

INTEROFFICE MEMORANDUM

Center	<u>DATE:</u>	September 14, 1989	<u>FROM:</u> Information
	<u>PRODUCT:</u>	WordPerfect	<u>VERSION:</u> 5.1
	<u>RELEASE DATE:</u>	All	
	<u>SUBJECT:</u>	Merging One Record Over and Over Again	

At times, users would like to print one name and address over and over again on labels. This might be for return address labels, for example. This can be done by changing the number of copies to print. This would work especially well for users printing one-up labels on a dot matrix or daisy wheel printer. Users of laser or other page printers may find that merging the same record over and over works better. There are two ways to do this.

1. Add a counter to the primary file by using the {FOR} command so that it will repeat the address a specific number of times. For an example of this, see the {FOR} command in the Advanced Merge section of the 5.1 manual.
2. A secondary file should be created with one record (for example, the company name and address). The primary file should contain the appropriate field codes and followed by {NEST PRIMARY}~. Press Enter when prompted for a file name. This will cause WordPerfect to merge the same record over and over until the user presses the F1 key. When the desired number of pages (labels) are merged, then the user can print without having to change the number of copies. For example, if he wants 11,000 copies, he can do the merge, wait a couple minutes, then press F1 and check to see how many pages there are, and then print. If this is to be printed on a page printer, the necessary formatting codes (margins, columns, etc.) should be placed in the primary file and the merged document.

Memo ID: 002_WP51

INTEROFFICE MEMORANDUM

<u>DATE:</u>	April 4, 1991	<u>FROM:</u> Info. Center
<u>PRODUCT:</u>	WordPerfect	<u>VERSION:</u> 5.1
<u>RELEASE DATE:</u>	ALL	
<u>SUBJECT:</u>	Merge Codes - Printing A Merge Files	

If the option for Merge Codes is set to display NO in the setup, the codes will not print. This is helpful for those wishing to print a merge file without having to run another merge to eliminate the actual merge characters. If the user is printing an address list, the document is still formatted with the hard returns and prints only one address per page.

Memo ID: WP51_019

INTEROFFICE MEMORANDUM

DATE: November 16, 1989 FROM: Info. Center
PRODUCT: WordPerfect VERSION: 5.1
RELEASE DATE: 11/06/89
SUBJECT: Error: "Unable To Process Secondary File

Field Names"

If the user received the error message, "Unable To Process Secondary File Field Names" while merging, check to see if there is a missing tilde at the end of the {FIELD NAMES} definition. If there is a missing tilde, enter it, resave the secondary file, and rerun the merge.

Memo ID: WP51_145

INTEROFFICE MEMORANDUM

Center	<u>DATE:</u>	February 5, 1990	<u>FROM:</u> Information
	<u>PRODUCT:</u>	WordPerfect	<u>VERSION:</u> 5.1
	<u>RELEASE DATE:</u>	All	
Secondary Merge File	<u>SUBJECT:</u>	Merge - Editing Field Names In A	

When using field names in a secondary merge file, the user may edit the field names if desired. However, the display on the lower left side of the screen will continue to display the old field names until the user presses the Home key three times followed by the Up Arrow key.

Memo ID: WP51_674

INTEROFFICE MEMORANDUM

<u>DATE:</u>	October 22, 1990	<u>FROM:</u> Info. Center
<u>PRODUCT:</u>	WordPerfect	<u>VERSION:</u> 5.1
<u>RELEASE DATE:</u>	All	
<u>SUBJECT:</u>	Merge - Using An INPUT Command Within	

Underline Codes

A customer created a merge file with an underscore line created with underline. On the line the user included an {INPUT} command. Each time they merged, the line moved to the right when entering text.

To avoid the movement, the user pressed Shift-F8, 4, 7 and answered yes to underline spaces and tabs. After making the change, the user pressed F8 to turn underline on and then entered the {INPUT} command and message. Next the user pressed Alt-F6 (Flush Right) and turned off underline. The line then remained in the desired position when running the merge.

Memo ID: WP51_3121M

INTEROFFICE MEMORANDUM

<u>DATE:</u>	February 27, 1991	<u>FROM:</u>	Info. Center
<u>PRODUCT:</u>	WordPerfect	<u>VERSION:</u>	5.1
<u>RELEASE DATE:</u>	ALL		
<u>SUBJECT:</u>	Merging With A File Created In A Database		

Program

Customers can merge a WordPerfect primary file with a secondary file created in a database program other than DataPerfect. To do so the user must first export the data as a delimited file from the database program. After exporting the file the user need only to preform the merge in WordPerfect. After entering the name of the database file, the screen listing the current merge delimiters will appear. If the settings are not correct, the user may change them at that time. The merge will continue as expected after exiting the merge delimiter screen.

Note: Exporting a file into an delimited format means that a character or ASCII code has been assigned for the end field and end record. The default delimiters in WP are a comma (,) for the end field and a carriage return ([CR]) for the end record. The defaults values for the merge codes can be changed in the Setup menu under Initial Settings and Merge (Shift-F1, 4, 1). This information would apply regardless of the database program being used (e.g., dBASE, Oracle, Foxpro, Advanced Revelation, Paradox, Clearion, etc).

Memo ID: WP51_4420M

INTEROFFICE MEMORANDUM

<u>DATE:</u>	April 1, 1991	<u>FROM:</u>	Info. Center
<u>PRODUCT:</u>	WordPerfect	<u>VERSION:</u>	5.1
<u>RELEASE DATE:</u>	ALL		
<u>SUBJECT:</u>	Merge - Creating A Secondary File From A		

Secondary File

One customer wanted to create a smaller secondary file using a secondary file that contained 60 fields per record. This is an example of how this was done:

1. Create a primary file containing the field codes needed for the secondary file. For example: {FIELD}3~

{FIELD}50~
{FIELD}9~

2. Use the {MRG CMD} (Shift-F9, 6) code to insert the new {END FIELD} and {END RECORD} codes. For example:

{FIELD}3~{MRG CMD}{END FIELD}{MRG CMD}
{FIELD}50~{MRG CMD}{END FIELD}{MRG CMD}
{FIELD}9~{MRG CMD}{END FIELD}{MRG CMD}
{MRG CMD}{END RECORD}{MRG CMD}

3. Perform the merge using the new primary file.

Source: Customer testing

Memo ID: WP51_4816K

INTEROFFICE MEMORANDUM

DATE: April 2, 1991 FROM: Info. Center
PRODUCT: WordPerfect VERSION: 5.1
RELEASE DATE: All
SUBJECT: Mail Merging ASCII

Delimited Files

Where do these files come from?

ASCII delimited files mostly come from Database packages such as Dbase III or IV, however, they can also come from spreadsheets programs.

To save a Dbase III or IV file in ASCII delimited format do the following command at the Dot prompt in Dbase.

USE (name of the database file)
COPY TO (file to be saved in ASCII Delimited format) DELE.

What ASCII delimited files look like?

ASCII is simply DOS Text or a general text format recognized by IBM compatible computers. Delimited means separated. If you were to look at an ASCII delimited file it would look like normal text with a comma or other characters to separate the fields, and probably a new line to separate a record.

The following is an example of an ASCII file without delimiters.

```
John Doe 100 N. 200 W. Provo UT 84057  
Tom Smith 100 N. 200 W. Provo UT 84057
```

Below is an ASCII file with delimiters, and it is called ASCII Delimited.

```
"John","Doe","100 N. 200 W.,""Provo","UT","84057"  
"Tom","Smith","100 N. 200 W.,""Provo","UT","84057"  
└── The comma is the field delimiter
```

A new line (Carriage return and a line feed) indicate a new record.

The quotation marks simply indicate that the information was a character field, and the quotations are stripped out since they are not needed.

The field delimiter does not have to be a comma, it could be almost any character, however, commas are generally used. Similarly a new line (Carriage return and a line feed) indicate a new record.

You can merge directly with an ASCII file without converting it to a secondary merge file if you have WP51.

To do this you must be in WordPerfect, press Ctrl-F9, 1 and type the name of the primary file. When WP asks for secondary file, press F5 and change to the directory which contains the ASCII text file, highlight the ASCII text file, and press 1 for Retrieve. WordPerfect is able to detect that this is an ASCII delimited file and a menu appears which asks for the beginning and ending delimiters for the fields and records. In most cases the beginning delimiter for a field will be a quotation mark ("), and the ending delimiter for a field will be a quotation mark with a comma ", and you should use a single " without the comma if it does not merge properly. There generally will not be a beginning record delimiter, but the ending delimiter will be a [CR] for carriage return which can be obtained by pressing Ctrl-M. When you are sure that you have entered the correct information press Enter and the merge will be performed.

If you desire to change the initial defaults for the delimiter characters, press Sh-F1,4, 1 and enter the delimiter character. This allows you alter initial settings permanently for the beginning and ending delimiter characters used for the fields and records.

Source: Other

Memo ID: WP51_4833B

INTEROFFICE MEMORANDUM

<u>DATE:</u>	April 18, 1991	<u>FROM:</u> Info. Center
<u>PRODUCT:</u>	WordPerfect	<u>VERSION:</u> 5.1
<u>RELEASE DATE:</u> ALL		
<u>SUBJECT:</u>	Advanced Merge - {IF} Statement Is Case Sensitive	

The {IF} statement used in the Advanced Merge is case sensitive. For example, the macro statement {IF}"{VARIABLE}9~"="ALL"~{NEST}infob~{END IF} (as apposed to {IF}"{VARIABLE}9~"="all"~{NEST}infob~{END IF}) would find occurrences of the word **all** when it appeared in caps.

The reason for this is found on page 839 of the WordPerfect 5.1 Manual (the italic footnote). It says, "In a string comparison, the WordPerfect Character set values are compared." Since the characters in the word "ALL" have different character set values than the characters found in the word **all**, the IF comparison can tell the difference between the two words.

Memo ID: WP51_5412M

DATE: April 2, 1990 FROM: Info. Center
PRODUCT: WordPerfect VERSION: 5.1 RELEASE DATE: All
SUBJECT: Merging Into Table -- Table Expanding in Size

To merge into a WordPerfect 5.1 table and have the table automatically increase the number of rows as needed, use the following format:

{LABEL}loop~{FIELD}1~		{FIELD}2~	
{NEXT RECORD}{GO}LOOP~			

As the file merges, the table will expand in size. That is, the number of rows needed for all of the records will be added. There will be a blank line at the bottom of the final merged table. The bottom line of each cell will become double.

Users wishing to keep the bottom line single can add an extra row to the table before running the merge. After the merge is complete, the user can delete the extra rows. Users do not have to enter Table Edit to delete the row. They need only to position the cursor in the extra row and pressing Ctrl-Delete.

{LABEL}loop~{FIELD}1~		{FIELD}2~	
{NEXT RECORD}{GO}LOOP~			

Memo ID: WP51_1326

DATE: January 28, 1991 FROM: Info. Center
PRODUCT: WordPerfect VERSION: 5.1 RELEASE DATE: All
SUBJECT: Merge - Explanation Of {IF}, {IF NOT BLANK} and
 {IF BLANK}

This memo explains the purpose of the {IF}, {IF NOT BLANK} and the {IF BLANK} merge statements. WordPerfect 5.1's merge language makes it possible to make many more selections when merging. The commands have also been adapted to work with Tables.

{IF}

The {IF} statement is very popular. Customers may use the {IF} statement when wanting to merge certain records only if one or two conditions exist. When using the {IF} statement, it should always be paired with an {END IF} command. The {END IF} statement functions somewhat like a period in a regular sentence. It must be included for the merge to work properly.

Users may also use the {ELSE} statement with {IF}. The {ELSE} statement functions like the word or in a conditional sentence. For example, {IF} it rains today we will go to the mall {ELSE} we will go to the park for a picnic {END IF}.

Another example of using the {IF} statement would be when wanting a merge to create a list of all of the females included on a file. The syntax for such a procedure, if a gender field is present, would be:

```
{IF}{FIELD}gender~=Female~  
          {FIELD}name~, {FIELD}address~  
{END IF}
```

This is very close to the final form needed for the merge. There is, however, one problem. Computers work only with numbers and because of this, WordPerfect has no idea what female is. Because of this, the word female must be enclosed in quotation marks. The quotation marks tell WordPerfect to compare what is in the quotation marks instead of trying to equate the information in number form. Because female will be in quotes, the command {FIELD}gender~ should also be enclosed in quotation marks. Without the quotation marks, the computer will try to compare the word female with a number since it thinks the field is a number unless told otherwise. The final form would look like the following:

```
{IF}"{FIELD}gender~"="Female"~  
          {FIELD}name~, {FIELD}address~  
{END IF}
```

The tilde after the word female works like a comma; it ends the first part of the condition. The tilde could be treated like the word then. ({IF} field gender equals female {THEN} print the name and address.)

In this example it is possible for the word company to appear in the gender field. If so, and the user would like a list of all males and females but not companies the expression would something like the following:

```
{IF}"{FIELD}gender~"="Female"
```

```

    "{FIELD}gender~"="Male"~
      {FIELD}name~, {FIELD}address~
  {END IF}

```

(The two vertical bars or pipes represent the or command.) When using the vertical bars, the syntax tells the program to print the name and address if the gender field is equal to female or male. Since the company name would not meet this criteria, WP will skip the company names and only print the male and female information.

The {IF} command can be used to pull certain records from a file containing a list of names and ages. For example, when wanting to obtain a list of all females who are 50 years old, the syntax would read as follows:

```

  {IF}"{FIELD}gender~"="Female"           <----First condition
    {IF}{FIELD}age~>50~                   <----Second condition
      {FIELD}name~, {FIELD}address~
    {END IF}
  {END IF}

```

This procedure is called nesting. If the gender is not female the merge will automatically move on to the next record. If the gender is female, it will then compare to see if the female's age is over 50. Since the age field is a number, the field does not need to be in quotation marks. Also, since two {IF} statements were used two {END IF} statements are needed. If having trouble with the merge working correctly verify that there is a balance between the number of {IF}s and {END IF}s.

Shown below is an example of using the and command in a merge.

```

  {IF}"{FIELD}gender~"="Female"&"{FIELD}gender~"="Male"~
    {FIELD}name~
    {FIELD}address~
  {END IF}

```

This will print a list of all the males and the females.

{IF BLANK}

Customers using versions of WordPerfect before 5.1 were given the question mark (?) to use when wanting to close up a line if a field was blank. The question mark can still be used, but since it also sends a command to delete to the end of the line (thus eliminating the option of having additional text on the line) it makes it impossible to add additional text after the question mark on the same line. Because of this, the {IF BLANK} command was created. The syntax for this command would look something like the following:

```
{FIELD}1~{IF BLANK}2~ {FIELD}3~{ELSE} {FIELD}2~ {FIELD}3~{END IF}
```

This setup will always print fields 1, 2 and 3, but if field 2 is blank it will shorten the space and print just field one and three.

The syntax would look something like the following if the possibility of having two or more blank fields on one line arises:

```
{FIELD}1~ {IF BLANK}2~{FIELD}3~{ELSE}{FIELD}2~ {FIELD}3~{END IF}{IF
BLANK}3~{FIELD}4~{ELSE} {FIELD}4~{END IF}
```

{IF NOT BLANK}

The {IF NOT BLANK} command can be very useful for users who want records if they are not blank. For example, a user can create a field that is only filled if the record is to be taken. If the record is desired a character can be placed within it. If the record is not desired the field is left blank.

The possible syntax would look something like the following:

```
{IF NOT BLANK}include~{FIELD}name~  
{FIELD}address~{END IF}
```

This syntax would only print the person's name and address if the include field was filled. If the field was blank the program would skip over it to the next record.

When using the merge language there is often times more than one way to handle a situation. For example, the above statement for the include field could have been written as follows:

```
{IF}"{FIELD}include~"="x"~{FIELD}name~  
{FIELD}address~{END IF}
```

Certain codes may take less space and work better for the individual user.

The examples given below were designed for customers who print the mailing address of one of their clients, unless it is blank. If it is blank, they want to print another field that contains the person's PO Box address. The syntax could use the {IF NOT BLANK} or the {IF BLANK} commands.

If using the {IF NOT BLANK} statement the syntax would look as follows:

```
{IF NOT BLANK}mail~{FIELD}name~  
{FIELD}mail~{ELSE}{FIELD}name~  
{FIELD}postbox~{END IF}
```

If using the {IF BLANK} statement the syntax would look as follows:

```
{FIELD}name~  
{FIELD}mail~{IF BLANK}mail~{FIELD}postbox~{END IF}
```

Which ever way is used be sure to include the {END IF} and tildes in the appropriate spots.

The example below can be used if wanted to print a variable number of labels of just one address. The paper size and type code would be in the initial settings like normal. This merge is designed to stop, ask the user for the number of labels and then make that many labels.

Insert the following codes in the primary file:

```
{TEXT}1~Enter number of times to repeat label: ~{COMMENT}  
~{FOR}count~1~{VARIABLE}1~~1~{FIELD}1~  
===== (hard page) =====  
{END FOR}
```

Insert the following codes in the secondary file:

Joe Smith

100 North West Rd

Orem, UT 93933{END FIELD}

{END RECORD}

======(hard page)===

Source: In-house testing

Memo ID: WP51_4112M

INTEROFFICE MEMORANDUM

<u>DATE:</u>	May 21, 1991	<u>FROM:</u> Info. Center
<u>PRODUCT:</u>	WordPerfect	<u>VERSION:</u> 5.1
<u>RELEASE DATE:</u>	All	
<u>SUBJECT:</u>	Merge - Prevent Blank Lines	

Using {COMMENT}{Enter}~ is one way of preventing the blank lines in a merge using the {IF NOT BLANK} command. A better way is to arrange your commands like the following:

```
{IF NOT BLANK}title1~{FIELD}title1~{Enter}
{END IF}{IF NOT BLANK}title2~{FIELD}title2~ .....etc.
```

The key is to put the {END IF} after the {Enter}.

Memo ID: WP51_5427B

INTEROFFICE MEMORANDUM

DATE: May 22, 1991 FROM: Info. Center
PRODUCT: WordPerfect VERSION: 5.1
RELEASE DATE: ALL
SUBJECT: Merging With DOS Delimited Text Files

Using Question Marks

When merging with DOS delimited text files, if the customer is using the {IF BLANK} or {IF NOT BLANK} statements, sometimes the merge leaves blank lines. The merge will work properly; however, if question marks are used and there is only one field on each line. If there is more than one field on a line, and a question mark is used in the first field, and the first field is blank, the other information on that line will be deleted.

Memo ID: WP51_5435F

INTEROFFICE MEMORANDUM

<u>DATE:</u>	June 5, 1991	<u>FROM:</u> Info. Center
<u>PRODUCT:</u>	WordPerfect	<u>VERSION:</u> 5.1
<u>RELEASE DATE:</u>	ALL	
<u>SUBJECT:</u>	Merging One Record A Specified Number	

Of Times

The following codes will merge an address the number of times specified by the second number listed in the {FOR} command line. The {COMMENT} command removes the blank line which is caused by the {FOR} command.

Merge File Set Up **Without** Secondary File

{FOR}count~1~30~1~{COMMENT}

~John Smith

1234 Center St.

Anywhere, UT 84604

======(Hard Page Break)=====

Merge File Set Up **With** Secondary File

{FOR}count~1~30~1~{COMMENT}

~{FIELD}1~

{FIELD}2~

{FIELD}3~

{FIELD}4~

======(Hard Page Break)=====

See also WP51_4112M.

Memo ID: WP51_5631F

INTEROFFICE MEMORANDUM

<u>DATE:</u>	June 5, 1991	<u>FROM:</u> Info. Center
<u>PRODUCT:</u>	WordPerfect	<u>VERSION:</u> 5.1
<u>RELEASE DATE:</u>	ALL	
<u>SUBJECT:</u>	Merge - Creating A Block Protected List In	

Columns

To create a primary file that merges three fields into three separate block protected columns (without using the Labels Paper Size/Type definition) a user can do the following:

1. Define three columns as Parallel With Block Protect (press Alt-F7, 1 (Columns), 3 (Define), 1 (Type), 3 (Parallel With Block Protect), 2 (Number Of Columns), type **3** then press Enter and F7 (Exit).
2. Turn the columns on (press 1 (On)).
3. Put a field code in the first column (press Shift-F9 (Merge Codes), 1 (Field), type **1** then press Enter). Press Ctrl-Enter to go to the second column.
4. Put a field code in the second column (press Shift-F9 (Merge Codes), 1 (Field), type **2** then press Enter). Press Ctrl-Enter to go to the third column.
5. Put a field code in the third column (press Shift-F9 (Merge Codes), 1 (Field), type **3** then press Enter).
6. Turn the columns off (press Alt-F7, 1 (Columns), 2 (Off)).
7. Create a blank line between rows (press Enter).
8. Press Shift-F9, 5 (Next Record).
9. Press Shift-F9, 4 (Page Off).

The primary file should look like this:

{Field}1~ {Field}2~ {Field}3~

{Next Record} {Page Off}

Memo ID: WP51_5643F

INTEROFFICE MEMORANDUM

<u>DATE:</u>	June 5, 1991	<u>FROM:</u> Info. Center
<u>PRODUCT:</u>	WordPerfect	<u>VERSION:</u> 5.1
<u>RELEASE DATE:</u>	ALL	
<u>SUBJECT:</u>	Work Around For Decimal Places In Macros	

And Merge

In Memo WP51_2350, it is stated that WP 5.1 does not currently support decimal places in merge and macros. The following merge primary file is a work around for a situation where a secondary file's first field contains a number between 1 and 4, and that number may contain a decimal value such as 2.34 or 3.9. The merge uses the {IF} statement and tests the value of the first whole number. For example, the 2 in 2.34 is evaluated to see if it is between 1 and 4. The {MID} command is used to do the evaluating, as it allows a focus on a number or numbers within a decimal numeral.

The following merge primary file is designed to work with numbers between 1 and 9.9; however, if numbers up to 10 or greater will be evaluated, the variables that are a part of the {MID} command would need to be adjusted to allow for numbers less than 9.999 to be evaluated by extracting 1 whole number character, and numbers greater than 9.999 to be evaluated by extracting 2 whole number characters. Numbers between 100 and 999.99 would need to use the {MID} command to extract 3 whole number characters, and so on. The {MID} command is further explained on page 892 of the WP 5.1 Manual. Please note, the {COMMENT} command is used to avoid blank lines that are created during the merge.

```
{LABEL}start~{COMMENT}
~{IF}{MID}{FIELD}1~~0~1~>1&{MID}{FIELD}1~~0~1~<4~{COMMENT}
~{GO}merge~{COMMENT}
~{ELSE}{COMMENT}
~{NEXT RECORD}{COMMENT}
~{CALL}start~{COMMENT}
~{END IF}{COMMENT}

~{LABEL}merge~{COMMENT}
~{FIELD}1~
```

The conditional merge works as described, but on the last line at the end of the merged file, part of the {IF} statement is printed. To avoid the problem, make sure the last record in the secondary file is blank, i.e., the last record should contain nothing but one {END FIELD} code for each field in the record and the record should end with an [HPg] code.

Memo ID: WP51_5645F

INTEROFFICE MEMORANDUM

<u>DATE:</u>	June 6, 1991	<u>FROM:</u> Info. Center
<u>PRODUCT:</u>	WordPerfect	<u>VERSION:</u> 5.1
<u>RELEASE DATE:</u>	ALL	
<u>SUBJECT:</u>	Merging Into A Tables Primary File Using	

{WHILE}

Customers who would like to merge information into a table can set up a primary file that looks like the following:

{WHILE}"{FIELD}1~!="end"~{FIELD}1~ {FIELD}2~ {FIELD}3~ {FIELD}4~
{NEXT RECORD}{END WHILE}

⌋

The following codes would display in Reveal Codes:

```
[Tbl Def:l;4,1.88",1.88",1.88",1.88"]  
[Row][Cell][Mrg:WHILE]"[Mrg:FIELD]1~!="end"~[Cell][Mrg:FIELD]1~  
[Cell][Mrg:FIELD]2~[Cell][Mrg:FIELD]3~[Cell][Mrg:FIELD]4~  
[Row][Cell][Mrg:NEXT RECORD][Mrg:END WHILE][Cell][Cell][Cell]  
[Tbl Off]
```

See also WP51_1326 for another method of creating a table merge file.

Memo ID: WP51_5662F

INTEROFFICE MEMORANDUM

DATE: May 22, 1991 FROM: Info. Center
PRODUCT: WordPerfect VERSION: 5.1
RELEASE DATE: ALL
SUBJECT: Error Message "Static Buffer Is Full" When

Merging

A customer received the error message "Static Buffer Is Full" after merging a primary file with a secondary file. It was discovered that the primary file contained a field code that was lacking a number, for example, {FIELD}~ instead of {FIELD}1~. When a number was inserted in the field code, the document merged just fine.

Memo ID: ALL_9366F

INTEROFFICE MEMORANDUM

<u>DATE:</u>	06/21/91	<u>FROM:</u> Info. Center
<u>PRODUCT:</u>	Office	<u>VERSION:</u> 2.0
<u>RELEASE DATE:</u>	ALL	
<u>SUBJECT:</u>	Merging Notebook Files With Different	

Field Orders

The following is an example of how to merge two Notebook files that have the same fields, but the field orders are different:

FILE 1		FILE 2	
1	Date	1	City
2	Time	2	Date
3	Home Phone	3	Time
4	Street	4	Home Phone
5	Modem	5	Zip
6	Visitor	6	Place
7	COUNTRY	7	Note
8	City	8	Modem
9	Place	9	Visitor
10	Note	10	COUNTRY
11	Sales	11	Sales

1. Merge File 2 into the following primary file (which uses the File 1 field order):

```
{FIELD}Date~*
{FIELD}Time~*
{FIELD}Home Phone~*
{FIELD}Street~*
{FIELD}Modem~*
{FIELD}Visitor~*
{FIELD}COUNTRY~*
{FIELD}City~*
{FIELD}Place~*
{FIELD}Note~*
{FIELD}Sales~*
@
```

The following is an example of how the merged file should look:

```
February 15, 1990*
2:00 P.M.*
222-6492*
199 Starcrest*
222-6492*
Y*
USA*
Orem*
Here*
Rusty Johnson February 21, 1991
```

Instructions to find this location are not clear.*
\$1,000,000,000*
@

2. Search and replace the * with a and the @ with a . The following is an example of what the file should look like when this step is completed:

February 15, 1990
2:00 P.M.
222-6492
199 Starcrest
222-6492
Y
USA
Orem
Here
Instructions to find this location are not clear.
\$1,000,000,000

3. If the two files don't have duplicate records, the two secondary merge files can be added together.
4. Retrieve the secondary file into Notebook.

Memo ID: LIB_388F

INTEROFFICE MEMORANDUM

DATE: April 2, 1990 FROM: Info. Center
Titara/Brenda Jones
SUBJECT: Merging Into Table -- Table Expanding in Size

To merge into a WordPerfect 5.1 table and have the table automatically increase the number of rows as needed, use the following format:

{LABEL}loop~{FIELD}1~		{FIELD}2~	
{NEXT RECORD}{GO}LOOP~			

As the file merges, the table will expand in size. That is, the number of rows needed for all of the records will be added. There will be a blank line at the bottom of the final merged table. The bottom line of each cell will become double.

Users wishing to keep the bottom line single can add an extra row to the table before running the merge. After the merge is complete, the user can delete the extra rows. Users do not have to enter Table Edit to delete the row. They need only to position the cursor in the extra row and pressing Ctrl-Delete.

{LABEL}loop~{FIELD}1~		{FIELD}2~	
{NEXT RECORD}{GO}LOOP~			

Customers who want to have other information merged in after the Table Merge can set up the Merge as follows:

{LABEL}loop~{ON ERROR}{GO}finish~~{FIELD}1~		{FIELD}2~	
{NEXT RECORD}{GO}LOOP~{LABEL}finish~			

If the merge runs into an error message, it goes to the label called finish and then stops the table. It then continues with the primary document, merging any other information after the table.

*This information is given strictly as a troubleshooting option.

Memo ID: WP51_1326

INTEROFFICE MEMORANDUM

DATE: July 30, 1991 FROM: Info. Center
PRODUCT: WordPerfect VERSION: 5.1
RELEASE DATE: ALL
SUBJECT: Merge/Macro - Cannot Use Greater Than Or Equal

To Combination

When creating a primary merge file or a macro, you cannot use the "greater than or equal to" (>=) combination. This combination will always return a **not true** value. If it is necessary to check both whether a variable is equal to a value or greater than a value, the OR operator (|) is necessary. The following is an example of the syntax:

```
{IF}{VARIABLE}1~=10~|  
{VARIABLE}1~>10~ Perform operation A{ELSE}Perform operation B{END IF}
```

Memo ID: WP51_6401E

INTEROFFICE MEMORANDUM

DATE: August 13, 1991 FROM: Info. Center
PRODUCT: WordPerfect VERSION: 5.1
RELEASE DATE: ALL
SUBJECT: "Error - Return Without Call" - Caused By {END

FOR}

A customer received the message "Error - Return Without Call" when doing a merge. The problem was corrected by replacing an {END FOR} command with the proper {END FIELD} code.

Memo ID: WP51_6564K