Your OneOnOne™ Guide to Better Data and Decisions

working smarter®

WITH MICROSOFT EXCEL ★ FOR ADVANCED USERS

Excel 2000–2003 *November 2006*

2 smarter ways to duplicate headers and footers

Copying headers and footers from one worksheet to another within the same workbook is relatively easy, but doing so from one workbook to another can be more daunting. One way to do so is to use the traditional editing method of copying and pasting. Select the header material you want to copy, press [Ctrl+C], display the header in the target worksheet, and then press [Ctrl+V]. The drawback to this approach is that it can involve quite a few steps. After all, there are three sections (left, center, and right) for each header and three for each footer. This means you might do six copy and paste operations to copy everything.

Creating a template

If the header or footer is one you often use in new workbooks (not existing ones), then the best approach is to create a template workbook. Set up a workbook as desired, including headers and footers, and save it as an Excel template. If you save it in the special XLStart folder, the template is used as the default for all future workbooks.

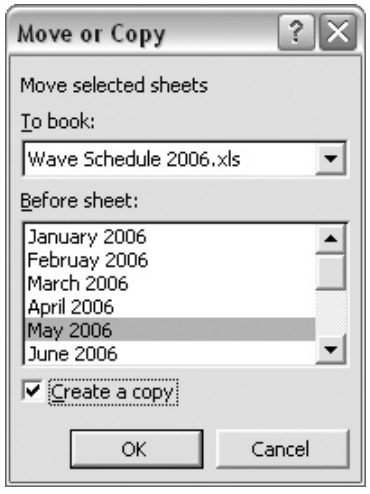

Figure A

Copying worksheets

Another way to accomplish your goal involves copying entire worksheets. What you essentially need to do is copy the original worksheet to the second workbook, copy that header and footer to the other worksheets, then delete the original worksheet.

Start by opening the target workbook; the one to which you want the headers and footers copied. Then open the workbook that is the source of your header and footer, and make ▶

In this issue...

Bigger, Better, Stronger than ever!

You told us you wanted a better newsletter, and we listened. Your *Working Smarter with Microsoft Excel for Advanced Users* is now an 8-page monthly, packed with more tips, illustrations, and challenges than ever before. Please tell us how we're doing by writing to me at editor@working-smarter.com.

Coming next month...

- An easy process for removing duplicate cells and rows
- Perform a quick sort on a data list—by more than three columns
- A trick for temporarily hiding graphics

Travel back a few centuries: Convert your numbers to roman numerals

Sometimes Excel's vast options for computing and communicating data are surprising. For instance, if you need to convert a set of numbers to roman numerals but you can't seem to get your L's and your V's straight, let Excel do the work for you.

There's a little-known function in Excel's arsenal called the ROMAN function. You can use this function to convert a static number (i.e., =ROMAN(145)) to a roman numeral. Or, just as you would with any other function, you can use a cell reference. For example, =ROMAN(A1) will convert the value in cell A1 to a roman numeral, as shown in **Figure A**.

Just remember: The ROMAN function cuts off its values to make them integers. If you type 46.8 in cell A1, cell A2 still returns XLVI. Also, you can't convert numbers larger than 3999, and negative numbers will produce an error.

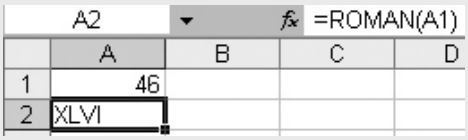

Figure A

Chart your data with one keystroke

Charts can take a lot of time to construct, and there are so many ways to customize them that it's often an intimidating task.

The next time you need a quick-and-dirty chart in seconds, just select any cell within the data table you want to chart and press [F11]. Excel creates a default Column chart in a new worksheet. From there you can either take the chart as is or tweak it to suit your needs — just as you would any other chart.

Copy headers and footers *(continued from page 1)*

sure the desired worksheet is displayed. Choose Edit | Move Or Copy Sheet to display the Move Or Copy dialog box.

Using the To Book dropdown list, select the target workbook. In the Before Sheet panel, indicate where you want the sheet copied, as shown in **Figure A**. Select the Create A Copy check box, also shown in **Figure A**, and click OK. The worksheet is copied to the target workbook, and you can close the source workbook.

Display the worksheet you just copied. In the tab area at the bottom of the window, right-click and choose Select All Sheets, as shown in **Figure B**. All the worksheets are now selected. Choose File | Page Setup, then click on the Header/Footer tab. Using the Header and Footer dropdown lists, select the header and footer used in the worksheet you just copied, as shown in **Figure C**. When you click OK, the header and footer are applied to all the worksheets. You can then delete the copied worksheet. ■

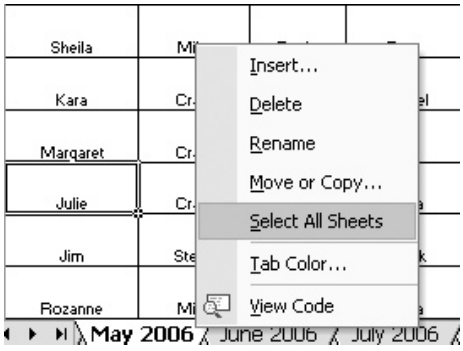

Figure B

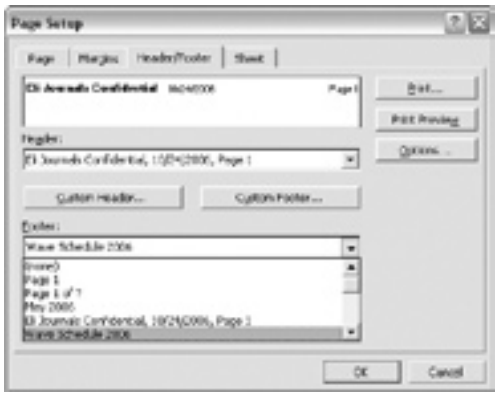

Figure C

T.O.P.S. AWARD

Are you *T.O.P.S.* (The Office Problem Solver) in your office? Surely you've encountered many challenges in your job. Is there a task or project in Microsoft Excel that you or a coworker found particularly difficult and then successfully solved? We want to hear about it.

Submit your story and examples of your work, and we'll choose a T.O.P.S. winner based on creativity and resourcefulness. We will announce the winner and print their story in a future issue. The winner will receive a free 6-month subscription to one of our other publications. Check out our titles at http://www.workingsmartertraining.com.

Email your submissions to editor@working-smarter.com with the subject line, "The Office Problem Solver."

Protect your macros so no one meddles with them

You work hard to ensure that your VBA (Visual Basic for Applications) code is perfect. In fact, that kind of attention to detail is crucial; one minor mistake can be the difference between a macro that works and a macro that doesn't work. The good news is that there's an easy way to make sure no one can view, much less edit, your code but you. All you have to do is assign a password.

Safeguard your code

When you assign a password to your VBA project, a user attempting to view the code must provide the correct password, as shown in **Figure A**.

What's the password? If you forget the password you assign to a VBA project, or you inherit a workbook with password-protected modules that you need to access, there are several third-party programs that recover lost passwords. However, we suggest you keep passwords in a safe place to avoid using a third-party program designed to break in to the securities you're trying to put in place.

To assign a password to a VBA project:

1. Launch Excel and open the workbook that includes the project you'd like to protect.
2. Press [Alt][F11] to open the VBE (Visual Basic Editor).
3. View the Project Explorer if it isn't already displayed by choosing View | Project Explorer from the VBE's menu bar.
4. Right-click on VBAProject and choose VBAProject Properties from the resulting shortcut menu.
5. Click on the Protection tab in the VBAProject-Project Properties dialog box.
6. Select the Lock Project For Viewing check box in the Lock Project panel.
7. Enter a password in the Password text box, and then re-enter the password in the Confirm Password text box.
8. Click OK to assign the password .
9. Save the change by clicking the Save button on the toolbar.

After the password is in place, you can attempt to view the code in the project by double-clicking on it in the Project Explorer. Excel prompts you for a password, and you can't view the code unless you provide the correct password.

Unprotect the VBA project

There may come a time when you no longer want to restrict your workbook's code. Fortunately, as long as you know the password, you can also remove it.
1. Open the workbook's VBE, if necessary.
2. Right-click on VBAProject in the Project window and choose VBAProject Properties from the shortcut menu.
3. Click on the Protection tab, and then deselect the Lock Project For Viewing check box.
4. Clear the Password and Confirm Password text boxes and click OK.
5. Click the Save button on the VBE's toolbar to save the changes. ■

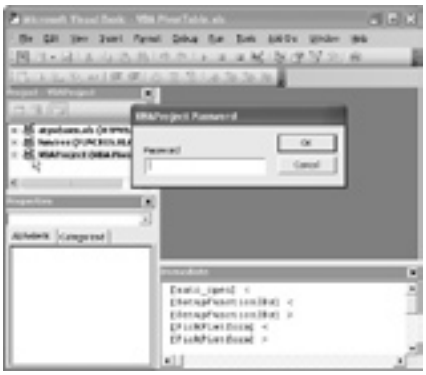

Figure A: You can't access a password-protected VBA project unless you provide the correct password.

Display your bar chart's data labels right in the bars

Data labels are tough to work with because they often make your chart look cluttered or cover up other important data points. The next time you want to display labels for your bar chart's data points, you can conserve space by embedding them right in the bars. To nest the data point labels right within the chart's bars:
1. Launch Excel and open the worksheet that contains your bar chart.
2. Right-click on one of the data labels and choose Format Data Labels from the resulting shortcut menu to display the Format Data Labels dialog box.
3. Click on the Alignment tab and then choose Center from the Label Position dropdown list.
4. Ensure that Center appears in both the Horizontal and Vertical dropdown lists.
5. Enter *90* in the Degrees spin box in the Orientation panel and click OK to apply the changes, as shown in **Figure A**.

Helpful Tips: You may need to change the color of your data labels so they contrast well with your chart's bars. You may also need to widen your chart's bars to accommodate the numbers. You can do so by decreasing the gap width between the chart's data points.

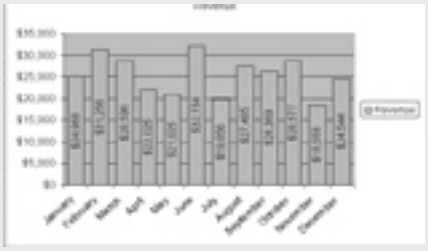

Figure A

SMART CHARTING
SMART CHARTING

Achieve a Stock chart look — even if you don't meet the Stock chart requirements

Download: www.ootraining.com/ws/samples/Contracts

If you scan the Chart Wizard's chart type options, you'll see that the Stock chart type is designed to plot high and low data. This might seem like the perfect option for charting other value ranges as well. However, the Stock chart requires a third data series for closing prices. This makes sense when you're charting stocks, but it doesn't work well for other kinds of data. Instead of adding a column of data just to fill the Stock chart's requirements, try our more efficient alternative. We'll use the Line chart type and customize it to appear as easy-to-read floating columns, as shown in **Figure A**.

Download:
You can follow along with our example by downloading and extracting the file **contracts.xls** from the URL given at the beginning of this article.

Plot a line chart

Let's say you're working with the data shown in **Figure B**. Our sample data is set up with highs and lows as follows:

- Column A lists the quarter and year chronologically.
- Column B lists the highest paying contract for each quarter.
- Column C lists the lowest paying contract for each quarter.

To create a simple line chart:

1. Select your entire data range, including column headings (in our example, cells A3:C23).
2. Click the Chart Wizard button on the Standard toolbar.
3. Select Line from the Chart Type list box in the Chart Wizard dialog box, then choose the Line chart option in the Chart Sub-type panel.
4. Click Finish.

Customize the line chart

At this point, your chart should resemble the one shown in the top portion of **Figure A**. To transform your line chart into a floating bar chart, you need to make some simple modifications to the chart's data series.

To add up/down bars and reduce the gap between them:

1. Double-click on the chart's blue line, which represents the High data series.
2. Click on the Options tab in the Format Data Series dialog box, and then select the Up/down Bars check box (Up-down Bars check box in Excel 2000).
3. Enter *0* in the Gap Width spin box.
4. Click OK to apply the changes to your chart. Your chart should now resemble the one shown in **Figure C**.

To hide the data series lines:

1. Double-click on the blue line that represents the High data series.
2. Click on the Patterns tab, and then select the None option button in the Line panel.
3. Click OK when you're finished.
4. Double-click on the pink data series line to open the Format Data Series dialog box.
5. Click on the Patterns tab, select the None option button from the Line panel, and then click OK.

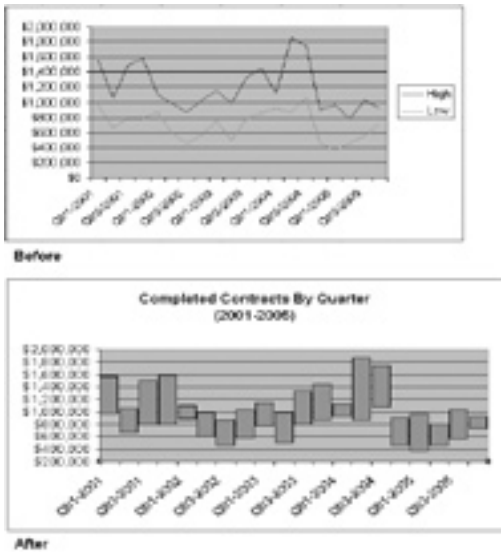

Figure A: By modifying our line chart's data series, we transformed its dull, flat lines into eye-catching floating columns.

	A	B	C	D
1	Completed Contracts by Quarter (2001-2005)			
2				
3	Quarter	High	Low	
4	Qtr1-2001	$1,566,526	$968,632	
5	Qtr2-2001	$1,055,514	$658,695	
6	Qtr3-2001	$1,489,574	$786,669	
7	Qtr4-2001	$1,586,300	$788,554	
8	Qtr1-2002	$1,102,548	$869,558	
9	Qtr2-2002	$986,298	$696,895	
10	Qtr3-2002	$886,486	$447,869	
11	Qtr4-2002	$1,023,548	$989,382	
12	Qtr1-2003	$1,147,456	$758,541	
13	Qtr2-2003	$988,799	$489,032	
14	Qtr3-2003	$1,326,200	$784,166	
15	Qtr4-2003	$1,445,898	$869,326	
16	Qtr1-2004	$1,125,405	$926,548	
17	Qtr2-2004	$1,866,296	$864,346	
18	Qtr3-2004	$1,745,200	$1,056,369	
19	Qtr4-2004	$899,636	$458,645	
20	Qtr1-2005	$965,341	$369,205	
21	Qtr2-2005	$789,566	$459,264	
22	Qtr3-2005	$1,026,547	$650,709	
23	Qtr4-2005	$922,864	$726,486	

Figure B: We'll use this sample data to create a floating column chart.

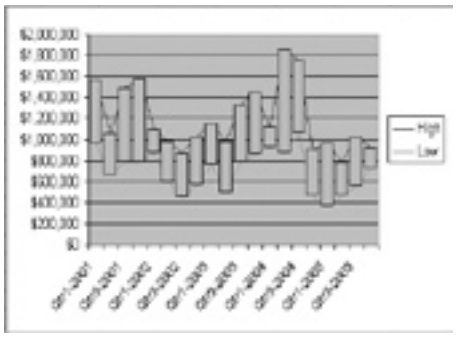

Figure C: You can create the effect of floating columns by displaying up/down bars in your line chart.

To change the color of the up/down bars:

1. Double-click on any of the up/down bars to open the Format Down Bars dialog box.
2. Choose a new color from the Area panel's color palette.
3. Click OK to apply the new color to your floating column bars.

Modify the *y*-axis' scale to increase readability

Your chart is looking better as a floating column chart, but the data points are constrained to a small section of the chart's plot area. This can make it hard to distinguish between data points.

To modify your chart's *y*-axis scale:

1. Double-click on the *y*-axis, also known as the value axis.
2. Click on the Scale tab in the Format Axis dialog box.
3. Enter *200000* in the Minimum text box, as shown in **Figure D**. You may also adjust the Maximum and Major Unit text box values depending on your data.
4. Click OK to accept the changes.

Add a few finishing touches

If you want your chart to be clear and effective, there's a little more work to do. Displaying a chart title defines your data. In addition, it's unnecessary to show a legend in this case — it's clear that the tops and bottoms of the columns illustrate each data point's highs and lows. Finally, the chart's *x*-axis labels show every other quarter, and it may be easier to correlate a floating column bar with its quarter if you define them clearly.

To display a chart title:

1. Right-click on the chart's white space to display the Chart Area's shortcut menu and choose Chart Options.
2. Click on the Titles tab in the Chart Options dialog box, if necessary.
3. Type Completed Contracts by Quarter (2001-2005) in the Chart Title text box and click OK.
4. Place the insertion point after the word Quarter and press [Enter] to place the date range below the first line of the title.

To hide the chart's legend:

1. Right-click on the Chart Area and choose Chart Options from the resulting shortcut menu.
2. Click on the Legend tab in the Chart Options dialog box.
3. Deselect the Show Legend check box, as shown in **Figure E**, and click OK.
Your final chart should look similar to the one in the lower portion of **Figure A**. ■

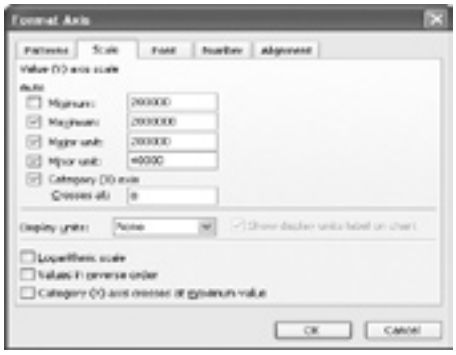

Figure D: You can change the minimum and maximum values on the y-axis to make your data easier to read.

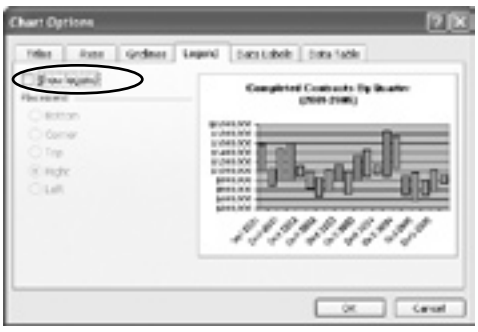

Figure E: If you don't need a legend, don't let it clutter your chart.

Trick conditional formatting into acting on data from other worksheets

Conditional formatting is an excellent tool if you know the ins and outs of using it. One of the most frustrating aspects about it, however, is the error message you receive when you try to input a conditional formatting fomula that refers to data in another worksheet. Excel tells you that you can't refer to data outside of the worksheet in your conditional formats — but we say you can.

All you need to do is create a named range for the data you want to use in another worksheet. You can create a range called CFRange in Sheet2. Then when you want to use that range in Sheet1's conditional formatting formula, simply input CFRange in the formula. Excel accepts it even though the data is from a different worksheet.

Keep your scrolling under wraps: Use a shortcut key

When you have a large set of data that spans several screens, scrolling to the end of the data isn't a great option. It's difficult to control and you often end up exceeding the last row (or column) of data.

Fortunately, there's a simple shortcut key that not only finds but also selects every cell to the right and below the currently active cell until it reaches the furthest point. Just select your starting cell and press [Shift][Ctrl][End].

Snag: If you've entered a value and then deleted it, Excel may still consider it the last cell in your worksheet's used range. So, if you had data up to row 200 and then delete the last 10 rows, the shortcut key selects up to row 200.

HELP DESK

Solve the mystery of the invisible workbook

Q *I have one workbook that doesn't display. It appears in the Window menu with a checkmark next to its name, so I know Excel acknowledges that it's open. It even prompts me to save changes when I close the program. However, all I can see is a gray, blank screen as if I didn't have any workbooks open. How can I solve this?*

A This happens occasionally, and fortunately there's a simple solution. Your workbook is opening in a minimized view.

To quickly display the workbook in its true maximized view and restore the active workbook, press [Ctrl][F10]. Your workbook should appear like magic. However, the root of the problem may be the number of windows that open when you access the file.

Now, check to see if there's a :1 or :2 appended to the filename in the title bar at the top of the application. If so, your window arrangement may be off. This can cause some problems when you open the workbook, especially if you've closed (and saved) the workbook while unintentionally having two instances of it open. When you open it again, it will open two windows that display the same workbook. Excel adds the numbers to the workbook name to distinguish the identical workbooks from one another. To view all open workbooks and rearrange them:

1. Launch Excel and open the workbook that you're having trouble viewing.
2. Choose Window | Arrange from the menu bar to access the Arrange Windows dialog box.
3. Select the Tiled option button from the Arrange panel and click OK to view each open instance of the workbook side by side.

Flip flop your data for a different point of view

Q *I have a data table that has employees listed down the left side and dates listed across the top. How can I rotate the table so that employees are on the top and dates on the side?*

A The name for what you are trying to do is called *transposing*—rotating the orientation of a data table so that rows are now columns and columns are now rows. Excel makes transposing easy, but you need to know where to find the command (it isn't in a particularly intuitive place).

Start by selecting the data table you want to transpose, including any header rows or columns. Press [Ctrl+C], which copies the data to the Clipboard. Select a blank cell somewhere below the data table, or select a cell in a blank worksheet. Choose Edit | Paste Special and select the Transpose check box, as shown in **Figure A**. When you click OK, the data table pastes into the new location, but it is transposed from its original orientation. You can then delete the original data table, as desired.

Figure A

The secret to sorting just a portion of a data table

Q *I have a range of cells I want to sort. I don't want to sort the entire data table; just the cells I select. How can I do that?*

A Excel allows you to limit your sorting rather easily. As an example, let's assume you want to sort the cells in the range of C10:F40, based on the values in column D.

To do this, select the range of cells, then choose Data | Sort. Using the Sort By dropdown list, choose Column D. Make sure the No Header Row option button is selected, then click OK. Only the data you selected is sorted; all other information in the data table remains unaffected.

You can also sort by using the Sort Ascending or Sort Descending tools on the toolbar, but the process is just a bit different. Select the range of cells, as before, but press the Tab key to make sure that Column D is selected. (You can tell when it is selected because one of the cells in column D will appear in white, rather than being shaded.) Now click the Sort Ascending or Sort Descending tools, as desired.

Uncover the hidden equations in trendlines

Q *I created a chart and included a trendline for my data series. Is there a way to figure out the equation used for the trendline so I can calculate other points along the line?*

A You can add trendlines to charts to help "smooth out" the data so you can more clearly see the movement represented by a series of data points. If you want to figure out the equation used by Excel to plot the trendline, the easiest way is to ask Excel to display that equation.

Right-click on a data series and choose Add Trendline. Excel displays the Add Trendline dialog box. Choose the type of regression you want used for the trendline, then click on the Options tab. Make sure the Display Equation On Chart check box is selected, then click OK.

Excel will now display a formula on the chart that represents how it calculated each point along the line. You can use this formula to determine points, as well. Once you know the formula, you can turn off the formula display if you'd like. ■

Expertly control how your program executes by flexing its For ... Next structure

Macros in Excel are written in a language called Visual Basic for Applications (VBA). Like any other programming language, VBA includes programming structures that are used to control how the program executes. One of these structures is the For … Next structure. This structure is used to repeat a block of program code a set number of times.

The most common use of the For … Next structure has the following syntax:

```
For X = 1 To 99
    program statements
Next X
```

You aren't limited to using the X variable; you can use any numeric variable you desire. You also aren't limited to the numbers 1 and 99 in the first line; you can use any numbers you desire, or you can use numeric variables.

When a macro executes and encounters this structure, Excel repeats every program statement between the *For* and *Next* keywords a set number of times. In the syntax example, the statements would be executed 99 times (1 through 99). The first time through the structure, X would be equal to 1, the second time through it would be equal to 2, then 3, 4, 5, and so on, until it equaled 99 on the last iteration.

The loop variable (X) is often used in the program statements in the middle of the loop. A common example would be to reset all the elements in an array, as shown here:

```
For X = 1 to 24
    iMyArray(X) = 0
Next X
```

To illustrate how the value of X varies when stepping through a For … Next loop, try the **TestLoop** macro found at **www.ootraining/ws/adv-excel**. (All the macro code for this article can be found there.) This example displays a message box that shows the values of both *X* and iCount during each iteration of the For … Next loop. Before the loop is entered, iCount is set to zero, and then the value of X is added to iCount each time through the loop. The text in sTemp is "built up" with each iteration of the loop.

Changing how For... Next counts

As VBA is working through the For … Next structure, it normally increments the counter by one on each iteration. You can also add a Step modifier to the For … Next structure, thereby changing the value by which the counter is incremented. For instance, consider the following example:

```
For X = 1 To 99 Step 5
    program statements
Next X
```

The first time through this example, X equals 1, and the second time through, X equals 6 because it has been incremented by 5. Similarly, the third time through, X equals 11. You can also use negative numbers for Step values to count downward.

Exiting a loop early

Stepping through a For … Next loop can take a lot of time. To minimize this, check only what you need. Consider the following code, which checks an array to see if a value exists. If it doesn't, then it adds the value to the end of the array. If it does, then the value isn't added.

```
AddIt = False
For J = 1 to NumEntries
    If NumValues(J) = ToAdd Then AddIt = True
Next J
If AddIt Then
    NumEntries = NumEntries + 1
    NumValues(NumEntries) = ToAdd
End If
```

This works, but if the array gets large, you can end up going through the For ... Next loop quite a few times. The following code dumps out of the For ... Next loop early if a match is detected.

```
AddIt = False
For J = 1 to NumEntries
    If NumValues(J) = ToAdd Then
        AddIt = True
        Exit For
    End If
Next J
If AddIt Then
    NumEntries = NumEntries + 1
    NumValues(NumEntries) = ToAdd
End If
```

If a match is found early on in the loop, the rest of the iterations are skipped because the Exit For statement is encountered and the loop exits right away. The result is a faster running macro. ■

INSTANT TUTORIAL

Don't fret! You can use data validation across multiple workbooks

The data validation feature of Excel is quite handy. You can use it to specify a range of values that are considered acceptable for user input. If a user tries to enter a value that is not in the range you specify, then Excel notifies the user and (optionally) denies the entry.

Data Validation caveat: Normally, Excel expects you to specify your validation range as being on the same worksheet where you are setting up the validation rule. This can be a real bother, since the validation range could entail quite a bit of data, and including it on the same worksheet as your rule may mess up the appearance of the worksheet.

The problem is that if you try to enter a range that is on another worksheet or in another workbook, Excel balks and gives you an error message. There is a way around this roadblock, however.

Say that you have worksheet where you're supposed to enter an employee name and phone extension. Click the Standard toolbar's New tool to open a blank workbook. On Sheet1, in cell A1, type *Employee*; and in B1, type *Ext*.

	A
1	Atkinson, Chris
2	Curtis, Dave
3	Garrison, Toni
4	Johnson, Doug
5	Kinerley, Frank
6	Nelson, Bill
7	Sorendi, Rachel
8	Thomas, Pat
9	Vanderton, Paul
10	Wilson, Nancy

Figure A

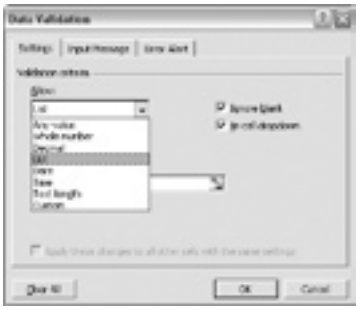

Figure B

Set up the employee list

To assist with entry, you want to specify a list of valid employee names, and then set up data validation to present the list to users. Of course, you don't want the list of employees on the same worksheet; having them on a different worksheet would make matters so much nicer. Click on Sheet2, and enter the list of employee names as shown in **Figure A**. Select the entire list, then choose Insert | Name | Define to display the Define Name dialog box. Replace the suggested name with a unique name for your data validation range. In the Names In Workbook text box, type *Employees*. Click Add and then click OK to dismiss the dialog box. You've now created a named list for your employees. Save the file as DataValid.xls.

Define the validation rule

Click on Sheet1 and select the range A2:A10, the cells for which you want to define a validation rule. Choose Data | Validation to display the Data Validation dialog box.

Open the Allow dropdown list to see the different types of data you can use for your validation, as shown in **Figure B**. For this example, select List, since your data validation range consists of a list of names. Selecting List changes the options in the dialog box. (Different selections in the Allow dropdown list produce different configurations of the dialog box.)

In the Source text box, enter *=Employees* (an equal sign followed by the name you assigned to your list). It's also a good idea to select the two check boxes at the right side of the dialog box (**Figure B**).

Click OK to close the Data Validation dialog box. To test this, in A2, enter your own name. You should get a message that says the value you entered is not valid. Click Cancel, and enter *Wilson, Nancy* instead. Your data validation rule works—it used the data from Sheet 2 within that workbook and accepted your entry. Close the DataValid .xls file without saving, and reopen it.

Data in a different workbook

If you want to use a data validation range that is in a different workbook entirely, you need to trick Excel into accepting your external reference.

Click the New button to open a new workbook. Click in any cell, and choose Data | Validation. Choose List from the Allow dropdown list, and in the Source text box, enter the following formula:

```
=INDIRECT("[DataValid.xls]
Sheet2!A10")
```

This formula uses the INDIRECT function to return the value in another worksheet, and the data validation feature accepts it with no problems. In this example, the cell being checked is at cell A10 on Sheet2 of DataValid.xls. In order for this to work, make sure that both the worksheet you are working in and the worksheet whose data you are referencing are open at the same time. ■

OneOnOne ComputerTraining
OneOnOne Computer Training
P.O. Box 12038
Durham, NC 27709
Searchable archive: www.ootraining.com/ws-search.html

Editor: Rebecca O'Keefe
Managing Editor: Craig Watkins
Publisher: Sean Kavanagh
editor@working-smarter.com
Subscriptions: (800) 424-8668
In UK/Europe, call +44 (0)1371 810680

ISSN 1530-4337
WEA 0611