

Connect Two Tables with a Macro
WordPerfect Magazine
By Julie Westergren

My boss wanted revisions on last year's sales lists to take to an important meeting. The sales were charted on twelve separate tables, each in separate files (see Figure 1 below). He wanted them combined into one table for quick reference, and he wanted it quick.

I knew there must be a way, so I racked my brain and finally came up with a solution. I could add multiple rows to the bottom of the January table. Taking one month at a time, I could then block the information and copy it into the empty rows of the January table.

Since this solution involved a lot of keystrokes, I set out to create the included JOIN.WPM macro to speed things up.

Using the macro

You're ready to test the macro. To do so, Retrieve (Shift-F10) your first table to the screen. Press (Home), (Home), (Down Arrow) to move the cursor to the bottom of the document, then Retrieve (Shift-F10) the second table so that both tables are now on the same document screen.

Note: The two tables should have the same number of columns. If not, you'll get unpredictable results. Also, if the first row of the second table is a header, it will no longer be a header once it's in the first table. If this row contains the same information as the first table (as in Figure 1 below), it probably isn't needed. Delete the row before running the macro by placing the cursor in the row and pressing (Ctrl-Del), (Y) Yes.

To run the macro, press Macro (Alt-F10), type "join" and press (Enter). First, the macro prompts you to move the cursor inside the second table and press (Enter). The macro proceeds to copy text from the second table into the first table. The prompt ***Table Joining in Progress. Please Wait*** appears on-screen until the two tables are joined. If any cells have been joined, or attributes added, this information is transferred as well. However, you may get unpredictable results from cells that have been split. Also, only text inside the table will be moved. Any text outside the table will be left at the bottom of the document.

If you're joining more than two tables, press (Home), (Home), (Down Arrow) and Retrieve (Shift-F10) the next table, then run the macro again. Continue this process for all tables to be joined.

Alphabetizing and cleanup

Figure 2 (see below) shows the final result after joining and alphabetizing the two tables in Figure 1. If you'd like your final table alphabetized, place the cursor inside the table and press Merge/Sort (Ctrl-F9), (2) Sort. Set the keys to sort and choose (1) Perform Action.

After alphabetizing the table, you'll probably have to clean up some of the lines, especially at the bottom. Place the cursor inside the table and press Columns/Table (Alt-F7). If your bottom line disappeared after alphabetizing, press (Home), (Down Arrow), then Block (Alt-F4), (End). Next choose (3) Lines, (4) Bottom, then either (2) Single or (3) Double, depending on what your border is. You may have to perform similar keystrokes to clean up other lines in the table.

How the macro works

Line 1 begins with a {LABEL}Start~. This is a marker the macro can come back to if you make a mistake and don't place the cursor inside the second table. Note: If the cursor is in the first table, the contents will be deleted and the macro will quit. Be sure the cursor is in the second table.

The {PROMPT} command (line 2) prompts you to place the cursor anywhere in table 2. The {PAUSE} command on line 3 waits for you to do so. Pressing (Enter) ends the pause.

Lines 4-8 of the macro check to see if the cursor is inside a table. If so, the macro proceeds to lines 9-10, prompting you that the table joining is in progress. If the cursor isn't inside a table, line 7 returns the macro to line 1, where you'll again be prompted to place the cursor in table 2. If you don't have a table on-screen, press Cancel (F1) to cancel the macro.

Line 11 turns display off so you can't see the execution of the joining process. Line 12 of the macro edits the second table and moves to the last cell in the bottom row. Line 13 uses the {SYSTEM}row~ command to determine the number of rows in the table. The {ASSIGN} command assigns this number to variable 2.

Line 14 then blocks all the contents of table 2 and moves it out of the table and into the computer's memory. The {Exit} command exits the table.

Lines 15 and 16 search upward in the document for the first table. The table is then edited and the cursor moves to the last cell in the bottom row of the table.

The macro again uses the {SYSTEM} row~ command on line 17 to determine the last row number in table 1 and uses the {ASSIGN} command to assign it to variable 1. The number of rows in table 2 (variable 2) is added to the number of rows in table 1 (variable 1) in line 18 and the sum is then assigned to variable 3.

Line 19 then increases the size of the first table by using the value of variable 3 to determine the size. This way the table contains its original number of rows, plus the number of rows from table 2.

After the new rows are added, line 20 moves the cursor up to the first blank row where line 21 copies the contents of table 2 and exits the table.

The last four lines (22-25) of the macro search for the Table Definition code of table 2, which is now empty. The table is blocked and deleted from the document, leaving only one table with the combined contents of both. The cursor is then returned to the top of table 1 and the macro quits.

Figure 1. X

Monthly Sales-January				it
Sales Rep.	Date of Report	Territory	Amount	
Marilyn Baker	1/7/92	San Francisco	\$17,900.00	
Suzanne Benson	1/23/92	North L.A.	\$15,900.00	
Paul Kelly	1/28/92	Sacramento	\$23,900.00	
Kep Kennedy	1/11/92	South L.A.	\$32,400.00	
Joe Smith	1/15/92	San Diego	\$45,900.00	
Monthly Sales-February				
Sales Rep.	Date of Report	Territory	Amount	
Marilyn Baker	2/8/92	San Francisco	\$31,500.00	
Suzanne Benson	2/22/92	North L.A.	\$27,300.00	
Paul Kelly	2/26/92	Sacramento	\$26,600.00	
Kep Kennedy	2/9/92	South L.A.	\$37,200.00	
Joe Smith	2/16/92	San Diego	\$15,500.00	

Figure 1. Two of the twelve tables to be joined into one table.