

Finding Duplicates in Secondary Files
WordPerfect Magazine
By Matt Beck

Few things are more frustrating than merging your primary and secondary files and finding you have three letters addressed to the same John Doe and four letters going to the same Sally Smith.

How can you determine if your secondary file contains duplicate records? This was a problem for me until I created the included FINDDUPE.WPM macro.

This macro first prompts you to enter the number of the field that contains the last name; it then prompts you to enter the number of the field that contains the address.

Once this is done, the macro sorts your secondary file by the indicated fields and compares each record to find possible matches. If a match is found, a comment indicating that the records could be duplicates is inserted.

You'll need to determine if the records are really identical, or if they just have a common name. For example, there could be five records with "John Anderson," but only two may be actual duplicates.

If the macro doesn't find a match, the second record is compared with the third and so on until all records have been compared. When the macro is finished, a prompt appears indicating how many duplicate records have been found. All you then need to do is search for the comments.

Using the macro

With your secondary file on-screen, press Macro (Alt-F10), type "finddupe" and press (Enter). First you'll be prompted to enter the field that contains the last name, and then to enter the field that contains the address. You should always enter the number of the field. If you use {FIELD NAMES}, do not enter the name of the field; you'll still need to enter the number.

Your file will be sorted and then the prompt "Please wait, working on record X" (where "X" is the record number) will display. If duplicate records are found, a comment is inserted in the matching record. When the macro is finished, the prompt "Finished! X possible duplicate(s). Press any key to continue, then press Search (F2) twice to find comments." appears, indicating how many matching records have been found. This message remains until you press any key, then the cursor is left at the first comment, or, if no duplicates have been found, the cursor is left at the top of the document. If there are duplicates, simply press Search (F2) twice to find each comment the macro inserted. Since names can be alike without the record being duplicate, you're given a chance to manually delete the records that are duplicate instead of having the macro automatically delete records with identical names and addresses that are not actually duplicates. *Note: This macro is case sensitive and will not find duplicates with an irregular case. For example, Jones and JOnes would not be considered duplicate.*

Explaining the macro

After invoking the macro, line 1 turns the display off to hide the actual keystrokes of the macro. Line 2 assigns the value of one to {VARIABLE}rcrdcnt~. This variable keeps track of the record the macro is on and is used in the prompt on line 14.

Since a macro normally quits if it encounters a "Not Found" error, the {ON NOT FOUND}~ on line 2 tells the macro to ignore any "Not Found" error that occurs during the macro. Line 3 prompts you to enter the field number that contains the last name. Once this is done, you're prompted for the field number that contains the address (line 4).

After you press (Enter), your secondary file is sorted (lines 5-7). This places all the records in alphabetical order according to the last name and, at the same time, places the addresses in numerical order.

When the file is sorted, the value of zero is assigned to {VARIABLE}cnt~ (line 9). This variable keeps track of how many records are duplicates. It is used in the prompt on lines 33 through 35. Next, line 10 uses the {SYSTEM} command to determine if the cursor is on a {FIELD NAMES} code. If it is, the macro searches for a hard page (line 11) which would actually start the list of names. If the cursor is not on the {FIELD NAMES} code, the macro skips lines 11 and 12 and prompts you with: Please wait, working on record n (line 14) while the macro searches for duplicate records. Line 16 "calls" lines 38-44 to determine where to start comparing fields. These lines also check if the macro is at the end of the secondary file (line 39). If it is, it goes to line 32, displays a prompt and ends the macro.

Once the starting point is found, the macro assigns the last name in the first record to {VARIABLE}record1~ (line 18).

Line 19 sends the cursor back to the top of the record by searching backwards for a hard page code. Then line 20 calls the routine on lines 45-48, which finds the address field. Line 21 blocks the first word of that field and line 22 assigns it to {VARIABLE}check1~.

The macro then goes to the next record (line 23) and loops back to line 15 to execute the commands in the {FOR} loop (lines 15-24).

The second time these commands are used, {VARIABLE}m~ is incremented. This will now assign the last name and address to {VARIABLE}record2~ and {VARIABLE}check2~ respectively. Once this is done, lines 25 and 26 check if {VARIABLE}check1~ and {VARIABLE}check2~ match. If they do, {VARIABLE}cnt~ is incremented by one and a comment is inserted (lines 27-28). If they don't match, the cursor just goes to the next record in the secondary file (line 30). Line 31 increments {VARIABLE}rcrdcnt~ by one and goes back to line 13, where the macro starts to compare the record number two with record number three.