

Marvelous Margins with a Macro
WordPerfect Magazine
By Elden Nelson

Note: The macro discussed below has been converted from WP 5.1 for use in WP 6.x.

The included MARGINS.WPM macro gives you more control over setting margins.

Using the Macro

To use the macro, move your cursor to where you want the new margins to take effect, such as the beginning of a page or paragraph. Press Macro (Alt-F10), type "margins" and press (Enter). A menu displays (see below). Press an option to indicate which margins you want to change.

Margins: 1 Lft; 2 Rt; 3 Lft & Rt; 4 Top; 5 Btm; 6 Top & Btm; 7 All Four:

You can change just the left, right, top or bottom margin individually, change the left and right margins to the same setting, change the top and bottom margins so they're the same or change all the margins so they're all the same. Once you've made your choice, the prompt "Enter measurement for margin(s):" displays. Type the value you want, such as 1.5", and press (Enter). The macro sets your new margins for you. If you chose 1 Lft, the macro sets the left margin to your new measurement. If you chose 3 Lft & Rt, the macro sets both the left and right margin to the measurement you typed. If you chose 7 All Four, the macro sets all four of your margins to your measurement. And so on.

How the Macro Works

The first thing the macro needs to do is display the menu and let you pick one of the options. Lines 3-10 use a {CHAR} command to take care of this. The character that you press is then assigned to variable Set.

Next, the macro checks to see whether the character you pressed is a valid option for the menu. To do this, the macro uses a close cousin of the {CASE CALL} command: {CASE}. {CASE} works just the same as {CASE CALL}, with one exception. With the {CASE} command, the macro finds which label name to go to and continues from that label. It never returns to the line following the {CASE} statement.

So, on lines 11-12 the {CASE} command checks all of the correct options against variable Set. If you pressed any of them, the macro jumps to line 14 and continues on. If you pressed an incorrect letter or number, the {ELSE} on line 13 sends the macro back to the {LABEL}Menu~ command on line 2, which brings up the menu again, giving you another chance to press a valid option.

Once you've pressed a workable number or letter, the {TEXT} command on line 15 prompts you to enter the measurement for your margin or margins. What you enter is stored in variable Margin. Line 16 then takes the macro into the Format menu and starts the {CASE CALL} command on line 17. The first line of this long statement (lines 17-24), tells the macro to check variable Set against the options in the upcoming list. If you pressed a "1," an uppercase "L" or lowercase "l," line 18 tells the macro to go to {LABEL}SetLeft~ on line 27.

On lines 28-29, the macro sets the left margin to be the value you typed. The {RETURN} command on line 30 then returns the macro to the command below {CASE CALL}, which is {Exit} on line 25. This returns you to the document screen.

Lines 19-24 work pretty much the same as line 18. {CASE CALL} compares variable Set against the options listed. If it finds a match, the macro goes to the specified label, makes the margin change and returns to line 25, which takes you back to the document screen. Line 26 ends the macro.

A couple more things about this macro are worth noting. First, the {QUIT} command on line 26 is very important. If it weren't there, the macro would continue to {LABEL}SetLeft (line 27), even though the macro has already set a margin. When you use the {CASE CALL} command, chances are the last thing the macro does won't be on the last line. Make sure you have a {QUIT} command where you want the macro to quit.

Next, note the {CALL} commands in lines 52-53. {CALL} is another of {CASE CALL}'s cousins. It tells the macro to go to the label it specifies and perform the commands until it comes to a {RETURN} command. The macro comes back to the command directly after {CALL}. So, {CALL} just has the macro take a detour. It's not a decision-making command, like {CASE CALL}. This command comes in handy when there are a series of commands that need to be performed a number of times throughout the macro. The {CALL} command allows you to frequently use a series of commands, but you only have to enter them once in the macro.