

Create and Update To-Do Lists
WordPerfect Magazine
By Elden Nelson

I'm not what you'd call an organized person. I'm constantly forgetting deadlines and appointments, so I used to keep scores of tiny to-do lists on those square sticky notes.

Thank goodness for WordPerfect, its Tables feature and the macro language, especially the {FOR EACH} command. When you use these together, it's easy to create and maintain a prioritized list of things you need to do (see Figure 1 below).

The included TODO.WPM macro creates a table for the tasks you need to do or updates an existing list. The macro prompts you for new to-do items, then automatically sorts your list by importance. (You rank each item from 1 to 10 – the lower the number, the more important the task.)

Another benefit is that your list is kept as a WordPerfect file. This is nice for people like me, who keep their to-do list handy in a pocket, where it'll be later eaten by the washing machine. When this happens, just print another.

Using the Macro

Once you've created the macro, you can use it to make a new to-do list or update an earlier one. To create a new list, start at a blank document screen. Press Macro (Alt-F10), type "todo" and press (Enter). An empty table appears on your screen, along with the prompt "Type the Task, then (Enter):" Here you'll type one of the items you need to do, then press (Enter).

Next, you'll be asked to enter the due date. Type the date you need to have the item done by (in the MM/DD/YY format), then press (Enter). If the item you're listing has to be done today, just press (Enter), and the macro will automatically insert today's date.

Finally, you'll be asked to enter a priority for the item you're listing. Type a number from 1 to 10. (You can use higher numbers, if desired. And remember: the lower the number, the higher the priority.) You can also give several items the same priority number. If you have three important things to do today, you make them all priority 1. Press (Enter) after you've typed the priority number.

The prompt "Add another item?" appears. Press (Y) Yes or (N) No. If you press (Y) Yes, the whole process repeats. You can add as many items to your list as you like. If you answer (N) No, the macro sorts your tasks so the most important ones are at the top, the least important at the bottom. If more than one task has the same priority number, they're also sorted by date, with earlier items listed first. When the list is sorted, the macro ends.

Once you've finished making a list, you can save it, then exit the document. That way, if you want to add items to the list later on, you can retrieve the list and run the macro again. To do this, retrieve an existing to-do list into a document screen, press Macro (Alt-F10), type "todo" and press (Enter).

The prompts will begin appearing, allowing you to type in the new things you need to do. When you've finished adding items, press (N) No at the "Add another item?" prompt, and your to-do list will be sorted according to the priorities you've set.

Sometimes you'll want to remove items from your list. You won't need the macro to do this. Just move your cursor to the row of the item you want to remove, then press (Ctrl-Delete). (This option is available on enhanced keyboards only.) At the bottom of the screen, you'll be prompted "Delete Row? No (Yes)." Press (Y) Yes to remove the item you don't want.

Once you've finished creating or updating your list, print it and carry it with you; check the "Done" column as you complete each task.

One last note on using this macro: If you try to run the TODO.WPM macro from anywhere but a blank screen or a document with a table in it, the macro says "Be at a blank doc or a to-do table to use this macro." Press any key to make the message go away, then go to a blank document or a document that has a to-do list table in it and run the macro again.

Learning to Use the {FOR EACH} Command

Use the {FOR EACH} command when you want to repeat a certain part of your macro a number of times, with one part changed each time.

The TODO.WPM macro asks several questions: What's the task? What's the due date? What's the priority? Since they're similar questions, they're a good candidate for a {FOR EACH} loop. The parts of the {FOR EACH} command look like this:

```
{FOR EACH}variable~expr1~expr2 ~exprN~~
```

The first time through the loop, the first expression (expressions are usually words or numbers) is assigned to the variable, then any commands that follow {FOR EACH} are carried out until the {END FOR} command. Once the {END FOR} is reached, the macro goes back to the {FOR EACH} command, and the second expression is assigned to the variable. This continues until the macro has looped the same number of times as there are expressions listed. The macro then continues running, beginning with the first command after the {END FOR}.

Here's an example of how a {FOR EACH} command works. On line 9 in Figure 2, the {FOR EACH} command is shown as: {FOR EACH}Item~Task~Due Date~Priority~~. The first piece of information after the command, "Item~," is the variable that each expression is assigned to every time the loop runs.

The rest of the information, Task~Due Date~Priority~~, are the values that will be assigned to variable Item as the {FOR EACH} loop runs. Note that "Priority~~" has two tildes after it. This tells the macro that the list of expressions is done. Make sure that you always separate each piece of information with a tilde and that you end the list of expressions with two tildes.

The first time the loop runs, variable Item is assigned the value "Task." The second time, it is "Due Date" and the third time it is "Priority." Even though this particular {FOR EACH} command has three items and runs three times, you can have almost any number of expressions listed.

Now look at how the rest of the {FOR EACH} loop works. Line 10 asks you to type in certain information, using the {TEXT} command. Here it uses {VARIABLE}Item~ in its message. Since variable Item contains

either the words "Task," "Due Date" or "Priority," the message will appear as "Type the Task (or Message, or Priority), then (Enter)."

Line 11 checks to see if the current question is asking for the due date; if it is, the macro checks to see whether you pressed (Enter) instead of typing a date. If you did, line 12 inserts today's date for you.

Line 14 types variable Answer (which was assigned in the {TEXT} command, line 10) into the current cell, then line 15 moves to the next cell.

The {END FOR} on line 16 ends the {FOR EACH} loop. Depending on whether there are any other items in the {FOR EACH} list, the macro either goes back to start the loop over or continues on down from immediately below the {END FOR}.

You may have noticed that all the commands between {FOR EACH} and {END FOR} are indented one level. You don't have to do this, but most macro writers do. Why? This makes it easy to tell where the loop begins and ends.

How The Macro Works

The {FOR EACH} loop is the real star of this month's column, but the supporting cast is important, too. Here's a quick look at how the macro works. On line 3, the macro tries to go into the table editor. If a table isn't available in the current document, line 2 has the macro jump to the NewTable label (line 30), where lines 31-46 create a new to-do list. If you're not at a blank screen at this point, line 31 catches it, and lines 32-33 tell you what's wrong and stop the macro.

Once the new table is built, line 47 returns you to line 4, which takes the cursor to the bottom of the table and leaves Table Edit mode. Line 5 changes date formatting to the MM/DD/YY format. Line 6 turns macro display on, so you can see the to-do list grow as you type.

Lines 7-29 are the main part of the macro. Line 8 puts a new row in the table and moves the cursor to the correct cell. Lines 9-16, which we've already covered in excruciating detail, prompt you for the various kinds of information for an item in your list. Line 17 asks if you want to add another item, and line 18 proceeds to move the cursor so it's under the "Y" in "Yes," so you know the default answer is "Yes."

The {IF} command in line 19 means that if you answer "n" or "N," lines 21-26 will sort the items in your to-do list and end the macro. If you press any other key in line 17, the macro skips down to line 29. The {GO} command here has the macro jump back to line 7, where the macro begins assisting you in adding another item to your list.

That {FOR EACH} loop technique is great for macros that need to do similar – yet somewhat different – things several times. And the TODO.WPM macro makes me a much more organized person – or at least it would make me more organized if I could clear enough junk off my keyboard to start the macro.

■ Figure 1.



To-Do List			
Done	Task	Be Done By	Priority
	Send birthday card to Uncle Lou	09/21/93	1
	Type report for boss	09/22/93	1
	Return books to library	10/02/93	2
	Learn to play the banjo	09/22/93	3
	Join athletic club	09/26/93	5
	Shave the poodle	09/23/93	9

Figure 1. Create and update a prioritized list of things to do using TODO.WPM.