect can be time-consuming whereas recording a macro just the way you want it takes very little time.

Recording the Fax Macro

Begin by opening a new, blank document. Then follow these steps:

1. Turn on macro recording by pressing Ctrl+F10.
2. In the Macro Record dialog box, type *FAXCOV* in the Name box, and choose Record.
3. Choose Format, Font, and select 18 in the Font Size list, then choose OK.

4. Press Ctrl+B to turn on bold.
5. Choose Format, Line, Center.
6. Type *FAX COVER SHEET*.
7. Press Ctrl+B to turn off bold.
8. Press Enter two times.
9. Choose Format, Font, and select 12 in the Font Size list, then choose OK.
10. Type *Date:*, and a space.
11. Choose Insert, Date/Time. This inserts the current date.
12. Press Enter two times.
13. Type *From:*, and a space.
14. Choose Tools, Macro, Pause. This places a checkmark beside the Pause command (see Figure 6.1).
15. Choose Tools, Macro, Pause again. This removes the checkmark beside the Pause command, and inserts a pause command in the macro.
16. Press Enter two times.
17. Type *To:*, and a space.
18. Repeat steps 14-16.
19. Type *Subject:*, and a space.
20. Repeat steps 14-16.
21. Type *Number of pages:*, and a space.
22. Repeat steps 14-16.
23. Press Enter twice.
24. Turn off macro recording by pressing Ctrl+F10.

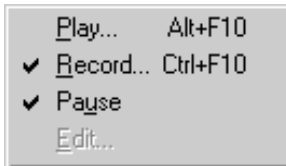


Fig 6.1. A check appears next to the Pause command when macro recording is paused

Playing the Fax Macro

Test the macro by closing the current document without saving it. Open a new blank document. Play the macro by pressing Alt+F10, typing *FAXCOV*, and choosing OK. The macro types **FAX COVER SHEET** in big letters in the center of the page, then inserts the date. It also inserts *To:*, and stops for your input.

1. Type the name the fax is to, and press Enter. Pressing Enter “releases” the pause, and the macro continues.
2. The macro inserts *From:*, and stops for your input. Type who the fax is from, and press Enter.
3. The macro inserts *Subject:*, and stops for your input. Type what the fax is about, and press Enter.
4. The macro inserts *Number of pages:*, and stops for your input. Type the number of pages in the fax, and press Enter.

The macro finishes by inserting two extra lines. The macro has now finished, and you can add a note to the fax cover sheet, or print it out.

Viewing the Fax Macro

You can view the contents of the FAXCOV.WCM macro to see how WordPerfect recorded the commands. Open the macro file by choosing File, Open, locating FAXCOV.WCM (it’s in your default macros directory), and choosing OK.

Here’s how your FAXCOV.WCM should look. Note that the pauses are recorded as *PauseKey (Key: Enter!)* commands. Every time you choose the Pause command from the macro pop-up menu, WordPerfect inserted the PauseKey command. When playing a macro, the PauseKey command pauses the macro and allows for user-input. The Enter! parameter indicates that the pause is released by pressing the Enter key. As you will learn later in this chapter, you can specify a different key or action to unpauses the macro by changing the Enter! parameter.



FAXCOV.WCM

```
Application (WordPerfect; "WordPerfect"; Default!; "EN")
FontSize (FontSize: 18.0p)
```

```
AttributeAppearanceToggle (Attrib: Bold!)
Center ()
Type (Text: "FAX COVER SHEET")
AttributeAppearanceToggle (Attrib: Bold!)
HardReturn ()
HardReturn ()
FontSize (FontSize: 12.0p)
Type (Text: "Date: ")
DateText ()
HardReturn ()
HardReturn ()
Type (Text: "From: ")
PauseKey (Key: Enter!)
HardReturn ()
HardReturn ()
Type (Text: "To: ")
PauseKey (Key: Enter!)
HardReturn ()
HardReturn ()
Type (Text: "Subject: ")
PauseKey (Key: Enter!)
HardReturn ()
HardReturn ()
```

If you've been following along in this book from the first chapters, you've already been introduced to the majority of the other commands in this macro. Commands you have encountered before include `Type`, `HardReturn`, `AttributeAppearanceToggle`, and `Application`. Three new commands are `FontSize`, `Center`, and `DateText`. As usual, the function of these commands is largely self-explanatory.

FontSize sets the size of the font. Note that the font size is provided as the parameter to the `FontSize` command. The font size is indicated in points, with the `p` character. The macro first sets the font size to 18.0p (18 points), and later sets it to 12.0p (12 points).

The *Center* command centers the current line. The `Center` command uses no parameters.

The *DateText* inserts the current date as text. The `DateText` command uses no parameters.

Note

If you see a command in a recorded macro that you are unfamiliar with, and you wish to learn more about it, you can look it up in the Macro Commands Inserter. To do so, click the `Commands` button, and choose `WordPerfect` as the command type. Locate the command in the list, select it, then right-click on it. The online help for that command appears.

Creating a Letter Opener Macro and Adding Pauses

Armed with the `PauseKey` command, you can add pauses to any macro, including macros you've already recorded. Let's see exactly how this process works. In the following, you'll create a macro for "opening" letters.

The macro is quite simple: it inserts the date, then pauses to let you type the recipient's name and address. It then types `Dear` and waits for you to fill in the name. The macro will be recorded without the pauses. The `PauseKey` command will be inserted at critical points to allow the macro to pause and wait for user-input.

Recording the Letter Opener Macro

To record the basic letter opener macro, start from a new, blank document. Then,

1. Press `Ctrl+F10` to start macro recording.
2. At the Record Macro dialog box, type `LETOPEN` in the Name box, and choose Record.
3. Center the text on the page by choosing Format, Page, Center. Choose Current Page, and choose OK.
4. Choose Insert, Date/Time, Insert
5. Press the Enter key *four* times.
6. Type `Dear`, and a space.
7. Press the Enter key twice.
8. Stop macro recording by pressing `Ctrl+F10`.

Clear the document without saving it. Open the `LETOPEN.WCM` macro. The macro should contain the following commands (your macro won't include the numbers; *they are shown here for references purposes only!*).



LETOPEN.WCM

```
1. Application (WordPerfect; "WordPerfect"; Default!; "EN")
2. CenterCurrentPage (State: On!)
3. DateText ()
4. HardReturn ()
5. HardReturn ()
6. HardReturn ()
7. HardReturn ()
8. Type (Text: "Dear ")
9. HardReturn ()
10. HardReturn ()
```

The LETOPEN.WCM macro contains a command that's new to the discussion: *CenterCurrentPage*. The *CenterCurrentPage* command centers the current page. The command uses a parameter, in this case *On!*. As you might guess, you can turn off the page centering by specifying *CenterCurrentPage (State: Off!)*.

Adding Pauses to LETOPEN.WCM

We'll add a number of pauses to the LETOPEN.WCM macro: a series of three pauses to allow you to type the letter recipient's name and address, and another pause to allow you to type the salutation.

Note

Of course, you could have recorded these pauses with the rest of the LETOPEN.WCM macro, but the main purpose of this example is to show you how to manually add pauses to already-recorded macros.

To insert the pause for the recipients name and address, add a blank line after line 5. Then type:

```
PauseKey (Key Enter!)  
HardReturn()  
PauseKey (Key: Enter!)  
HardReturn()  
PauseKey (Key: Enter!)
```

To insert the pause for the salutation, add a blank line after Type ("Dear "), and type the following lines:

```
PauseKey (Key: Enter!)
```

Let's make the letter opener macro even easier to use by providing the colon character after the salutation. Press Enter to add a new line, and type:

```
Type (":")
```

Your macro should now look like the following (again, without the line numbering, which is used here only for reference purposes). Choose File, Save As, and name this version of the macro LETOPEN0.WCM, to differentiate it from the original LETOPEN.

**LETOPEN0.WCM**

```
1. Application (WordPerfect; "WordPerfect"; Default!; "EN")
2. CenterCurrentPage (State: On!)
3. DateText ()
6. PauseKey (Key: Enter!)
8. HardReturn ()
9. PauseKey (Key: Enter!)
10. HardReturn ()
11. PauseKey (Key: Enter!)
12. HardReturn ()
13. HardReturn ()
14. Type (Text: "Dear ")
15. PauseKey (Enter!)
16. Type (Text: ":")
17. HardReturn ()
18. HardReturn ()
```

If, when you save the macro, WordPerfect warns you of a “syntax error,” you made a mistake typing one or more of the commands. Syntax errors are shown in a large message box. The syntax error will point you to the general spot in the macro where the error is. However, the exact error may be a line or two before or after the indicated problem. *Carefully review* your macro for mistakes, and try saving again.

Close the LETOPEN0.WCM macro after it has been successfully saved.

Playing the LETOPEN0.WCM Macro

Play the LETOPEN0.WCM macro to see how your modifications work. Start in a new, blank document. Then,

1. Press Alt+F10 to display the Play Macro dialog box.
2. Type LETOPEN0 in the Name box, and press Play.
3. The LETOPEN0.WCM macro starts. It begins by centering the page, typing the current date, and inserting two hard returns.
4. Type the recipient’s name and address. You need to type a total of three lines, as in:

```
Santa Claus
One North Pole Way
North Pole, Earth 01234
```

Press Enter after each line. (The macro pauses three times to allow you to type all three lines.

5. At the third line, the macro inserts two hard returns, and inserts Dear. Type the salutation, and press Enter.

The macro finishes by inserting a colon, and two hard returns. You can now finish composing the rest of the letter.

Using a Different Key/Action to Unpause the Macro __

The LETOPEN0.WCM macro, described in the previous section, uses a series of three pauses to allow you to type the recipient's name and address. You press the Enter key each time to un-pause the macro. This technique has a disadvantage: what if the address you want to type consists of only two lines, or contains four or even five lines? One method might be to create different macros, each one with a varying number of pauses for typing the recipient's name. But a better method is to simply use a different key or action to unpause the macro.

Instead of using the Enter key to unpause the macro, you can specify that any of the following releases the pause:

- Press the Cancel (Esc) key.
- Choose Close from the File menu.
- Type a specific character, such as A or z.
- Press any printable key.

The last two options, typing a specific character or pressing any printable key (a printable key is a key that prints a character), are unsuited for the task at hand. However, either of the first two methods — pressing the Cancel key or choosing File, Close — can be adopted for use in the LETOPEN.WCM macro. Let's use the Cancel key method as an example.

Open the LETOPEN.WCM macro. Make a copy of it by choosing File, Save As. Type LETOPEN1.WCM, and choose Save. The original LETOPEN.WCM will be left untouched, and we'll edit the new macro, LETOPEN1.WCM.

Delete lines 6 through 11 of the macro. In the place of these lines type the following:

```
PauseKey (Key: Cancel!)
```

The LETOPEN1.WCM macro should now look like this (of course, without the reference line numbers).



LETOPEN1.WCM

1. Application (WordPerfect; "WordPerfect"; Default!; "EN")
2. CenterCurrentPage (State: On!)
3. DateText ()

```
4. HardReturn ()
5. HardReturn ()
6. PauseKey (Key: Cancel!)
7. HardReturn ()
8. HardReturn ()
9. Type (Text: "Dear ")
10. PauseKey (Key: Enter!)
11. Type (Text: ":")
12. HardReturn ()
13. HardReturn ()
```

Save and compile the LETOPEN1.WCM macro, and close it. In a new, blank document, play the LETOPEN1.WCM macro by pressing Alt+F10, typing LETOPEN1, and choosing OK.

This time, enter as few or as many lines for the recipient's address as you want. When you are done, press the Cancel (Esc) key. Notice how this unpauses the macro. The macro inserts the Dear text, and waits for you to type the salutation. As before, press Enter when you are done typing the salutation.

Even More Key Choices

Personally, I dislike using the Cancel key as a macro un-pauser, as it is not very intuitive. Using `PauseKey (Key: Close!)` is an alternative, but it not as easy as simply pressing a single key. You must either choose File, Close, or press Ctrl+F4, to unpause the macro.

Instead of the `PauseKey` method, which allows for only a limited number of keys or actions you can use to unpause a macro, you can use the `PauseSet` command to specify any of dozens of possible key choices). `PauseSet` sets the key (or action) that unpauses the macro. Along with `PauseSet` you use the `Pause` command, which does the actual pausing.

My favorite keys for unpausing macros are Ctrl+Enter and Alt+Enter. These key combinations equate to a hard page break and an end-of-field merge command, respectively. You use the macro command for either of these two actions with `PauseSet`.

- To unpause a macro with Ctrl+Enter, use `PauseSet (Token: HardPageBreak)`.
- To unpause a macro with Alt+Enter, use `PauseSet (Token: MergeEndField)`.

To test the Ctrl+Enter method, open the LETOPEN1.WCM macro file. Choose File, Save As, and resave the file as LETOPEN2.WCM. Replace line 6 of the macro with the following:

```
PauseSet (Token: HardPageBreak)
Pause
```

The entire macro should look like:



LETOPEN2.WCM

```
1. Application (WordPerfect; "WordPerfect"; Default!; "EN")
2. CenterCurrentPage (On!)
3. DateText ()
4. HardReturn ()
5. HardReturn ()
6. PauseSet (Token: HardPageBreak)
7. Pause
8. HardReturn ()
9. HardReturn ()
10. Type (Text: "Dear ")
11. PauseKey (Key: Enter!)
12. Type (Text: ":")
13. HardReturn ()
14. HardReturn ()
```

Save the changes, and close the LETOPEN2.WCM macro file. In a new, blank document, play the macro by pressing Alt+F10, typing LETOPEN2, and choosing Play. At the first pause, type the recipient's name and address. When you are done, press Ctrl+Enter. This unpauses the macro, and it continues as usual. Type the salutation, press Enter, and the macro finishes.

Note

You can also use the `PauseCommand` as follows:

```
PauseCommand (Token: HardPageBreak)
```

It does the same thing as the `PauseSet/Pause` combination detailed above.

Telling the User What to Do When a Macro Is Paused_

When a macro is paused, WordPerfect displays a very dim *Macro Pause* indicator in the Application Bar at the bottom of the document window. It's easy to miss this message, and assume the macro has finished. This can cause problems for users who may not be aware that the macro has been paused.

Even more trouble is brewing if you are use a key other than Enter to unpause the macro. The user may know the macro is paused, but may not know how to unpause it!

It's always a good idea to display a message, and perhaps even sound a beep, when pausing a macro. You can display your own, easier-to-spot message in the Status Bar. This is done with the *MacroStatusPrompt* command. The `MacroStatusPrompt` command uses two parameters: the first parameter tells WordPerfect to display or hide the Status Bar message. The second parameter specifies the message you want to display. Here's an example:


```
MacroStatusPrompt (State: On!; Prompt: "This message appears on the status bar.")
```

To see how the MacroStatusPrompt command works, open the LETOPEN2.WCM macro file. Choose File, Save As, and resave the file as LETOPEN3.WCM. After line 5 of the macro add the following:

```
MacroStatusPrompt (State: On!;  
    Prompt: "Type recipient's name and address; press Ctrl+Enter when done")  
Beep
```

And, following the Type ("Dear ") command, add:

```
MacroStatusPrompt (State: On!; Prompt: "Type salutation; press Enter when done")  
Beep
```

When finished, the macro should look like this:



LETOPEN3.WCM

```
1. Application (WordPerfect; "WordPerfect"; Default!; "EN")  
2. CenterCurrentPage (State: On!)  
3. DateText ()  
4. HardReturn ()  
5. HardReturn ()  
6. MacroStatusPrompt (On!; Prompt:  
7.     "Type recipient's name/address; then press Ctrl+Enter")  
8. Beep  
9. PauseSet (Token: HardPageBreak)  
10. Pause  
11. HardReturn ()  
12. HardReturn ()  
13. Type (Text: "Dear ")  
14. MacroStatusPrompt (State: On!; Prompt: "Type salutation; press Enter when  
done")  
15. Beep  
16. PauseKey (Enter!)  
17. Type (Text: ":")  
18. HardReturn ()  
19. HardReturn ()
```

Save the changes, and close the LETOPEN3.WCM macro file. In a new, blank document, play the macro by pressing Alt+F10, typing LETOPEN3, and choosing OK. The macro pauses to allow you to enter the recipient's name and address. It also beeps at you to get your attention, and display a message in the Status Bar. At the first pause, type the recipient's name and address. When you are done, press Ctrl+Enter. At the next pause, type the salutation, press Enter, and the macro finishes.